

**УКРАЇНА****(19) UA****(11) 123445****(13) C2****(51) МПК****H04L 12/22 (2006.01)****H04L 9/14 (2006.01)**

**НАЦІОНАЛЬНИЙ ОРГАН
ІНТЕЛЕКТУАЛЬНОЇ
ВЛАСНОСТІ
ДЕРЖАВНЕ ПІДПРИЄМСТВО
"УКРАЇНСЬКИЙ ІНСТИТУТ
ІНТЕЛЕКТУАЛЬНОЇ
ВЛАСНОСТІ"**

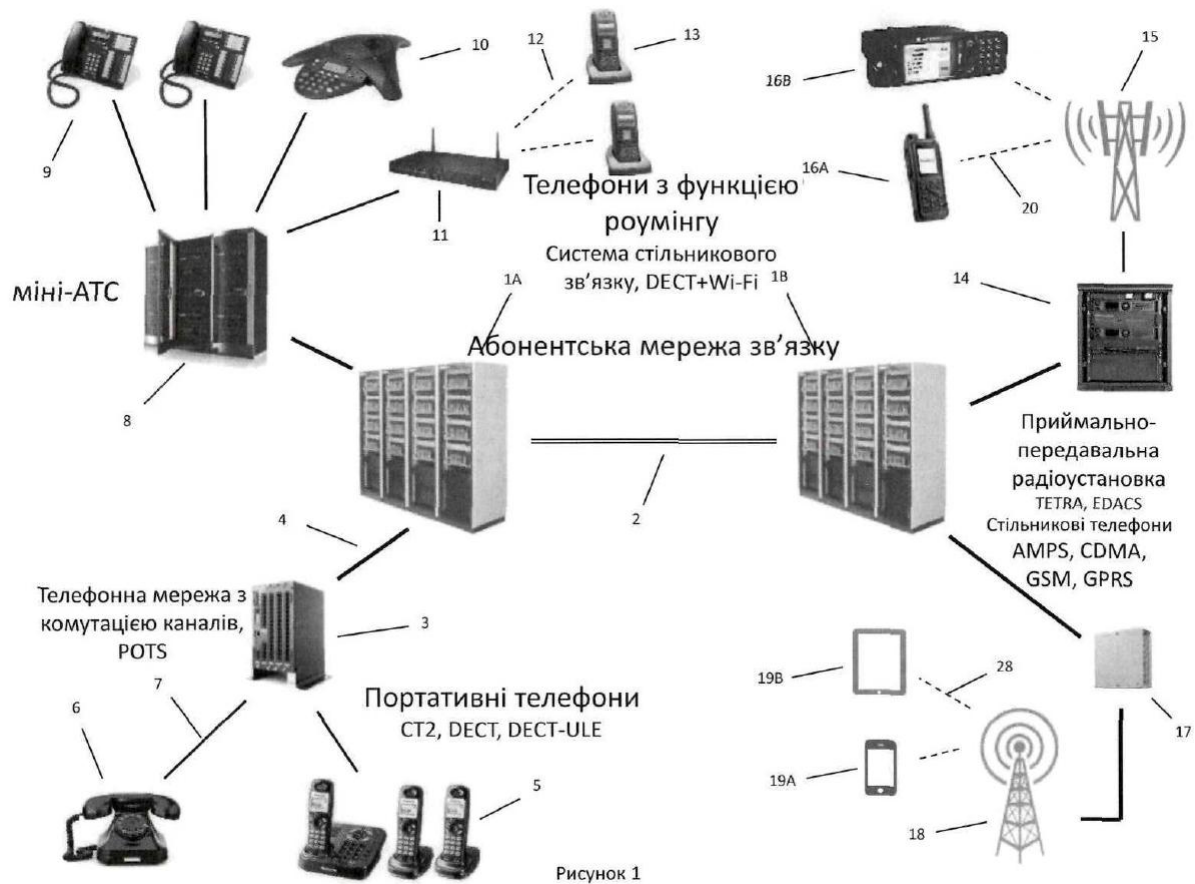
(12) ОПИС ДО ПАТЕНТУ НА ВІНАХІД

(21) Номер заявки: а 2018 07936	(72) Винахідник(и): Вільямс Річард К. (US), Верзун Євген (UA), Олександр Голуб (UA)
(22) Дата подання заявки: 23.01.2016	(73) Володілець (володільці): АДВЕНТІВ АЙПІБАНК, 10292 Norwich Ave., Cupertino, CA 95014, United States of America (US), Вільямс Річард К., 10292 Norwich Ave., Cupertino, CA 95014, United States of America (US)
(24) Дата, з якої є чинними права інтелектуальної власності: 08.04.2021	(74) Представник: Кістерський Тимофій Арсенійович, реєстр. №457
(31) Номер попередньої заявки відповідно до Паризької конвенції: 62/107,650, 14/803,869	(56) Перелік документів, взятих до уваги експертизою: EP 1802119 A1, 27.06.2007 US 20040160903 A1, 19.08.2004 Matalas, Yannis & Dragios, Nikolaos & Karetsos, George. A Scalable Framework for Content Replication in Multicast-Based Content Distribution Networks. 4267. - 2006. - pp. 110-115
(32) Дата подання попередньої заявки відповідно до Паризької конвенції: 26.01.2015, 20.07.2015	
(33) Код держави-учасниці Паризької конвенції, до якої подано попередню заявку: US, US	
(41) Публікація відомостей про заявку: 10.01.2019, Бюл.№ 1	
(46) Публікація відомостей про державну реєстрацію: 07.04.2021, Бюл.№ 14	
(86) Номер та дата подання міжнародної заявки, поданої відповідно до Договору РСТ PCT/US2016/014643, 23.01.2016	

(54) ДИНАМІЧНА ЗАХИЩЕНА КОМУНІКАЦІЙНА МЕРЕЖА ТА ПРОТОКОЛ**(57) Реферат:**

У безпечній хмарі для передачі пакетів даних у цифровому форматі пакети можна багаторазово скремблювати (тобто порядок розташування сегментів даних змінюється) з наступним дескремблюванням, розділенням на частини та подальшим змішуванням, та/або шифрувати з наступним дешифруванням у міру їхнього руху через медіа-вузли в хмарі. Методи, використовувані для скремблювання, розділення на частини, змішування та шифрування пакетів можуть бути різними залежно від параметрів стану, таких як час, внаслідок чого виконання поставленого хакером завдання стає практично неможливим, оскільки він або вона може побачити тільки фрагмент пакета, а методи, використовувані для маскування даних, постійно змінюються.

UA 123445 C2



Область техніки, до якої належить винахід

Винахід відноситься до мереж зв'язку, в тому числі до методів та пристроїв, призначених для оптимізації продуктивності та якості обслуговування, забезпечення цілісності даних, максимального збільшення часу роботи системи та стабільності мережі, а також забезпечення

5 конфіденційності та захищеності.
Огляд відомих технічних рішень

Удосконалення систем зв'язку сприяло прогресу цивілізації з найбільш ранніх стадій розвитку людства. Від кур'єрів та посланців, піших або на коні; доставки поштових відправлень на поїзді, автомобілі та літаку; до появи телеграми та телеграфу, телефону, радіо, телебачення, комп'ютерів, стільникового телефону; Інтернету, електронної пошти та Всесвітнього павутиння; а останнім часом через соціальні мережі, голосовий зв'язок через Інтернет, M2M-підключення, Інтернет речей (IoT) і Інтернет всього (IoE), зв'язок завжди прокладав шлях до використання новітніх технологій. З впровадженням кожного нового покоління телефонного технологічного обладнання число людей, між якими встановлювався

15 зв'язок, і швидкість передачі інформації між ними також збільшувалася.
Ефект цієї тенденції полягає в тому, що людство взаємопов'язане більш ніж коли-небудь в історії, при цьому люди довіряють комунікаційним технологіям та покладаються на них як на засіб безпечної та надійної доставки своєї конфіденційної, особистої, сімейної та фінансової інформації тільки тим, з ким вони мають намір зв'язатися. Знання та інформація зараз можуть в лічені секунди поширюватися серед мільйонів людей, а для друзів та членів сім'ї спілкуватися один з одним, попутно займаючись іншими справами, так само просто, як натиснути кнопку. Часто кажуть: "світ став дуже малим".

Незважаючи на те, що такий прогрес надзвичайно корисний для всіх, у нашій великій залежності від технологій є також і негативні наслідки. Не дивно, що, коли система зв'язку не виконує свої функції, наприклад, під час землетрусу або при складних погодних умовах, люди стають дезорієнтованими або навіть впадають в паніку від того, що їх "відключили", навіть якщо це тимчасово. Якість обслуговування, або QoS (англ. Quality of Service – якість обслуговування), системи зв'язку при цьому є критично важливим показником ефективності мережі зв'язку. Душевний спокій, фінансове благополуччя, самосвідомість і навіть саме життя людей залежать від надійності та захищеності з'єднання.

Іншим ключовим аспектом мережі зв'язку є її здатність забезпечити конфіденційність, безпеку та захищеність для клієнта, що її використовує. З розвитком комунікаційних технологій також розвинулась і витонченість злочинців та "хакерів", які мають намір наносити шкоду, руйнувати системи, красти гроші та випадково або зловмисно шкодити іншим. Шахрайство з кредитними картами, викрадення паролів, крадіжка особистих даних і несанкціонована публікація конфіденційної інформації, особистих фотографій, файлів, електронних листів, текстових повідомлень та приватних твітів (викрадених, щоб скомпрометувати або шантажувати жертву) є лише кількома прикладами сучасної кіберзлочинності.

Щоб виділити епідемічну частку проблеми захищеності в сучасних відкритих мережах зв'язку, нижче перераховані відомі приклади порушень конфіденційності та кіберзлочинності на момент оформлення даної заявки на видачу патенту (наведені в хронологічному порядку):

- "Мета: викрадення інформації не менше ніж у 70 млн. чоловік", - CNBC, 10 січня 2014 р.
- "Хакери відправляють шкідливі електронні листи на розумний холодильник та телевізор", - BGR (www.bgr.com), 20 січня 2014 р.

45 - "Nest відновлює роботу з Google після проблем з конфіденційністю та злому термостата", - Slash Gear (www.slashgear.com) 24 червня 2014 р.

- "Захищеність даних утриманні під питанням через піратське захоплення облікових записів. Лінія, керована додатком безкоштовного виклику та обміну повідомленнями, була вражена нещодавною низкою порушень безпеки даних. Додаток виявив, що до сотень облікових записів користувачів отримано несанкціонований доступ сторонами, які не є користувачами цих облікових записів", - Nikkei Asian Review, 2 липня 2014 р.

- "Звичайні американці, залучені до перевірки даних Агентством національної безпеки, заявляють про претензії", - AP, 6 липня 2014 р.

- "Витік паролів Wi-Fi через розумні світлодіодні лампи", - BBC News, 8 липня 2014 р.

55 - "Шестеро людей звинувачені в шахрайстві під час продажу квитків на престижні заходи через StubHub." StubHub був мішенню хакерів, які використовували вкрадені паролі та номери кредитних карт для покупки та продажу тисячі квитків на концерти поп-музики і гри "Нью-Йорк Янкіз", - зробила заяву влада Нью-Йорка", - Bloomberg 24 липня 2014 р.

- "Дослідження показують, що Інтернет речей" дуже легко піддається злому", - International Business Times (www.ibtimes.com), 4 серпня 2014 р.

- "Російські хакери вкрали понад мільярд інтернет-паролів", - New York Times, 5 серпня 2014 р.

- "Новий витік інформації, яка розкриває секрети США, - підсумовує уряд", - CNN, 6 серпня 2014 р.

5 - "Хакери отримують доступ до кореневого каталогу термостата Google Nest за 15 секунд", - Enquirer (www.theinquirer.net), 11 серпня 2014 р.

- "Dairy Queen зламано тим же шкідливим ПЗ, яке вразило Target", - Christian Science Monitor, 29 серпня 2014 р.

10 - "Жертви серед знаменитостей в результаті несанкціонованого доступу до фотографій в оголошеному вигляді - вразливість захисту облікових записів iCloud", - CBS News 1 вересня 2014 р.

- "Home Depot може стати новою метою несанкціонованого доступу до кредитних карток ... Злом Home Depot може мати набагато серйозніші наслідки, ніж Target (40 млн. карток вкрадено протягом трьох тижнів)", - Fortune 2 вересня 2014 р.

15 - "Таємничі фейкові вежі стільникового телефонного зв'язку перехоплюють дзвінки на всій території США", - Business Insider 3 вересня 2014 р.

- "Хакерська атака: від банків до роздрібної торгівлі, ознаки кібервійни?" - Yahoo Finance 3 вересня 2014 р.

- "Home Depot підтверджує злом платіжної системи в магазинах США та Канади", - Fox News 9 вересня 2014 р.

20 - "Yahoo веде судову війну з урядом США через стеження", - CBS/AP 11 вересня 2014 р.

- "Ваша медична карта для хакерів варта більше, ніж ваша кредитна карта", - Reuters, 24 вересня 2014 р.

25 - "Червоний рівень тривоги: HTTPS зламаний. Використання браузера для атаки на SSL/TLS (BEAST) буде вважатися одним з найгірших зломів (worst hacks [орфографія та пунктуація оригіналу]), тому що ставить під загрозу з'єднання з браузером, на безпеку якого щодня покладаються сотні мільйонів людей", - InfoWorld, 26 вересня 2014 р.

- "Кібератака Sony, що почалася з дрібної неприємності, стрімко переросла у великий скандал", - New York Times, 30 грудня 2014 р.

30 Дивлячись, як відбувається ескалація темпів кіберзлочинності, порушень безпеки, крадіжки особистих даних та вторгнення до приватного життя, виникає питання: "Чому всі ці кібератаки можливі та що можна зробити, щоб зупинити їх?" У той час як суспільство прагне до підвищення конфіденційності та захищеності, споживачі також потребують більшої комунікабельності, більш дешевого зв'язку підвищеної якості, а також в підвищенні зручності проведення фінансових транзакцій.

35 Щоб зрозуміти, в чому полягають обмеження якісних характеристик та вразливість сучасних мереж зв'язку, засобів зберігання даних і мережевих пристроїв, спочатку потрібно зрозуміти, як сучасний електронний, оптичний та радіозв'язок працює, транспортує та зберігає дані, в тому числі файли, електронну пошту, текст, аудіоінформацію та відеозображення.

Робота телефонної мережі з комутацією каналів

40 Електронний зв'язок включає в себе велику кількість апаратних компонентів або мережевих пристроїв, підключених за допомогою дротів, радіоліній, радіорелейних або волоконно-оптичних ліній зв'язку. Інформація передається з одного пристрою до іншого шляхом відправки електричної або електромагнітної енергії через цю мережу з використанням різних методів вбудовування або кодування інформаційного "контенту" в потік даних. Теоретично закони фізики встановлюють максимальну швидкість передачі даних для таких мереж що дорівнює швидкості світла, але в більшості випадків практичні обмеження в кодуванні даних, маршрутизації і управлінні трафіком, відношення сигнал-шум, а також подолання електричного, магнітного та оптичного шуму і небажаних паразитних сигналів спотворюють потік інформації або створюють перешкоди його проходженню, обмежуючи можливості мережі зв'язку до якоїсь частини від її ідеальної продуктивності.

50 Історично, електронна передача даних була вперше здійснена з використанням виділених "дротяних" електричних з'єднань, що утворюють "мережу" зв'язку між двома або кількома електрично мережевими пристроями. При роботі телеграфу за допомогою механічного ключа вручну замикають і розмикають електричну мережу постійного струму, намагнічуючи соленоїд, який в свою чергу переміщує металевий важіль, викликаючи в слухаючому пристрої або "реле" клацання з такими ж інтервалами часу, з якими відправник натискає ключ. При цьому відправник використовував узгоджену мову, наприклад, азбуку Морзе, для кодування інформації та перетворення її в імпульсний потік. Аналогічно, слухач також повинен був зрозуміти азбуку Морзе - послідовність довгих і коротких імпульсів, які називаються точками та тире, щоб розшифрувати повідомлення.

Пізніше Александер Грем Белл розробив перший телефон, користуючись поняттям "хвилеподібного струму", нині званого змінним струмом, щоб переносити звук за допомогою електричного з'єднання. Телефонна мережа складалася з двох магнітних перетворювачів, з'єднаних електричною мережею, в якій кожен магнітний перетворювач складався з рухомої діафрагми та котушки, або "звукової котушки", оточеній нерухомою оболонкою з постійних магнітів. Під час розмови біля цього перетворювача, зміна тиску повітря, що була викликана звуком, змушує звукову котушку виконувати зворотно-поступальне переміщення в навколишньому магнітному полі, створюючи змінний струм в котушці. На стороні слухача струм, що змінюється в часі, протікаючи в звуковій котушці, створює ідентичне коливання та магнітне поле, що змінюється в часі, протилежне навколишньому магнітному полю, змушуючи звукову котушку виконувати зворотно-поступальне переміщення у такий же спосіб, що і пристрій, який записує звук. Результуюче переміщення забезпечує відтворення звуку у такий же спосіб, що і пристрій, який записує звук. Говорячи сучасною простою мовою, коли перетворювач перетворює звук в електричний струм, він працює як мікрофон, а коли перетворювач перетворює електричний струм у звук, він працює як гучномовець. Крім того, оскільки електричний сигнал, що передається, є аналогом звукового сигналу, який переноситься як звичайна хвиля тиску в повітрі, тобто звук, на даний час такі електричні сигнали називають аналоговими сигналами або аналоговими коливаннями.

Оскільки перетворювач, як було описано вище, використовується і для розмови, і для прослуховування, під час бесіди обидві сторони повинні знати, коли говорити і коли слухати. Як в струні зі з'єднаних жерстяних банок, в такій системі абонент не може одночасно говорити та слухати. Незважаючи на те, що така одностороння робота, яка називається "напівдуплексним" режимом, може здатися архаїчною, насправді вона як і раніше широко використовується в радіозв'язку в даний час в портативних радіостанціях, а в сучасній телефонії вона називається "push-to-talk" (натисни та говори) або РТТ.

Згодом широкого поширення набули повнодуплексні (тобто, двосторонні або приймально-передавальні) телефони з окремими мікрофонами та гучномовцями, де сторони могли говорити та слухати одночасно. Але навіть сьогодні при роботі з повнодуплексним телефонним зв'язком потрібно приділяти належну увагу запобіганню зворотного зв'язку – стані, при якому звук, що надходить у мікрофон, повертається до того, хто телефонує, створюючи відлуння, а іноді і неприємний свист - проблеми, які особливо характерні для телефонного зв'язку на великій відстані.

Ранні телеграфні та телефонні системи мали ще один недолік - відсутність конфіденційності. У цих ранніх реалізаціях мереж зв'язку кожен, підключений до цієї мережі, чує все, що передається по даній мережі, навіть якщо він цього не хоче. У сільських телефонних мережах ці загальні мережі називалися "лініями колективного користування". Потім система телефонного зв'язку швидко перетворилася в багатоканальні мережі, в яких місцевий телефонний вузол підключав виділені канали безпосередньо до телефонів індивідуальних клієнтів. У місцевому пункті зв'язку системний оператор вручну з'єднував абонентів один з одним через комутатор за допомогою переминок, а також мав можливість підключатися до іншого пункту зв'язку, та вперше надавати послуги телефонних переговорів "на великій відстані". Великі комплекси, що містять велику кількість реле, що утворюють телефонні "комутаційні" мережі, поступово замінювали людину-оператора, а згодом та вони були замінені електронними перемикачами на вакуумних лампах.

Після того, як в кінці 1950-х років в компанії Bell Laboratories розробили транзистор, автоматичні та місцеві телефонні станції з їх крихкими і швидконагріваючими лампами замінили спочатку твердотільними пристроями - що охолоджуються під час роботи - на транзисторах, а потім і пристроями на інтегральних схемах. З розвитком мережі, число цифр телефонного номера збільшувалось від семизначного префікса та особистого номера до додавання кодів регіонів і, у підсумку, кодів країн, для обробки міжнародних викликів. Мідні кабелі, що передають голосові виклики, швидко покривали весь світ та перетинали океани. Незалежно від величини мережі, принцип дії залишався незмінним - виклики представляли собою пряме електричне з'єднання або "мережу" між абонентами з передачею голосу аналоговими сигналами та з маршрутизацією виклику, яка визначається автоматичною телефонною станцією. Таку телефонну систему зрештою стали називати "телефонною мережею з комутацією каналів" або в розмовній мові - традиційною телефонною мережею (POTS – Plain Old Telephone System). Пік розвитку телефонії з комутацією каналів припав на 1980-і роки, а згодом відбувалася її планомірна заміна "телефонією з комутацією пакетів", що розглядається в наступному розділі.

Майже паралельно з телефонною мережею відбувався розвиток регулярного радіозв'язку, який починався з радіомовлення в 1920-х роках. Це мовлення було односпрямованим,

здійснювалося радіомовними станціями на спеціальних частотах, що надавались за ліцензією уряду, а прийом здійснювався будь-якою кількістю радіоприймачів, налаштованих на цю частоту мовлення або радіостанцію. Радіомовний сигнал був аналоговим або з амплітудною модуляцією (АМ), або пізніше з частотною модуляцією (ЧМ) у виділеній ділянці ліцензійного радіоспектра. У США для управління наданням і регулюванням таких ліцензійних діапазонів була створена Федеральна комісія зі зв'язку (FCC). Концепція мовлення була поширена на ефірні телевізійні програми, які використовують радіопередачу, яка спочатку складалась з чорно-білого, а потім і кольорового контенту. Згодом також з'явилися можливості доставляти телевізійні сигнали до будинків людей або за допомогою радіорелейної супутникової антени, або через коаксіальні кабелі. Оскільки будь-який слухач, налаштований на конкретну частоту мовлення, може приймати мовленнєвий сигнал, термін "мультимовлення" тепер використовується для односпрямованого багатокористувацького зв'язку.

Одночасно з появою радіомовлення почалося створення перших систем двостороннього зв'язку для торгових та військових океанських суден, та до початку Другої світової війни радіостанції перетворилися в дуплексні переносні приймачі, які об'єднують передавачі та приймачі в єдиному пристрої. Як і телефонні системи, перші системи повнодуплексного радіозв'язку працювали в "симплексному" режимі, що дозволяє здійснювати передачу тільки однієї радіомовної станції по одному радіоканалу, в той час як інші слухали. Після об'єднання передавачів та приймачів, що працюють на різних частотах, стали можливими одночасна передача та прийом на кожному кінці радіолінії, забезпечуючи можливість повнодуплексного режиму зв'язку між двома сторонами.

Однак для запобігання перекриття передачі від декількох сторін для управління каналом зазвичай використовується протокол, званий напівдуплексним або push-to-talk (натисни і говори), що дозволяє здійснювати передачу виключно за певним каналу за принципом "першим прийшов - першим обслужений". Стандартні промислові типи радіостанцій, що використовують аналогову модуляцію, включають аматорські (хем-радіо та безліцензійні) радіостанції, морські УКХ-радіостанції, радіостанції універсальної об'єднаної системи зв'язку UNICOM (Universal Integrated Communication) для управління повітряним рухом і радіостанції FRS для особистого спілкування. У цих мережах двостороннього радіозв'язку радіостанції відправляють свої дані по конкретних частотних "каналах" на центральну вежу радіозв'язку, а базова станція підсилює та ретранслює сигнал, передаючи його по всій мережі радіозв'язку. Кількість доступних частот, що передають інформацію в широкомовній області, визначає загальну смугу пропускання системи та кількість користувачів, які можуть одночасно незалежно обмінюватися даними в цій мережі радіозв'язку.

Щоб розширити загальну пропускну здатність мережі радіозв'язку для роботи більшого числа абонентів, в 1970-х роках була продемонстрована та протягом десятиліття набула широкого поширення концепція мережі, в якій велика площа розбита на більш дрібні частини або "стільники" радіозв'язку. Стільникова концепція полягала в тому, щоб обмежити широкомовний діапазон вишки радіозв'язку меншим простором, тобто меншою дальністю зв'язку, і, отже, отримати можливість використовувати одні й ті ж смуги частот для одночасної роботи різних абонентів, що знаходяться в різних стільниках. Для цього було створено програмне забезпечення для управління перемиканням виклику при переході з одного стільника в сусідній без "зникнення сигналу" і раптового відключення виклику. Так само, як і традиційна телефонна мережа (POTS), повнодуплексний радіозв'язок, а також радіо- та телемовлення, спочатку мережі стільникового зв'язку були аналоговими за своєю природою. Для управління маршрутизацією викликів була прийнята система телефонних номерів для визначення належного бездротового електричного з'єднання. Цей вибір також був корисний тим, що він дозволив легко підключити нову бездротову стільникову мережу до традиційної телефонної "провідної" системи, забезпечивши міжмережеві з'єднання та спільну роботу в рамках цих двох систем.

Починаючи з 1980-х років, телефонний і радіозв'язок, поряд з радіо та телевізійним мовленням, почали невблаганний перехід з методів і форматів аналогового зв'язку на цифрові, що було обумовлено необхідністю знизити енергоспоживання та збільшити термін служби акумуляторів, поліпшити якість за рахунок підвищення продуктивності співвідношення сигнал-шум, а також почати приділяти увагу необхідності передачі даних та тексту голосом. З'явилися такі формати радіозв'язку, як EDACS і TETRA, здатні одночасно забезпечувати режими зв'язку "один-до-одного", "один-до-багатьох" та "багато-до-багатьох". Стільниковий зв'язок також швидко перейшов на цифрові формати, такі як GPRS, і телевізійне мовлення.

До 2010 року більшість країн припинило або знаходилося в процесі припинення всього аналогового телевізійного мовлення. На відміну від ефірного телебачення, у провайдерів

кабельного телебачення не було необхідності переходити на цифровий формат, і вони підтримували гібридний склад аналогових та цифрових сигналів аж до 2013 року. Їх остаточний перехід на цифрові методи був мотивований не державними стандартами, а комерційними причинами - необхідністю розширення кількості доступних каналів мережі, щоб мати можливість

доставляти контент високої (HD) та надвисокої (UHD) чіткості, пропонувати більше послуг платного телебачення (PPV, також відомих як "одностороння передача даних"), а також надавати послуги цифрового зв'язку з високою пропускнуою спроможністю для своїх клієнтів.

Незважаючи на те, що прийнято зіставляти перехід глобальних мереж зв'язку з аналогового формату на цифровий з появою Інтернету і, більш конкретно, з прийняттям і широким поширенням інтернет-протоколу (IP), перехід на цифровий формат передував комерційному визнанню IP в телефонії, забезпечуючи, якщо не прискорений, то універсальний перехід зв'язку на IP і "мережі з комутацією пакетів" (що розглядаються в наступному розділі).

Результуюча еволюція телефонного зв'язку з комутацією каналів схематично представлена на рисунку 1 як "абонентська мережа зв'язку" (PSTN), що включає об'єднання радіозв'язку, стільникового зв'язку, міні-АТС, а також POTS-з'єднань та підмереж, кожна з яких включає різноманітні технології. Ця мережа включає в себе шлюзи абонентської мережі зв'язку 1А та 1В, з'єднані магістральними лініями 2 з високою пропускнуою здатністю, а також, наприклад, підключення через дротяні з'єднання 4 шлюзу POTS 3, мережа стільникового зв'язку 17, міні-АТС 8 та мережа двостороннього радіозв'язку 14. Кожна підмережа працює незалежно, керуючи пристроями відповідного типу. Наприклад, шлюзу POTS 3, все ще поширений в сільській місцевості, з'єднується з допомогою крученої пари мідних дротів 7 зі звичайними аналоговими телефонами 6 або з портативними телефонами 5. Портативні телефони 5 зазвичай використовують технологію бездротового зв'язку (DECT), його варіант для наднизької потужності DECT-ULE або попередній йому стандарт CT2 - всі вони поширюються на виділені замкнуті радіосистеми, зазвичай з несучими частотами 0,9; 1,9; 2,4 і 5,8 ГГц. Телефони, що використовують класичний DECT, не можуть безпосередньо отримувати доступ до мереж стільникового зв'язку, незважаючи на те, що вони є бездротовими пристроями на радіо основі.

Міні-АТС 8 управляє будь-якою кількістю пристроїв, що використовуються в офісах компанії, в тому числі провідними стаціонарними телефонами 9, пристроями гучного зв'язку 10 для конференц-дзвінків, а також базовою станцією 11 приватної бездротової мережі, пов'язаної бездротовими з'єднаннями 12 з портативними або бездротовими роумінговими телефонами 13. Бездротові роумінгові телефони 13 представляють собою бізнес-орієнтоване удосконалення звичайного портативного телефону, що забезпечує доступ телефону до корпоративних Wi-Fi-з'єднань або, в випадку з системою стільникового зв'язку Японії (PHS), для доступу до загальнодоступної мікростільникової мережі, розташованої за межами компанії в місцях з великим об'ємом трафіку та в ділових районах густонаселених міст, таких як Сіндзюку в Токіо. Смуга пропускання, дальність передачі та час автономної роботи в продуктах PHS надзвичайно обмежені.

Абонентська мережа зв'язку також підключається до стільникових мереж з комутацією каналів 17, використовуючи аналогові і цифрові протоколи AMPS, CDMA і GSM. Через вежу стільникового зв'язку 18 мережі стільникового зв'язку з комутацією каналів 17 з'єднуються з використанням стандартизованих частот стільникового радіозв'язку 28 з такими мобільними пристроями, як мобільні телефони 19А. У разі мереж GPRS, надбудови над технологією GSM, мережі стільникового зв'язку з комутацією каналів 17 можуть також з'єднуватися з планшетами 19В, одночасно забезпечуючи низькошвидкісну передачу даних і голосу. Мережі повнодуплексного радіозв'язку 14, такі як TETRA і EDACS, з'єднують абонентські мережі зв'язку з портативними радіостанціями 16А і більшими вбудованими та настільними радіостанціями 16В через вишки радіозв'язку високої потужності 15 та радіоз'єднання 28. Такі мережі повнодуплексного радіозв'язку, які зазвичай використовуються працівниками поліції, швидкої допомоги, парамедиками, пожежними і навіть працівниками портових установ, також вважаються службовими мережами та службами зв'язку, та призначені для урядових, муніципальних служб та ліквідаторів надзвичайних ситуацій, а не для звичайних споживачів. (Примітка. Використовувані тут терміни "стаціонарний комп'ютер", "планшет" і "ноутбук" використовуються в якості скороченого посилання на комп'ютери відповідного типу).

На відміну від шлюзу POTS 3, мережі стільникового зв'язку 17 та міні-АТС 8, які використовують традиційні телефонні номери для виконання маршрутизації викликів, мережа повнодуплексного радіозв'язку 14 використовує виділені радіоканали (а не номери телефонів) для організації радіоліній між вишкою 15 і мобільними пристроями, які вона обслуговує. Таким чином, системи службового радіозв'язку зберігають явно виражені та унікальні відмінності від мереж стільникових телефонів для звичайних споживачів.

Рисунок 1 графічно ілюструє гнучкість абонентської мережі зв'язку для взаємного з'єднання підмереж, заснованих на різних технологіях. Саме ця різноманітність визначає внутрішню слабкість сучасних мереж з комутацією каналів - можливість спільної роботи підмереж. Оскільки різні підмережі не здійснюють зв'язок відповідно до будь-яких загальних протоколів управління або мовою, і оскільки кожна технологія обробляє передачу даних і голосу по-різному, ці різні системи по суті несумісні, за винятком їх обмеженої можливості направити телефонний виклик через основну магістраль абонентської мережі зв'язку або міжміські магістральні лінії. Наприклад, під час терористичної атаки 11 вересня в Центрі міжнародної торгівлі в Нью-Йорку багато рятувальників з різних районів США юрилися в Мангеттені, намагаючись допомогти в боротьбі з катастрофою, тільки щоб дізнатися систему радіозв'язку, їх радіостанції були несумісні з обладнанням добровольців з інших країн і міст, що унеможливило управління централізованим командуванням та контролем зусиль з надання допомоги. Через відсутність стандартизації протоколів зв'язку, їх радіостанції просто не могли встановити зв'язок одна з одною.

Крім того, при безпосередньому електричному та радіоз'єднанні телефонних мереж з комутацією каналів, особливо з використанням аналогових або незахищених цифрових протоколів, для хакера не складає труднощів за допомогою радіосканеру знаходити активні канали зв'язку і аналізувати, відбирати, прослуховувати або перехоплювати розмови, що відбуваються в даний момент часу. Оскільки абонентська мережа зв'язку утворює "постійно включений" канал або мережу зв'язку між сторонами, що спілкуються, у хакера достатньо часу, щоб ідентифікувати з'єднання і "використати його": у законний спосіб урядовими організаціями, замовивши прослуховування за рішенням федерального суду; або злочинним шляхом: кіберзлочинцями або урядовими організаціями, що здійснюють незаконне, заборонене або несанкціоноване спостереження. Визначення законного і незаконного шпигунства та спостереження та зобов'язання оператора мережі зі співробітництва в цій діяльності в різних країнах суттєво різняться і є предметом гарячих суперечок серед таких глобальних компаній, як Google, Yahoo і Apple, що працюють на багатьох міжнародних напрямках. Мережі зв'язку та Інтернет є глобальними та не знають кордонів, але закони, що регулюють таку електронну інформацію, є місцевими, та підкоряються юрисдикційному органу уряду, який в даний час контролює внутрішній та міжнародний зв'язки і торгівлю.

Незалежно від його законності або етики, електронне відстеження та спостереження сьогодні є звичайним явищем, починаючи від моніторингу за допомогою всюдисущих камер відеоспостереження, розташованих на кожному розі вулиці і над усіма автошляхами та лініями метрополітену, до складного злому і розшифровки кодів, що виконується різними підрозділами та агентствами національної безпеки держав. Незважаючи на те, що вразливі місця мають всі мережі, архаїчні та недостатні засоби захисту абонентських мереж зв'язку роблять їх особливо легкими для злому. Таким чином, абонентська мережа зв'язку, навіть підключена до захищеної сучасної мережі, є слабкою ланкою в загальній системі та створює вразливе місце, що дозволяє долати захист та здійснювати кіберзлочин. Проте, ще потрібно багато років, якщо не десятиліть, для виходу з глобальної абонентської мережі зв'язку та її повної заміни на мережу зв'язку з комутацією пакетів на основі IP. Такі мережі з комутацією пакетів (розглядаються нижче), хоча вони і більш сучасні, ніж абонентська мережа зв'язку, як і раніше не захищені від подолання засобів захисту, злому, атак, що викликають відмову в обслуговуванні (DoS-атак), та проникнення в зону конфіденційної інформації.

Робота мережі зв'язку з комутацією пакетів

Якщо роботу сучасної телефонної мережі з комутацією каналів можна образно уявити, як дві бляшані банки, з'єднані струною, то аналогічним образним поданням роботи мереж зв'язку з комутацією пакетів може служити робота поштового відділення. У такому підході текст, дані, голос та відео перетворюються в файли та потоки цифрових даних, та потім ці дані об'єднуються в квантовані "пакети" даних, які передаються по мережі. Механізм передачі заснований на електронних адресах, які однозначно визначають, куди вирушає пакет даних та звідки він надходить. Формат та протокол зв'язку також передбачають включення інформації про характер даних, що містяться в пакеті, в тому числі про контент, призначений для програми або програм, в яких він буде використовуватися; а також для обладнання, що забезпечує фізичні канали зв'язку та електричні або радіоз'єднання для передачі пакетів.

Народжена в 1960-х роках концепція мереж з комутацією пакетів була створена в параноїдальну епоху холодної війни після запуску радянського супутника. У той час Міністерство оборони США (МО) висловило занепокоєння у зв'язку з тим, що нанесення ракетно-ядерних ударів з об'єктів космічного базування може знищити всю комунікаційну інфраструктуру США, придушивши її здатність реагувати на попереджуючий удар СРСР, і що

вразливість до такої атаки може дійсно спровокувати її. Тому МО профінансувало створення надлишкової системи зв'язку або решітчастої "мережі", в якій здатність мережі передавати інформацію між військовими об'єктами не могла бути порушена внаслідок знищення будь-якого конкретного каналу передачі даних або навіть великої кількості каналів передачі даних в мережі. Ця система, названа АРПАНЕТ, стала прабатьком Інтернету та горезвісною своєю сучасних цифрових комунікацій.

Незважаючи на створення мережі з комутацією пакетів, масового поширення Інтернету не відбувалося до 1990-х років, поки в результаті спільного впливу першого простого у використанні веб-браузера Mosaic, появи веб-сторінок на основі гіпертексту, швидкого впровадження Всесвітнього павутиння, а також широкого використання електронної пошти не відбулося глобальне прийняття інтернет-платформи. Один з основоположних принципів - відсутність централізованого контролю або необхідності в центральному комп'ютері - спонукали до повсюдного використання Інтернету певною мірою тому, що ніяка країна і жодний уряд не могли його зупинити (або навіть повністю усвідомити глобальні наслідки цього), а також тому, що його призначену для користувача базу становили споживачі, які працюють на своїх нещодавно придбаних персональних комп'ютерах.

Ще одним наслідком зростання Інтернету стала стандартизація інтернет-протоколу (IP), що використовується для маршрутизації пакетів даних в мережі. До середини 1990-х років користувачі Інтернету усвідомили, що для тої самої мережі з комутацією пакетів, яка передає дані, також може використовуватися для передачі голосу, і незабаром після цього з'явилася технологія VoIP. Незважаючи на те, що концепція теоретично дозволяла будь-кому, хто має доступ до Інтернету, здійснювати голосовий зв'язок через Інтернет безкоштовно, затримки поширення в мережі знижували якість передачі мови та часто робили її нерозбірливою. Незважаючи на те, що, завдяки впровадженню високошвидкісних ліній Ethernet, високошвидкісного підключення Wi-Fi, покращенню якості зв'язку 4G на ділянці "останньої милі", час затримки зменшився, сам по собі Інтернет створювався для забезпечення точної доставки пакетів даних, а не для того, щоб гарантувати доставку цих пакетів в заданий час, тобто Інтернет не створювався для роботи в мережі реального часу.

Таким чином, мрія про використання Інтернету замість дорогих послуг провайдера телекомунікації залишилася практично невиконаною, незважаючи на наявність таких ОТТ-провайдерів (абр. від англ. Over the Top), як Skype, Line, KakaoTalk, Viber та інших. ОТТ-телефонія страждає від якості зв'язку QoS внаслідок неконтрольованої затримки в мережі, низької якості звуку, скидання виклику, луни, реверберації, зворотного поширення, переривчастого звуку та часто навіть від нездатності зробити виклик. Низька якість ОТТ-зв'язку за своєю суттю є вадою не технології VoIP, а самої мережі, в якій ОТТ-провайдери не мають можливості контролювати шлях проходження даних, або затримки, з якими здійснюється зв'язок. По суті, ОТТ-провайдери не можуть надати якість зв'язку, тому що ОТТ-зв'язок працює як автостоп в Інтернеті. За іронією долі, найкращим чином використовувати зв'язок на основі технології VoIP сьогодні можуть компанії, які є провайдерами міжнародного телефонного зв'язку з виділеними апаратними мережами з малим часом затримки - саме ті телефонні компанії, які мають найменшу мотивацію для цього.

Крім внутрішньо властивій їй мережевій надмірності, однією з найсильніших сторін зв'язку з комутацією пакетів є її здатність передавати інформацію від будь-якого джерела в будь-який пункт призначення, якщо дані впорядковані в пакетах, відповідних Інтернет-протоколу, та за умови, що пристрої зв'язку підключені до мережі та пов'язані з Інтернетом. Інтернет-протокол управляє здатністю мережі доставляти корисне навантаження в пункт призначення, не піклуючись і не турбуючись про те, яка інформація передається, або про те, яка програма буде її використовувати, виключаючи при цьому необхідність в налаштуванні програмних інтерфейсів і дорогому пропрієтарному обладнанні. У багатьох випадках навіть додатки, пов'язані з корисним навантаженням, налаштовують на задані формати, наприклад, для читання електронної пошти, для відкриття веб-сторінки в браузері, для перегляду зображення або відео, перегляду файлу прошивки або читання PDF-документа і т.д.

Оскільки він використовує універсальний формат файлів, що дозволяють уникнути залежності від пропрієтарного або фірмового програмного забезпечення, Інтернет можна розглядати як комунікаційну платформу з "відкритим вихідним кодом", здатну встановлювати зв'язок з найширшим діапазоном мережних пристроїв: від комп'ютерів до стільникових телефонів, від автомобілів до побутової техніки. Сучасний вислів, що описує цю універсальну зв'язність - це "Інтернет всього" (IoE).

На рисунку 2 показано лише кілька прикладів таких мережних пристроїв, підключених до Інтернету. Згідно рисунку, велика кількість комп'ютерів, в тому числі швидкодіючих хмарних

серверів 21A, 21B і 21C, і хмарне сховище даних 20, з'єднані лініями зв'язку 23 з високою пропускну здатністю, зазвичай оптоволоконними, поряд з безліччю інших серверів (не показані), що формує хмару Інтернету 22. Образне уявлення у вигляді хмари в даному випадку доречно, тому що немає чітко визначеної межі, що визначає, які сервери вважаються частиною хмари, а які ні. Щодня і навіть щохвилини, одні сервери можуть виходити в Інтернет, а інші можуть бути відключені для обслуговування, що не робить ніякого впливу на функціональні можливості та продуктивність Інтернету. Ця перевага дійсно надлишкової розподіленої системи - немає єдиної точки управління і, отже, немає єдиної точки відмови.

Хмара може бути підключена до призначеного для користувача або інших мережних пристроїв за допомогою будь-якого з'єднання дротової, Wi-Fi або бездротової лінії зв'язку. Згідно рисунку, хмарний сервер 21A підключається за допомогою дротової або волоконно-оптичної лінії 24 до вишки бездротового зв'язку 25, до точки доступу Wi-Fi 26 або до розподіленого блоку провідної лінії зв'язку 27. Ці канали "останньої милі", в свою чергу, підключаються до будь-якої кількості пристроїв зв'язку або мережних пристроїв. Наприклад, вишка бездротового зв'язку 25 може з'єднуватися стільниковим радіозв'язком 28 зі смартфоном 32, з планшетом 33 або автомобілем з мережним інтерфейсом 31 і може використовуватися для обслуговування мобільних користувачів 40, в тому числі, наприклад, пішоходів, водіїв персональних транспортних засобів, працівників правоохоронних органів та водіїв-професіоналів у галузі вантажоперевезень і доставки. Бездротовий телефонний зв'язок з комутацією пакетів використовує стільникові протоколи 3G, включаючи HSPA і HSDPA, а також 4G/LTE. Стандарт LTE або "довготривалий розвиток" відноситься до мережних стандартів, що забезпечують сумісність з різними стільниковими протоколами, в тому числі, можливість безшовного перемикавання телефонних дзвінків з одного стільника в інший, навіть якщо ці стільники працюють з різними протоколами. Примітка. В контексті цього документа терміном "остання миля" називається канал передачі даних між будь-яким типом клієнтського пристрою, наприклад, планшетом, стаціонарним комп'ютером або мобільним телефоном, і хмарним сервером. Залежно від напрямку передачі даних також іноді використовується термін "перша миля" для вказівки каналу передачі даних між абонентським пристроєм, який ініціює передачу даних і хмарним сервером. У таких випадках канал "остання миля" також є і каналом "перша миля".

При меншій дальності зв'язку точка доступу Wi-Fi 26 з'єднується Wi-Fi радіомодулем 29 зі смартфоном 32, планшетом 33, ноутбуком 35, стаціонарним комп'ютером 36 або мережним пристроєм побутової техніки 34 і може використовуватися в локальних бездротових пристроях в будинках, кафе, ресторанах і офісах. Бренд Wi-Fi поширюється на зв'язок, що працює відповідно до стандартів IEEE для специфікацій 802.11a, 802.11b, 802.11g, 802.11n з однією частотою, а останнім часом для формату 802.11ac з двочастотним діапазоном. Засоби захисту Wi-Fi на основі простого статичного ключа входу в основному використовуються для запобігання несанкціонованому доступу до з'єднання, але не призначені для захисту даних протягом необмеженого часу від прослуховування або злому.

Розподілений блок дротових ліній 27 може підключатися волоконно-оптичним кабелем, коаксіальним кабелем або по каналу Ethernet 30A до ноутбука 35, стаціонарного комп'ютера 36, телефону 37, телевізійного приймача 39 або мідною крученою парою дротів 30B до телефонних ліній, що йдуть до касового терміналу торгової точки 38, що обслуговує стаціонарні або підключені дротовою лінією зв'язку центри торгівлі 42, в тому числі готелі, заводи, офіси, центри обслуговування, банки та будинки. Дротове підключення може включати розділення волоконно-оптичним або коаксіальним кабелем до дому, офісу, заводу або комерційної організації, підключені локально за допомогою модему для перетворення ліній передачі високошвидкісних даних (HSD) в лінії на основі Wi-Fi, Ethernet або мідною крученою пари дротів. У віддалених областях, де немає волоконно-оптичного або коаксіального кабелю, все ще використовується з'єднання по цифровій абонентській лінії (DSL), але з істотним зниженням швидкості передачі даних та надійності з'єднання. В цілому, враховуючи доступ по бездротових, Wi-Fi та провідних ліній зв'язку прогнозують, що до 2020 року кількість об'єктів, підключених до Інтернету в глобальному масштабі, досягне 20 мільярдів.

На відміну від мереж з комутацією каналів, які встановлюють та підтримують безпосереднє з'єднання між приладами, зв'язок з комутацією пакетів використовує адресу для "маршрутизації" пакета через Інтернет до пункту призначення. Таким чином, в мережах зв'язку з комутацією пакетів немає окремої виділеної мережі, яка підтримує з'єднання між пристроями зв'язку, і дані, проходячи через Інтернет, не йдуть по одному та тому ж конкретному шляху. Кожен пакет повинен знайти свій шлях через лабіринт взаємопов'язаних комп'ютерів, щоб потрапити в пункт призначення.

На рисунку 3 показаний гіпотетичний приклад маршрутизації IP-пакета від ноутбука 60 до стаціонарного комп'ютера 61 при використанні мережевого зв'язку з комутацією пакетів. В процесі роботи перший пакет даних, відправлений з ноутбука 60 в маршрутизатор Wi-Fi 62A за допомогою бездротової технології 63A, направляється до ряду DNS-серверів 70, де DNS (Domain Name Server) - сервер доменних імен. Завданням ряду DNS-серверів 70 є перетворення текстового імені або номера телефону пристрою-адресата, в даному випадку стаціонарного комп'ютера 61, в IP-адресу. Перед маршрутизацією пакета кореневий сервер DNS 72 завантажує велику таблицю адрес у вторинний сервер DNS 71. Коли надходить запит від ноутбука 60, вторинний сервер DNS 71 видає у відповідь IP-адресу адресата, тобто стаціонарного комп'ютера 61. У разі, коли вторинний сервер DNS 71 не знає адреси пристрою-адресата, він може запитувати інформацію, якої бракує від кореневого сервера DNS 72. В результаті, IP-адреса передається з ряду DNS-серверів 70 назад на адресу відправника, тобто ноутбука 60.

Після цього ноутбук 60 збирає свої пакети IP-даних та послідовно починає їх відправку адресату, спочатку через Wi-Fi радіомодуль 63A до маршрутизатора Wi-Fi 62A, а потім через мережу маршрутизаторів та серверів, які виступають в якості проміжних маршрутизаторів до адресата. Наприклад, ряд виділених маршрутизаторів, згідно з рисунком, включає пристрої 65A, 65B і 65C, а ряд комп'ютерних серверів, які працюють в якості маршрутизаторів, включає пристрої 66A-66E, які разом утворюють мережу маршрутизаторів, що працює або як для з'єднання з Інтернетом, або як точки присутності, або POP (англ. Point Of Presence), тобто шлюзи з обмеженою зв'язністю, здатні здійснювати доступ до Інтернету. У той час як деякі маршрутизатори або сервери, що діють як точки присутності, підключаються до Інтернету за допомогою лише невеликої кількості сусідніх пристроїв, сервер 66A, згідно з рисунком, з'єднаний з великою кількістю пристроїв та іноді називається "суперточкою присутності". Для більшої ясності слід зауважити, що термін POP (точка присутності) не слід плутати з ім'ям програми POP (англ. Plain Old Postoffice) або звичайним поштовим відділенням, використовуваним в поштових додатках.

Кожен маршрутизатор або сервер, який виступає в ролі маршрутизатора, містить в своїх файлах пам'яті таблицю маршрутизації, що ідентифікує IP-адреси, які він може адресувати, і, можливо, також адреси, які можуть адресувати маршрутизатори над ним. Ці таблиці маршрутизації автоматично завантажуються та встановлюються на кожному маршрутизаторі при першому Інтернет-підключенні та зазвичай не завантажуються в процесі маршрутизації пакета через мережу. Коли IP-пакет входить в маршрутизатор, точку присутності або суперточку присутності, маршрутизатор зчитує достатню частину IP-адреси, як правило, цифри старших розрядів адреси, щоб знати, куди далі направляти пакет на його шляху до адресата. Наприклад, пакет, відправлений в Токію з Нью-Йорка, може проходити спочатку через Чикаго, потім через сервери в Сан-Франциско, Лос-Анджелесі або Сієтлі, а потім вирушати в Токію.

У прикладі, показаному на рисунку 3, пакет, відправлений з ноутбука 60 в маршрутизатор Wi-Fi 62A, потім прямує в маршрутизатор 65A по маршруту 64A; останній, хоча у нього є безліч варіантів, вирішує відправити пакет в суперточку присутності 66A по маршруту 67A. Незважаючи на те, що суперточка присутності 66A також має багато варіантів продовження, вона вирішує, що найкращий шлях в цей конкретний момент - маршрут 68 до сервера-маршрутизатора 66D, відправляє його в локальний маршрутизатор 65C за маршрутом 67B, який, в свою чергу, з'єднується по маршруту 64B з маршрутизатором та точкою доступу Wi-Fi 62B, що зв'язує Wi-Fi радіомодуль 63B зі стаціонарним комп'ютером 61. Таким чином, незважаючи на те, що шлях проходження був прокладений від суперточки присутності 66A до сервера-маршрутизатора 66D і місцевим маршрутизатора 65C, він міг бути точно так же прокладений від суперточки присутності 66A до маршрутизатора 65B і місцевим маршрутизатора 65C, або від суперточки присутності 66A до сервера-маршрутизатора 66D, сервера-маршрутизатора 66E до місцевого маршрутизатора 65C. І оскільки кількість маршрутизаторів, через які проходить пакет, і доступна швидкість передачі даних для кожного із з'єднань між маршрутизаторами розрізняються по інфраструктурі, а також з мережевого трафіку та завантаження, немає способу апіорі визначити, який шлях є найшвидшим або кращим.

На відміну від телефонного зв'язку з комутацією каналів, яка встановлює та підтримує безпосереднє з'єднання між клієнтами, для комутації пакетів даних немає універсального інтелекту, що переглядає Інтернет, щоб вирішити, який шлях є найкращим, оптимальним або найшвидшим для маршрутизації пакета, і немає ніякої гарантії, що два пакета, відправлених послідовно, підуть за одним та тим самим маршрутом. Таким чином, пакет "визначає" свій шлях через Інтернет на основі пріоритетів компаній, що експлуатують маршрутизатори та сервери, через які проходить цей пакет. Кожен маршрутизатор, по суті, містить певні таблиці

маршрутизації і алгоритми маршрутизації, які визначають його кращі маршрути на основі стану мережі. Наприклад, переваги маршрутизатора можуть встановити високий пріоритет для відправки пакетів іншим маршрутизаторам, що належать тій же компанії, врівноважуючи трафік серед підключень до сусідніх маршрутизаторів, знаходячи найкоротшу затримку для наступного маршрутизатора, укладаючи прямі угоди зі стратегічними партнерами по бізнесу або створюючи лінію термінової відправки для VIP-клієнтів за рахунок пропуску якомога більшої кількості проміжних маршрутизаторів. Коли пакет надходить в маршрутизатор, немає способу дізнатися, чи були варіанти маршрутизації обрані конкретною точкою присутності з максимальним урахуванням інтересів відправника або оператора мережевого сервера.

Таким чином, в якомусь сенсі маршрут, по якому проходить пакет, залежить від часу і удачі. У попередньому прикладі маршрутизації від Нью-Йорка до Токіо сам маршрут і результуюча якість обслуговування можуть істотно змінюватися в залежності навіть від невеликих відхилень в дорозі, тобто в нелінійних рівняннях так званого "ефекту метелика". Розглянемо випадок, коли пакет з Нью-Йорка проходить через "маршрутизатор А" в Чикаго, і, через тимчасово високий трафік у Каліфорнії, він буде відправлений швидше в Мехіко, ніж у Каліфорнію. Потім маршрутизатор Мехіко відправляє IP-пакет в Сінгапур, звідки він, нарешті, відправляється в Токіо. Наступний пакет відправляється через "маршрутизатор В" в Чикаго, який при низькому трафіку в цей момент часу направляє пакет в Сан-Франциско, а потім прямо в Токіо всього за два "хмарних стрибка". В такому випадку другий пакет може прибути в Токіо раніше першого, спрямованого по більш довгому шляху. Цей приклад наочно ілюструє проблемне питання використання Інтернету для зв'язку в реальному часі, наприклад, для передачі онлайн-відео або технології VoIP в реальному часі, а саме те, що Інтернет не призначений для забезпечення гарантованого часу передачі або для контролю за затримками при передачі. Затримка може змінюватися від 50 мс до 1 с і більше тільки в залежності від того, через два сервера проходить пакет або через п'ятнадцять.

Відсутність контролю за маршрутизацією в Інтернеті є проблемою для додатків реального часу і, зокрема, є причиною низької якості обслуговування для ОТТ-провайдерів, які намагаються організувати надання VoIP (Voice over Internet Protocol)-телефонії, використовуючи Інтернет як безкоштовну інфраструктуру для своїх сервісів. Оскільки ОТТ-провайтери не контролюють маршрутизацію, вони не можуть контролювати затримку в мережі. Ще одна проблема зв'язку з комутацією пакетів полягає в тому, що можна легко заволодіти даними, не будучи виявленим. Якщо пірат перехоплює пакет та визначає IP-адресу його відправника або одержувача, він може різними методами перехоплювати дані від зламаних маршрутизаторів і або прослуховувати їх, або перенаправляти трафік через свою власну піратську мережу для здійснення шпигунського прослуховування розмови та навіть розшифровки зашифрованих файлів.

Вихідні та цільові IP-адреси відправника і одержувача та інша важлива інформація, яка використовується для маршрутизації пакета (а також використовувана піратами для злову пакета), вказані у вигляді рядка цифрових даних на рисунку 4. IP-пакет містить цифрову інформацію, що визначає фізичне з'єднання між пристроями, спосіб організації даних для зв'язку пристроїв, мережеву маршрутизацію пакета, засоби забезпечення точності передачі корисних даних (корисного навантаження) та тип даних корисного навантаження, а потім самі дані корисного навантаження, призначені для використання різними прикладними програмами.

IP-пакет відправляється та приймається послідовно у вигляді ланцюжка послідовних цифрових бітів, що відображаються по ходу часу 8б зліва направо, і організованих певним чином, який називається Інтернет-протоколом, та встановлений різними комітетами по стандартизації, в тому числі IETF (Internet Engineering Task Force - інженерним радою Інтернету) і іншими. Стандарт гарантує, що будь-який IP-пакет, що відповідає встановленому протоколу, може бути переданий та зрозумілий будь-яким мережним пристроєм, відповідним цьому ж IP-стандарту. Забезпечення зв'язку та спільної роботи мережевих пристроїв і додатків, підключених до Інтернету, є характерною особливістю Інтернету і керівним принципом організації Open Source Initiative (OSI), що не дозволяє будь-якій компанії, уряду або окремій особі взяти під контроль Інтернет або обмежити його доступність або функціональні можливості.

Модель OSI - це абстрактна конструкція, що складається з семи рівнів функціональних можливостей, яка точно визначає формат IP-пакета і кожного сегмента цього пакета. Кожна частина або "сегмент" IP-пакета відповідає даним, що застосовуються до функції конкретного рівня моделі OSI, зведеним в таблицю 87 на рисунку 4. Ці сім рівнів моделі OSI виконують такі ролі:

- Рівень 1- фізичний рівень - містить конкретну інформацію про обладнання, що визначає фізичну природу зв'язку як електричні, радіо і оптичні сигнали, а також спосіб перетворення цих сигналів в біти для використання в системі зв'язку. Завданням рівня PHY (абревіатура від англ. Physical layer - фізичний рівень) є перетворення в потік бітів певних носіїв інформації, таких як Wi-Fi-радіо, Ethernet, послідовні порти, оптичне волокно, стільниковий радіозв'язок 3G або 4G, DSL на мідній кручений парі, USB, Bluetooth, кабельне або супутникове телебачення, цифрове аудіомовлення, відео або мультимедіа контент. В IP-пакеті преамбула 80 представляє дані рівня 1 та використовується для синхронізації всього пакета даних або "кадру" з передавачем та приймаючим обладнанням.

- Рівень 2- канальний рівень, що містить біти, розташовані у вигляді кадрів, який визначає правила та засоби, за допомогою яких потоки бітів, передані з фізичного рівня 1, перетворюються в інтерпретовані дані. Наприклад, потоки бітів, засновані на даних Wi-Fi радіомодуля, можуть відповідати будь-якому заданому стандарту IEEE, в тому числі 802.11a, b, g, n і ac; сигнали радіозв'язку 3G можуть бути модульовані з використанням високошвидкісних методів пакетного доступу HSDPA або HSUPA; модульований світловий сигнал у волоконно-оптичній лінії або електричні сигнали в коаксіальному кабелі можуть бути декодовані в дані відповідно до стандарту DOCSIS 3 і т.д. В IP-пакеті дані рівня 2 охоплюють решту пакету - сегменти 82, 83 і 84, з попереднім "заголовком каналу передачі даних" 81 та завершувачем каналу передачі даних 85, які спільно визначають, коли починається та закінчується охоплюване корисне навантаження, а також забезпечують контроль за відсутністю втрат в процесі передачі. Одним з ключових елементів даних рівня 2 є MAC-адреса або адреса доступу до середовища, що використовується для направлення трафіку даних на конкретні адреси Ethernet, радіолінії або конкретні апаратні приймально-передавальні лінії.

- Рівень 3- мережевий рівень або рівень Інтернету - містить пакети, "датаграми", що містять інформацію інтернет-протоколу (IP), яка використовується для маршрутизації IP-пакета, в тому числі, чи містить пакет дані IPv4 або IPv6 та відповідні IP-адреси відправника та одержувача, а також інформацію про характер корисних даних, що містяться в пакеті, наприклад, використовує даний тип протоколу управління передачею TCP (англ. Transmission Control Protocol – протокол управління передачею), протокол датаграм користувача UDP (англ. User Datagram Protocol) та ін. Рівень 3 також включає функцію запобігання утворенню "безсмертних" пакетів - IP-пакетів, які ніколи не доставляються, а й ніколи не вмирають. Спеціальний тип пакета рівня 3-ICMP (протокол міжмережових керуючих повідомлень) - використовується для діагностики стану мережі, включаючи добре відому функцію "ping". В IP-пакеті рівень 3 містить "IP-заголовок" 82 та включає дані про корисне навантаження, що міститься в сегментах 83 і 84 транспортного та верхнього рівня.

- Рівень 4- транспортний рівень - містить сегменти даних, що визначають характер з'єднання між комунікаційними пристроями, де протокол UDP визначає мінімальний опис корисного навантаження без встановлення з'єднання, а саме, наскільки великий обсяг корисного навантаження, чи є втрачені біти і яка служба додатків (порт) буде використовувати доставлені дані. Протокол UDP розглядається без встановлення з'єднання, оскільки він не підтверджує доставку корисного навантаження, натомість покладаючись на додаток для перевірки наявності помилок або втрати даних. Зазвичай UDP використовується для чутливого до часу зв'язку, наприклад, для широкомовної, мультимовленневої та онлайн-передачі, де повторне відправлення пакета не використовується. Навпаки, TCP забезпечує віртуальне з'єднання, підтверджуючи, що пакет і корисне навантаження достовірно доставлені до відправки наступного пакета та повторно відправляє недоставлені пакети. TCP також перевіряє цілісність даних доставлених пакетів по контрольній сумі та включає засоби повторного складання несвоєчасно доставлених пакетів в їх первинному порядку. Як TCP, так і UDP визначають порти відправлення і отримання, опис служби або програми верхнього рівня, наприклад, веб-сервер або поштовий сервер, пов'язаний з інформацією, що міститься в корисних даних рівня 4. У IP-пакеті рівень 4 містить заголовок TCP/UDP 83 і містить відомості про даних/корисного навантаження 84, включаючи контент, призначений для використання верхніми рівнями 5, 6 і 7 моделі OSI.

- Рівні 5, 6 і 7- верхні або прикладні рівні - описують контент, що доставляється через Інтернет як дані/корисне навантаження 84. Рівень 7 - прикладний рівень - є найвищий рівень в моделі OSI та спирається на шість базових рівнів моделі OSI для підтримки як ПО з відкритим кодом, так та пропріетарного прикладного програмного забезпечення. До часто використовуваних додатків рівня 7 відносяться електронна пошта з використанням SMTP, POP або IMAP; перегляд веб-сторінок з використанням HTTP (Chrome, Safari, Explorer, Firefox); передача файлів з використанням FTP і емуляція терміналу з використанням Telnet.

Пропріетарні програми включають набір продуктів Microsoft Office (Word, Excel, PowerPoint), Adobe Illustrator і Photoshop; додатки для роботи з базами даних Oracle і SAP; фінансове програмне забезпечення Quicken, Microsoft Money і QuickBooks; плюс аудіо- та відеоплеєри (наприклад, iTunes, QuickTime, Real Media Player, Window Media Player, Flash), а також такі програми для читання документів, як Adobe Acrobat Reader і Apple Preview. Додатки рівня 7 також зазвичай використовують вбудовані об'єкти, синтаксично визначаються рівнем 6 - представницьким рівнем - включає текст, графіку та зображення, звук та відео, такі презентації документів, як XML або PDF, а також такі функції захисту, як шифрування. Рівень 5 - сеансовий рівень - встановлює можливість підключення декількох додатків, наприклад, імпорт одного об'єкта в файл іншої програми і управління ініціюванням та завершенням сеансу.

Як описано вище, семирівнева модель OSI визначає функції кожного рівня, а відповідний IP-пакет включає дані, що відносяться до кожного рівня, один всередині іншого за принципом матрьошки ... Зовнішній пакет або фізичний рівень 1 визначає весь IP-кадр і містить інформацію, що відноситься до всіх більш високих рівнів. В рамках цих фізичних даних кадр даних рівня 2 описує канал передачі даних і містить мережеву датаграму рівня 3. Ця датаграма, в свою чергу, визначає рівень Інтернету як корисне навантаження, а сегмент даних рівня 4 описують транспортний рівень. Транспортний рівень передає дані верхнього рівня як корисне навантаження, в тому числі контент рівнів 5, 6 і 7. Це семирівневе вкладення іноді також мнемонічно співвідносять з фразою англійською мовою "all people seem to need data processing" (схоже, що всі люди мають потребу в обробці даних), в якій перші літери слів відповідають першими літерами назв семи рівнів, починаючи з самого верхнього: application (прикладний), presentation (представницький), session (сеансовий), transport (транспортний), network (мережевий), data-link (канальний) і physical (фізичний).

У той час як нижні рівні - фізичний і канальний - пов'язані з обладнанням, середні рівні моделі OSI, що включаються в IP-пакет для опису мережевої та транспортної інформації, абсолютно незалежні від апаратної платформи, що використовується для здійснення зв'язку та доставки IP-пакета. Більш того, верхні рівні, що включаються як корисне навантаження транспортного рівня, специфічні тільки для додатків, для яких вони призначені, та працюють абсолютно незалежно від маршрутизації пакета і його доставки через Інтернет. Таке розділення дозволяє незалежно контролювати кожен рівень, підтримуючи величезну кількість можливих комбінацій технологій і користувачів, які не потребують організаційного схвалення формату пакета або перевірки життєздатності корисного навантаження пакета. Неповні або неправильні IP-пакети просто відкидаються. Таким чином, мережі з комутацією пакетів здатні маршрутизувати, транспортувати та передавати різноманітну інформацію, пов'язану з додатками, в різноманітних комунікаційних середовищах узгодженим чином між будь-якими мережевими пристроями або об'єктами, підключеними до Інтернету.

Підводячи підсумок, слід зазначити, що мережі з комутацією каналів вимагають одного безпосереднього з'єднання між двома або кількома сторонами, що зв'язуються (подібно традиційній телефонній мережі), в той час як в мережах з комутацією пакетів здійснюється поділ документів, звуку, відео та тексту з кількох пакетів, що передаються декількома мережевими шляхами (аналогічно поштовому відділенню, котрі використовують оптимальний варіант для забезпечення точної та своєчасної доставки), а потім виконується складання вихідного контенту з підтвердженням того, що на цьому шляху нічого не втрачено. Порівняння абонентської мережі зв'язку з комутацією каналів та технології VoIP з комутацією пакетів представлено в наступній таблиці:

Мережа	Абонентська мережа зв'язку (PSTN)	Інтернет
Технологія	3 комутацією каналів	3 комутацією пакетів
Підключення	Виділена електрична лінія зв'язку	Кожен пакет маршрутизується через Інтернет
Передача даних	У режимі реального часу (мережа)	Оптимальний варіант (пакет)
Сигнал	Аналоговий або цифровий	Цифровий, IP, технологія VoIP
Контент	Голос	Голос, текст, дані, відео
Швидкість передачі даних	Низька	Висока
Контроль помилок	Відсутня або мінімальний	Глибокий
Вплив пошкодження лінії зв'язку	Не здійснений або перерваний виклик	Зміна маршруту виклику
Вплив відмови електроживлення	Електроживлення забезпечує мережу	Потрібна резервна акумуляторна батарея

Тут слід зазначити, що, незважаючи на те, що абонентська мережа зв'язку (PSTN) працює в режимі реального часу, з'єднання електричної лінії зв'язку використовують методи "оптимального варіанту" доставки пакета і корисного навантаження, на відміну від поштового відділення, що використовує будь-які автомобілі та листонош, щоб в кінцевому підсумку доставити пошту, навіть якщо вона вже запізнилася. Щоб краще зрозуміти метод, за допомогою якого мережі з комутацією пакетів досягають цієї мети, необхідно глибше вивчити функції і роль кожного рівня у семирівневій моделі OSI для мереж.

Рівень 1 моделі OSI - фізичний (PHY) рівень

Фізичний рівень, описуваний рівнем 1 моделі OSI, пов'язаний з роботою устаткування, використовуваного для здійснення зв'язку. Незважаючи на те, що це найголовніший рівень, що описує тільки електричну, радіо і оптичну передачі, він також є дуже різномірним, тому що кожний докладний опис має специфіку, властиву конкретному елементу обладнання. У широкому сенсі апаратні системи зв'язку можуть бути розбиті на два типи: обладнання для зв'язку з високою пропускнуною спроможністю, що використовується для каналів з великим обсягом трафіку, що з'єднують сервери, що утворюють основну мережу Інтернету, наприклад, "хмарні", і обладнання з більш низькою пропускнуною здатністю, яким завершує місцевий зв'язок між пристроями або підключає лінію зв'язку "останньої милі" від хмари до споживачів, підприємств і машин.

Рисунок 5А, Наприклад, ілюструє зв'язок з високою пропускнуною спроможністю між POP-серверами 21А і 21В, підключеними через вишки радіорелейного зв'язку 98, волоконно-оптичні кабелі 91 та супутники радіорелейного зв'язку 93. Радіорелейний зв'язок вимагає, щоб вишки радіорелейного зв'язку 96А і 96В перебували на лінії прямої видимості. Згідно рисунку, ці вишки з'єднані з POP-серверами 21А і 21В за допомогою дротових з'єднань 97А і 97В. Аналогічно, супутниковий зв'язок вимагає наявності радіорелейних висхідних (земля-борт) і низхідних (борт-земля) ліній зв'язку 95А і 95В між супутником 93 та супутниковими антенами 92А і 92В, з'єднаними з POP-серверами 21А і 21В. Як і в попередньому прикладі, сервери 21А і 21В з'єднуються з супутниковими антенами 92А і 92В дротовими з'єднаннями 94А і 94В. Сервери 21А і 21В також можуть з'єднуватися безпосередньо з використанням оптичного з'єднання 90 з високою пропускнуною здатністю, що забезпечується волоконно-оптичними кабелями 91. Незважаючи на те, що наземні та підводні кабелі раніше містили велику кількість багатожильних каналів з мідного дроту, обмежена смуга пропускання та висока вартість міді прискорили глобальний перехід на оптоволокно.

Рисунок 5В ілюструє різні приклади каналу "останньої милі" між хмарою 22, що містить сервери 21В і 21С та з'єднання з високою пропускнуною здатністю 23, та великою кількістю комп'ютерів, телефонів, радіостанцій та інших мережевих пристроїв. Згідно рисунку, дротяні з'єднання можуть містити волоконно-оптичний кабель 91 і коаксіальний кабель 105, та в меншій мірі виту пару мідних дротів. Бездротові з'єднання можуть здійснюватися за допомогою ряду засобів, включаючи вежу стільникового радіозв'язку 18, вишку двобічного радіозв'язку 15, точку доступу Wi-Fi 26 та супутник 93.

Як один із прикладів сервер 21С, який діє у ролі хмарного шлюзу, з'єднується волоконно-оптичною лінією зв'язку 24 з базовою станцією LTE 17, керуючою радіовишкою 18 для здійснення стільникового зв'язку 28 зі стільниковим телефоном 32, планшетом 33 або ноутбуком 35. Сервер 21С також підключається до загальнодоступного маршрутизатора Wi-Fi 100, пов'язаного каналом Wi-Fi 29 зі стільниковим телефоном 32, планшетом 33 або ноутбуком 35.

Сервер 21С підключається до передавальної системи кабельного модему CMTS 101, яка, в свою чергу, з'єднується коаксіальним кабелем 105 з телевізійною приставкою (TV STB) 102, що управляє телевізійним приймачем 39 з використанням HDMI (High Definition Multimedia Interface - мультимедійного інтерфейсу високої чіткості) 107, та з кабельним модемом 103. Кабельний модем 103 генерує два різних типи вихідних сигналів - голосовий та високошвидкісний цифровий (HSD). Голосовий вихідний сигнал може використовуватися з портативними телефонами 5, а HSD управляє стаціонарним комп'ютером 36, а також планшетом 33, побутовим приладом 34 та стільниковим телефоном (не показаний) за допомогою Wi-Fi сигналу 29, що генерується домашньою точкою доступу Wi-Fi 26. Кабельний модем 103 може в деяких випадках формувати HSD сигнал як Ethernet 104, підключений до стаціонарного комп'ютера 36. З іншого боку, телевізійна приставка TV STB 102 може приймати свої сигнали по супутниковому каналу зв'язку 95, що містить супутникові антени 92А і 92В зі супутником 93. Разом з TV STB 102 різні виходи кабельного модему 103 створюють домашню мережу зв'язку 100.

Сервер 21С також може підключатися до службових пристроїв зв'язку за допомогою сигналів радіостанції повнодуплексного зв'язку 20, керуючих радіостанціями 16А і 16В від базової станції 14 TETRA або EDACS і радіовежі 15, або через міні-АТС 8, керуючу стаціонарними телефонами 9. Оскільки більшість систем повнодуплексного радіозв'язку і міні-АТС не підтримують методи зв'язку з комутацією пакетів і не використовують загальнодоступні телефонні номери для маршрутизації викликів, інформація втрачається всякий раз, коли дані передаються між сервером 21С і міні-АТС 8 або базовою радіостанцією 14. Те ж саме відноситься і до мосту абонентської мережі зв'язку 3, підключеному до POTS (традиційної телефонної мережі) 6, оскільки POTS не призначена для одночасної обробки голосового сигналу та сигналу даних.

Роль фізичного (PHY) рівня змінюється в системах в залежності від режиму зв'язку - "один-до-одного", "один-до-багатьох" або "багато-до-багатьох". У режимі зв'язку "один-до-одного", принцип дії якого ілюструється на рисунку 6А, два та тільки два електронних пристрої 140А і 140В безпосередньо зв'язуються один з одним з використанням виділеного електричного, оптичного або радіоз'єднання для реалізації з'єднання "один-до-одного". Використовуючи запропонований та зумовлений протокол зв'язку, встановлений в інтерфейсних модулях 143А і 143В, між пристроями для здійснення зв'язку може бути встановлений тільки апаратний інтерфейс. Більш конкретно, дані, що генеруються електронною схемою 141А, передаються в інтерфейсний модуль зв'язку на фізичному рівні 143А, з'єднаний за допомогою електричних, радіо або оптичних сигналів 144 з інтерфейсним модулем зв'язку на фізичному рівні 143В ідентичної конструкції. Отримані дані обробляються електронною схемою 141В, та в деяких випадках повертають відповідний сигнал інтерфейсному модулю 143А в пристрої 140А.

Оскільки в режимі зв'язку "один до одного" є тільки два пристрої, немає необхідності включати програмне забезпечення, щоб направляти трафік, ідентифікувати пристрої або вирішувати, які пристрої реагують на інструкції. Приклади такого виділеного зв'язку "один до одного" включають в себе шини послідовного обміну, наприклад, RS232, спочатку використовувалися для підключення принтерів до стаціонарних комп'ютерів, і шина простого послідовного управління (S2C) (патент США №7 921 320), що використовувалась для управління яскравістю світлодіодного підсвічування на дисплеях стільникових телефонів.

Виділений зв'язок "один до одного" має низку переваг. По-перше, її легко реалізувати та при бажанні можна повністю виконати на апаратних засобах, навіть в межах однієї інтегральної схеми, без ядра центрального процесора (ЦП). З іншого боку, цей інтерфейс може бути реалізований мікропрограмно, тобто у вигляді програми спеціального обладнання, що вимагає мінімальної потужності обробки ЦП для виконання обмеженого набору команд управління обміном даними. По-друге, немає необхідності управляти трафіком - такі інтерфейси можуть працювати з дуже високою швидкістю передачі даних. Нарешті, вона має низку переваг з точки зору безпеки, оскільки жодний інший пристрій не користується цією лінією і не може "прослуховувати" інформаційний обмін, який по ній відбувається. У цьому випадку даний інтерфейс може бути реалізований для "перевірки ідентичності" або "автентифікації" будь-якого пристрою в момент підключення пристрою до порту, а також для відключення порту, якщо з'єднання переривається навіть на мить. Пристрої, що не автентифіковано, ігноруються, та порт залишається відключеним доти, поки пристрій, який підтвердив ідентичність, не замінить проблемний пристрій.

Зв'язком між двома пристроями в режимі "один-до-одного" можна управляти двома принципово різними способами. В "тимчасовій" системі зв'язку кожен пристрій має рівні повноваження щодо прийняття рішень, та пріоритет управління обміном інформацією зазвичай встановлюється за принципом "першим прийшов - першим обслужений". В якості альтернативи, в конфігурації "ведучий-ведений" ведучий пристрій бере на себе управління процесом прийняття рішень, а ведений пристрій має виконувати запити і отримувати дозвіл від ведучого пристрою для здійснення будь-яких дій.

Тільки фізичний інтерфейс "один-до-багатьох" показаний на рисунку 6В, де три пристрої 140А, 140В і 140С або більша кількість пристроїв з'єднані загальною лінією зв'язку, показаною як "шина" даних 144. Кожен пристрій включає в себе електронну схему 141А, 141В або 141С, з'єднану відповідною лінією даних 142А, 142В або 142С з фізичним інтерфейсом 143А, 143В або 143С. У цій конфігурації дані, що передаються з будь-якого пристрою, надходять на всі інші пристрої, підключені до шини або комунікаційного середовища. Наприклад, якщо пристрій 140С відправляє дані на шину 144, обидва пристрої 140А і 140В прийматимуть повідомлення, якщо пристрій 140В відправляє дані на шину 144, повідомлення будуть приймати пристрої 140А і 140С і т.д. Зв'язок, в якому всі слухають, називається "широкомовним", він аналогічний мовленнєвим телевізійним станціям, що передають контент до багатьох телевізійних приймачів.

На сучасному професійному жаргоні широкомовна передача "один-до-багатьох" називається мультимовлення. Тільки на фізичному рівні 1 мовлення за своєю суттю не є захищеною формою зв'язку, тому пристрій, що здійснює його, не знає, хто його слухає. Під час Другої світової війни мовлення використовувалося для передачі інформації військам, флотам та підводним човнам по незахищених каналах з використанням "шифрування", призначеного для позбавлення слухача можливості інтерпретувати повідомлення, за рахунок використання секретного алгоритму для скремблювання інформації. Якщо несанкціонований слухач був здатний "розшифрувати код", це серйозно підривало безпеку не тільки тому, що порушник міг перехоплювати конфіденційні повідомлення, але й тому, що той, хто передавав інформацію не знав, що це можливо. Таким чином, при реалізації тільки на фізичному рівні 1, зв'язок "один-до-багатьох" має кілька основних недоліків, а саме:

- будь-який пристрій, здатний підключатися до комунікаційної шини або середовища, може приймати або контролювати контент повідомлення, навіть якщо воно є ненавмисним одержувачем або загрозою безпеки;

- пристрій, що ініціює передачу даних, не має уявлення про те, що слухають інші пристрої;
- пристрій, що ініціює передачу даних, не може підтвердити правильність та точність прийому відправлених даних;

- передача комунікаційного трафіку випадковим або незацікавленим одержувачам марно витрачає значну частину пропускну здатності каналу зв'язку, змушуючи одержувачів приймати повідомлення, які вони не хочуть отримувати, не потребують їх і не цікавляться ними.

Проблема підключення декількох пристроїв з реалізацією тільки на фізичному рівні ще більш ускладнюється в конфігурації "один-до-багатьох" і особливо в конфігурації "багато-до-багатьох" через боротьбу за пропускну здатність каналу та при визначенні пріоритету пристроїв, яким дозволена передача. Щоб запобігти конфліктам даних у випадках, коли кілька пристроїв намагаються здійснювати мовлення одночасно, зв'язок тільки на фізичному рівні повинен приймати певну ієрархічну форму з присвоєнням кожному пристрою пріоритетних прав на використання каналу або систем зв'язку. При розробці центрального процесора (ЦП) передбачається кілька методів управління зв'язком усередині ЦП і між ЦП та пам'яттю. Ці концепції включають поняття "шини адреси", використовуваної для визначення того, з яким пристроєм або областю пам'яті ЦП намагається встановити зв'язок; "Шини даних", що використовується для перенесення даних окремо від адреси, і одна або кілька ліній "переривання", що використовуються для визначення того, коли повинна виконуватися будь-яка задача.

Таким чином, ЦП може динамічно реагувати на необхідні завдання, що дозволяє ЦП встановлювати зв'язок та здійснювати роботу з декількома периферійними пристроями в міру необхідності, знімаючи з ЦП відповідальність за постійне опитування або звернення за інформацією про статус від підключених периферійних пристроїв. Якщо під час роботи периферійних пристроїв потрібна увага, воно генерує сигнал "переривання", тобто запит на обслуговування шляхом короткочасного електричного замикання загальної мережі (лінії переривання) на землю. Після генерування переривання периферійний пристрій очікує, що процесор запитає у нього, що йому потрібно, таким же чином, як система "виклику бортпроводника" в літаку. Оскільки процедура обслуговування переривань зазвичай дозволяє ЦП завершити роботу над тим, що він в даний момент робить, перш ніж обслуговувати перериваючий пристрій, такий метод не підходить для обробки в реальному часі пріоритетних подій, що вимагають негайного реагування.

Щоб розширити можливості обміну на основі переривань для додатків реального часу, в архітектурі ЦП вводиться поняття концепції пріоритетної лінії, званої "немасковане переривання", яка дозволяє примусити ЦП скасувати все, що він робить, і негайно обслужити високопріоритетні події або події реального часу, наприклад, повідомлення, яке надходить в маршрутизатор або виклик, що надходить в стільниковий телефон. Подібно VIP-обслуговуванню невеликої кількості пасажирів в салоні першого класу, такі методи роботи передбачені для обмеженої кількості пристроїв, підключених до центрального комунікаційного або ведучого пристрою, але цей підхід не поширюється на обслуговування великого числа користувачів і не підтримує розподілені системи тимчасового зв'язку без централізованого управління.

За прийнятим в ЦП принципом адресації пристроїв рівня 2, 3 і 4 моделі OSI аналогічним чином використовують "ідентифікатор" пристрою в якості ключового компонента для направлення трафіку інформаційного обміну між пристроями. Наприклад, рівень 2 - каналний рівень - ідентифікує вхідні та вихідні з'єднання з використанням доступу до середовища передачі даних або MAC-адресу; рівень 3 - мережевий рівень - маршрутизує пакети через

мережу з використанням IP-адрес; а рівень 4 - транспортний рівень - використовує адреси портів для визначення типу даних, що переносяться, наприклад, електронної пошти, веб-сторінок, файлів і т.д. В ЦП шина адреси, шини даних та лінії переривання виконані у вигляді окремих ліній, також званих "паралельних" портом. Незважаючи на те, що паралельні порти ефективні з точки зору підвищення швидкості передачі даних для ліній зв'язку всередині однієї мікросхеми або для високошвидкісних з'єднань на короткій відстані на материнській платі комп'ютера, для зв'язку на більш далекій відстані велику кількість ліній є дорогим і недоцільним.

Замість цього найпоширенішим методом електронного зв'язку сьогодні є послідовний зв'язок з доставкою інформації в пакетах, переданих протягом більш тривалого часу. Пакет IP, показаний раніше на рисунку 4, містить всі необхідні дані маршрутизації та зв'язку для доставки контенту - корисного навантаження 84 - від відправника до одержувача через мережу зв'язку, локально або глобально. Кожен IP-пакет містить необхідні адреси, в тому числі інформацію канального рівня в заголовку каналу передачі даних 81, інформацію про IP-адресу в заголовку 82 IP і інформацію про адресу порту в заголовку TCP/UDP 83, при цьому вони організовані послідовно та приймаються в певному порядку протягом деякого часу 86, а не відправляються одночасно та паралельно.

Рівень 2 моделі OSI - канальний рівень

Щоб подолати вищезгадані проблеми в управлінні потоком інформації при зв'язку з декількома пристроями тільки на фізичному рівні, семирівнева модель OSI включає в себе абстрактну конструкцію рівня 2 або "канальний" рівень. По суті, канальний рівень виконує обов'язки координатора трафіку, направляє потік даних та вирішує, які дані на загальній шині даних або в спільно використовуваному середовищі призначені для конкретного пристрою. Роль рівня 2 - канального рівня - ілюструється на прикладі, наведеному на рисунку 7A, де пристрої 145A, 145B і 145C спільно використовують загальне з'єднання або "шину" 144, але кожне з них на канальному рівні має свій власний інтерфейс зв'язку 146A, 146B і 146C, що підтримує тільки один канал зв'язку 147 в конкретний момент часу. Тому, незважаючи на те, що багато пристроїв з'єднані один з одним на фізичному рівні, тобто спільно використовують загальну апаратну шину, на канальному рівні тільки два з них підключені один до одного в конкретний момент часу. Зокрема, якщо пристрій 145A бажає встановити зв'язок виключно з пристроєм 145B, то канал 147 утворюється тільки між пристроєм A та пристроєм B, навіть якщо пристрій C підключено до них обох на фізичному рівні.

Впровадивши пов'язане з рівнем 2 обладнання або програмне забезпечення в якості інтерфейсу канального рівня у всіх трьох пристроях, тобто канальні інтерфейси 146A, 146B і 146C, дані, що відправляються по шині даних 144, можна перевіряти та фільтрувати для обмеження обсягу інформаційного обміну між пристроями відправника та передбачуваного одержувача. Решта мережевих пристроїв, що підключені до шини, незважаючи на те, що вони як і раніше приймають ті ж дані, ігнорують їх і не вживають ніяких дій в результаті прийому вхідного повідомлення. Такий протокол використовується послідовним периферійним інтерфейсом (шиною SPI), де кілька пристроїв підключені до загальної "шини даних", ця шина передає дані, але реагує тільки тоді, коли певна адреса надходить по лінії адреси. Таким чином, шина SPI використовується для управління світлодіодами в системі підсвічування ЖК телевізорів, що дозволяє незалежно управляти кожним рядком світлодіодів на екрані телевізора, щоб полегшити регулювання яскравості та "місцеве затемнення" для відеоконтенту HD і UHD з високою контрастністю. Ця ж концепція також використовується в архітектурі шин комп'ютерної пам'яті для вибору банку пам'яті при читанні або запису в слотах розширення PCI Express на комп'ютерах та в шині CAN, використовуваної в автомобілях.

Аналогічно, концепція канального рівня використовується в системі бездротового зв'язку Bluetooth для бездротових навушників, гучномовців, відеокамер і т.д., де тільки раніше авторизовані або "пов'язані" парні пристрої можуть встановлювати зв'язок один з одним. У протоколі Bluetooth процес з'єднання, дії по організації каналу передачі даних, відбуваються заздалегідь і незалежно фактичної передачі даних. Як тільки зв'язок буде встановлений, два пов'язаних пристрої можуть, принаймні теоретично, здійснювати інформаційний обмін без перешкод з боку інших переговорів в системі Bluetooth, які відбуваються одночасно між іншими сторонами. Насправді, шина інформаційного обміну Bluetooth 144 являє собою загальний радіоканал з обмеженою пропускнуною спроможністю та інформаційною ємністю. За визначенням комітету стандартів Bluetooth та за обоюсторонньою згодою Федеральної комісії зі зв'язку США (FCC) та відповідних агентств інших держав, кожний сумісний з Bluetooth пристрій здійснює трансляцію передачу в одному та тому ж загальному радіодіапазоні або "каналі". Кожен з сеансів мовлення, які відбуваються одночасно, займає частину доступної смуги пропускання каналу і швидкості передачі даних. Незважаючи на перекриття сеансів передачі, дані не

конфліктують досить довго, поки канал не стає перенаселеним. Щоб звести до мінімуму ризик виникнення конфліктів між даними та обійти проблеми перенаселення і доступності каналів, з'єднання Bluetooth навмисно обмежується дуже малою дальністю і надзвичайно низькою швидкістю передачі даних.

В раніше описаній архітектурі шини фізичне з'єднання являє собою загальну лінію, електричне з'єднання або середу, підключену безпосередньо або спільно використовувану кількома пристроями. В архітектурі шини будь-який підключений до шини пристрій, споживає деяку енергію від шини, щоб здійснювати зв'язок, і, хоча та трохи, але знижує якість роботи шини. Це явище, поступово знижує якість роботи шини при кожному підключенні додаткового пристрою, називається "вантаженням". У разі якщо навантаження занадто велике, шина не в змозі працювати в межах заданих показників якості, та зв'язок або перестає працювати, стаючи занадто повільним, або працює з великою кількістю помилок. Максимальна кількість пристроїв, при підключенні яких до лінії або шини вона ще працює та забезпечує задані показники якості, називається "здатністю навантаження" шини або з'єднання. Щоб пом'якшити небезпеку вантаження, шина може бути розбита на кілька сегментів, кожен з яких працює за принципом "один-до-одного", де поліпшується цілісність сигналу, або здійснюється його буферизація по потужності перед відправкою інших пристроїв. З точки зору підключення, передані дані або сигнал, канал передачі даних залишаються такими ж, як і в шинній архітектурі, але потужність електричного, оптичного або радіосигналу, фізичні дані постійно підтримуються на потрібному рівні незалежно від кількості мережевих пристроїв.

Прикладом таким чином підключеної мережі, яка містить сполуки "один-до-одного" з посиленними сигналами, є архітектура концентратора, показана на рисунку 7В, де пристрої А, В і С, показані в простій формі стеками зв'язку 146А, 146В і 146С, відповідно, підключаються один до одного через шину посилення сигналу або "концентратор" 148. Концентратор точно відтворює сигнал вхідного контенту без зміни, фільтрації або інтерпретації потоку даних, а потім виводить посилену версію того ж сигналу на лінії, підключені до інших пристроїв.

Кожен пристрій з'єднується з концентратором 148 через свою власну виділену лінію зв'язку, а саме, 151А, 151В і 151С, поєднуючи стек зв'язку периферійного пристрою 146А зі стеком зв'язку концентратора 150А, стек зв'язку пристрою 146В зі стеком зв'язку концентратора 150В та стек зв'язку пристрою 146С зі стеком зв'язку концентратора 150С, відповідно. У свою чергу, стеки зв'язку всередині концентратора 148 підключаються до високошвидкісної внутрішньої шини 149 для мережевих пристроїв, підключених до концентратора. Незважаючи на те, що всі дані фізичного рівня проходять через концентратор 148 та внутрішню шину 149, зв'язок 147 рівня 2 - каналного рівня - працює так, як ніби тільки стек зв'язку 146А в пристрої А зв'язується виключно з стеком зв'язку 146В в пристрої В, але не з пристроєм С. Проте, дані фізичного рівня доставляються до кожного апарата, який підключається до концентратора, та з однаковими затримками поширення. Крім того, оскільки немає способу дізнатися, який пристрій здійснює мовлення, а які пристрої слухають, концентратор повинен підтримувати зв'язок у багатьох напрямках. Таким чином працюють концентратори для Ethernet і Thunderbolt. В інших концентраторах, наприклад, для "універсальної послідовної шини" (USB), концентратор має один вхід і кілька виходів, як правило, від двох до шести, що використовують роз'єми USB різної форми, щоб розрізнити ці два типи і напрямок потоку даних за замовчуванням.

Іншим способом з'єднання пристроїв, що забезпечує посилення сигналу, є архітектура "послідовної лінії зв'язку" (daisy chain), показана на рисунку 7С, де пристрої А, В і С з'єднані послідовно: стек зв'язку 152А пристрої А підключений до стека зв'язку 152В пристрою В за допомогою фізичного з'єднання шини 151А, стек зв'язку 152В пристрою В підключений до стека зв'язку 152С пристрої С за допомогою фізичного з'єднання шини 151В, а стек зв'язку 152С пристрої С з'єднаний за допомогою фізичного з'єднання шини 152С з наступним пристроєм, включеним в цю послідовну лінію зв'язку, якщо така є. Щоб прояснити той факт, що фізичне з'єднання і, власне кажучи, сам механічний з'єднувач в провідних системах, відрізняються один від одного, кожен стек зв'язку 152А, 152В і 152С містить два інтерфейси фізичного рівня 1, та тільки один каналного рівня 2.

При роботі послідовної лінії зв'язку потік фізичних даних надходить з каналного рівня стека зв'язку 152А в його фізичний інтерфейс, а потім по кабелю, що утворює фізичне шинне з'єднання 151А, у фізичний інтерфейс стека зв'язку 152В, вгору на його каналний рівень, вниз у другий фізичний інтерфейс пристрою В, по кабелю, що утворює фізичне шинне з'єднання 151В, у фізичний інтерфейс стека зв'язку 152С та вгору на його каналний рівень. Таким чином, поки фізичний сигнал блукає по всіх трьох показаних пристроях, каналний рівень з'єднує тільки стек зв'язку 152А пристрою А зі стеком зв'язку 152С пристрою С, а пристрій В ігнорує дані, які через нього проходять. Приклади мережевого зв'язку, заснованого на архітектурі послідовної лінії

зв'язку, включають шину FireWire, яка визначається стандартом IEEE 1394, цифровий інтерфейс музичних інструментів (MIDI), та вже застарілу технологію Token Ring, що використовувалася першими персональними комп'ютерами на базі Windows. Позитивною властивістю пристроїв послідовної лінії зв'язку є те, що не потрібно використовувати додатковий пристрій, тобто концентратор, та всі мережеві електричні з'єднання для підключення до нього. Однією з негативних властивостей архітектури послідовної лінії зв'язку є те, що затримка поширення між пристроями збільшується з підключенням кожного пристрою, через яке проходять дані, що призводить до неприпустимо низької продуктивності, особливо в швидкодіючих додатках реального часу.

У всіх трьох прикладах архітектури шини, архітектури концентратора та архітектури послідовної лінії зв'язку дані фізичного рівня відправляються кожному мережевому пристрою, навіть якщо воно не є ймовірним одержувачем. Пристрій самостійно виконує ідентифікацію і фільтрацію пакетів, порівнюючи адресу прийнятих даних з власною адресою, зазвичай заздалегідь запрограмованим як фіксована постійна адреса з використанням незалежної пам'яті, механічних мікроперемикачів або перемичок в пристрої або в одній з його мікросхем. Коли конкретний пристрій розпізнає пакет даних з адресою одержувача, який відповідає його адресі, він відповідає, в іншому випадку він взагалі ігнорує пакет. Адреса пристрою в пакеті повинна відповідати використовуваному протоколу зв'язку, будь то MIDI, USB, IEEE 1394, Thunderbolt і т.д. У разі, коли пакет використовує інтернет-протокол в якості канального рівня, адресі присвоюється певне ім'я, зване "media access" (доступ до середовища) або MAC-адрес, який буде розглядатися нижче в цьому описі.

Один з ключових атрибутів архітектури шини, концентратора та послідовної лінії зв'язку полягає в тому, що дані, які транслиються на фізичному рівні, тобто електричні, радіо або оптичні сигнали, відправляються кожному мережевому пристрою. Цей метод витрачає значну частину пропускної здатності мережі на непотрібну відправку пакетів пристроїв, які їх не потребують і для яких вони не призначені. Оскільки Ethernet став переважним стандартом для локальної обчислювальної мережі або LAN-підключення, ці втрати пропускної здатності мережі були ідентифіковані та в кінцевому підсумку усунені шляхом введення мережевого "комутатора".

У реалізаціях LAN, подібних показаної в прикладі з трьома пристроями на рисунку 8А, комутатор LAN 159 вставлений посеред комунікаційного фізичного рівня інтерфейсів зв'язку 146А, 146В і 146С, що містяться в пристроях 145А, 145В і 145С. На відміну від шинного з'єднання, показаного раніше на рисунку 7А, з однією спільною шиною даних 144, що з'єднує пристрої, додавання комутатора LAN 159 ділить цю шину на три окремих з'єднання "один-до-одного", а саме: фізичне з'єднання 148А між пристроєм 145А і комутатором 159, фізичне з'єднання 148В між пристроєм 145В і комутатором 159, фізичне з'єднання 148С між пристроєм 145С і комутатором 159 і т.д. Відповідно до рисунку, кожне фізичне з'єднання здійснюється тільки між двома пристроями "один-до-одного", при цьому проміжні пристрої відповідають за передачу потоку послідовних даних до його сусідніх мережевих пристроїв.

Цей принцип може поширюватися на будь-яку кількість пристроїв, при цьому робота комутатора LAN 159 може бути односпрямованою або двобічною, напівдуплексною або повнодуплексною. В процесі роботи, щоб встановити канал передачі даних 147 виключно між інтерфейсами зв'язку 146А і 146В підключених до мережі пристроїв 145А і 145В, комутатор 159 LAN встановлює з'єднання фізичного рівня тільки між двома комунікаційними мережевими пристроями 145А і 145В. Таким чином, на фізичному рівні з'єднання встановлюється виключно між двома комунікаційними пристроями, а саме, між пристроєм 145А та пристроєм 145В, але без підключення будь-яких інших мережевих пристроїв, наприклад, пристрою 145С. Однією з переваг використання комутатора LAN 159 є те, що пристрій 145С не займається прослуховуванням непотрібної інформації, наявної в мережі, і його інтерфейс зв'язку 146С залишається вільним до тих пір, поки він не буде викликаний.

Другою перевагою використання комутатора LAN 159 є те, що сигнал, що надходить в комутатор 159 LAN, посилюється перед відправкою сусіднім мережевим пристроєм, тому при збільшенні кількості пристроїв, що підключаються до комутатора LAN 159, не відбувається вантаження, ослаблення сигналу або впливу на швидкість передачі даних. Таким чином, здатність навантаження комутатора LAN 159 по суті не обмежена та визначається тільки кількістю підключень в комутаторі LAN.

Схематичне зображення комутатора LAN 159 ілюструється на рисунку 8В, на якому показані лінії 160А-160F. У точці перетину в кожній комбінації двох ліній знаходиться вузол LAN 161, що представляє собою двонаправлений перемикач та підсилювач. Наприклад, вузол АВ з'єднує лінію В 160В з лінією А 160А, вузол ВЕ з'єднує лінію В 160В з лінією Е 160Е, вузол СЕ з'єднує

лінію С 160С з лінією Е 160Е та так далі. При нормальному зв'язку кожна лінія з'єднується не більше ніж з однією іншою лінією для створення взаємопов'язаної пари. Відразу після розміщення пристрою таблиця маршрутизації MAC-адрес рівня 2 (не показана) зберігається з комутатором LAN, щоб відстежувати, які пристрої підключені і до якого роз'єму. Ця таблиця по суті зіставляє MAC-адреси з їх фізичним підключенням до комутатора LAN, встановлюючи точний зв'язок між рівнем 2 - канальним рівнем - і рівнем 1 - фізичним рівнем. Таблиця є динамічною, тому, якщо один пристрій відключається від мережі, а інший підключається до неї, таблиця маршрутизації MAC-адрес автоматично оновлюється в комутаторі LAN 159.

В особливих випадках, коли здійснюється ширококомовна передача даних кожному пристрою в мережі, наприклад, при запуску, коли один пристрій може шукати інший, але не ідентифікувати його місце розташування на комутаторі LAN, тоді кожен пристрій може бути з'єднаний одночасно тільки з одним джерелом, що передає дані, і іншими пристроями, які приймають їх. Завдяки вбудованим підсилювачам, навіть в ширококомовному режимі кожен сигнал буферизується, тому зниження швидкодії або спотворення сигналу не відбувається.

Третя і найважливіша перевага використання комутатора LAN 159 є такою, що він значно збільшує пропускну здатність всієї мережі, дозволяючи одночасно і незалежно здійснювати кілька розмов між парами пристроїв, згідно з рисунком 8С. У цьому прикладі пристрої 145А, 145В, 145С і 145F з'єднані з комутатором LAN 159 фізичними лініями 160А, 160В, 160С і 160F відповідно. Через канальний рівень 2 пристрої 160А і 160В встановлюють виділений канал зв'язку АВ 164, в той час як пристрої 160С і 160F одночасно встановлюють виділений канал зв'язку CF за допомогою з'єднання 165. При передачі інформації від пристрою 145А до пристрою 145В дані відправляються по лінії 160А через "включений" вузол LAN 162 та по лінії 160В до пристрою 145В. Одночасно при передачі інформації від пристрою 145С до пристрою 145F дані відправляються по лінії 160С через вузол LAN 163 та по лінії 160F пристрою 145F. Всі інші вузли LAN залишаються вимкненими, навіть якщо пристрої підключені до інших ліній.

Таким чином, два незалежні канали зв'язку або "розмови" можуть здійснюватися при максимальній швидкості передачі даних в з'єднанні АВ 164 та з'єднанні CF 165, не чекаючи спільної роботи на загальній шині даних. Таким чином, у наведеному прикладі пропускну здатність мережі, що з'єднує чотири пристрої, подвоюється при використанні комутатора LAN 159 і архітектури LAN, в порівнянні з використанням шинної архітектури мережі, концентраторів або послідовної лінії зв'язку. У комутаторі LAN з "n" лініями та з'єднаннями максимальне число одночасних розмов становить "n/2", в той час як альтернативні мережі, що використовують послідовні з'єднання, можуть в конкретний момент часу підтримувати тільки одну розмову.

Слід зазначити, що коли з'єднуються між собою два пристрої, наприклад, пристрої 145А і 145В з з'єднанням АВ 164, зв'язок з використанням однієї лінії є тільки напівдуплексний, оскільки тільки один пристрій може "говорити" в конкретний момент часу, в той час як інший слухає. Якщо потрібно повнодуплексний зв'язок, кількість ліній та з'єднань вузла в комутаторі LAN 159 має бути подвоєно, причому вихід пристрою 145А повинен бути підключений до входу пристрою 145В, і одночасно вихід пристрою 145В повинен бути підключений до входу пристрою 145А. Таким чином, для повнодуплексної розмови пристрою А з пристроєм В одночасно повинні бути задіяні два з'єднання - з'єднання АВ, за яким пристрій А відправляє дані пристрою В, та з'єднання ВА, за яким пристрій В відправляє дані пристрою А, при цьому кожне з них має перебувати на різних лініях та проходити через унікальні вузлові з'єднання.

Незважаючи на те, що ілюстрації, наведені на рисунку 8С, можуть мати на увазі, що лінії 160А-160F є дрти та вилки електричного роз'єму, такий опис буде також справедливий і для випадку, коли якщо ці лінії є лініями радіозв'язку або оптичного зв'язку. Для випадку радіозв'язку кожна лінія може, наприклад, являти собою унікальну смугу частот або "підканал", який використовується для передачі даних однієї лінії, і де 20 радіочастот, діапазонів або підканалів можуть використовуватися для одночасної і незалежної передачі до 10 різних розмов. Для випадку оптичного зв'язку, кожна лінія може мати різну довжину хвилі світла або унікальну схему модуляції. Радіо або оптичний інтерфейс перетворює електромагнітний зв'язок назад в електричні сигнали в пристроях зв'язку. Таким чином, комутатор LAN може використовуватися для підвищення пропускну здатності будь-якої налаштованої мережі системи зв'язку.

Незважаючи на появу великої кількості протоколів та стандартів для направлення трафіку та передачі даних в мережах з комутацією пакетів, з'явилося кілька широко поширених стандартів, які вимагають більш докладного пояснення. Нижче розглядаються широко застосовані або розроблені на основі існуючих застарілих стандартів протоколи зв'язку та пов'язане з ними обладнання, а саме:

- Ethernet (IEEE 802.3) для мереж зв'язку на основі електричних з'єднань;
- Wi-Fi (802.11) для мереж радіозв'язку ближньої дії;

- 4G/LTE для мереж радіозв'язку дальньої дії;
- DOCSIS 3.0 для мереж зв'язку на основі коаксіальних та волоконно-оптичних кабелів.

Ethernet (IEEE 802.3) - У сучасних мережах з використанням електричних з'єднань для створення LAN більшість пропрієтарних мереж замінено мережами на основі загально визнаного стандарту IEEE 802.3, званого Ethernet. Специфікація Ethernet встановлює вимоги до пакету даних, що використовується каналним рівнем 2, а також визначає електричні з'єднання, напруги, швидкість передачі даних, швидкості зв'язку і навіть фізичні з'єднувачі - вилки і розетки. Тому Ethernet як стандарт являє собою специфікацію як для каналного рівня 2, так і для фізичного рівня 1. Специфікація контенту пакета даних Ethernet - пакета Ethernet рівня 1 188 та пакета Ethernet рівня 2 189 - графічно показана на рисунку 9 як послідовні дані, представлені зліва направо в напрямку ходу часу 86. Відповідна таблиця 190 описує функцію кожного блоку або субпакета в пакеті Ethernet.

Пакет Ethernet рівня 2 189, згідно з рисунком, містить MAC-адресу одержувача 182, MAC-адресу відправника 183, додатковий блок віртуальної LAN 184, блок Ethertype 185, контроль кадру 186 і корисне навантаження 187, що представляють фактичні дані, що передаються пакетом Ethernet. Щоб забезпечити виконання вимог до швидкості, розмір пакета Ethernet-рівня 2 може, відповідно до специфікації Ethernet, змінюватися від 64 до 1518 байт для перенесення корисного навантаження від 42 до 1500 байт. У разі, коли в пакет включено додатковий блок VLAN 184, довжина пакета збільшується на 4 байта, а максимальна довжина пакету Ethernet рівня 2 становить 1522 байта.

Пакет Ethernet рівня 1 188 об'єднує весь вміст пакета Ethernet рівня 2 189 з заголовком, що містить SFD (від англ. Start Frame Delimeter - початковий роздільник кадру) 181 для синхронізації, та преамбулу 180 як заголовок кадру даних. Максимальна довжина пакета Ethernet рівня 1 188 при цьому на 8 байт більше, ніж довжина пакета Ethernet рівня 2 189: від мінімального розміру 72 байта до максимальної довжини 1526 байт без додаткової VLAN або до 1530 байт з урахуванням блоку VLAN 184.

Під час роботи завдання преамбули 180 як підполя заголовка кадру даних рівня 1 полягає в тому, щоб допомогти обладнанню спочатку ідентифікувати пристрій, що намагається відправити дані. Початковий роздільник кадру (SFD) 181, ще один артефакт рівня 1, використовується для синхронізації даних вхідних пакетів з синхронізуючими таймерами, щоб забезпечити достовірне зчитування даних. Після прийому цих двох блоків пакета Ethernet рівня 1 188 пакет Ethernet рівня 2 189 починається з MAC-адреси одержувача 182 і MAC-адреси відправника 183, що описують, для якого мережевого пристрою, підключеного до LAN призначені ці дані та звідки вони надходять. Комутатор LAN - це інтелектуальний пристрій, здатний маршрутизувати дані відповідно до цих адрес. Блок VLAN 184 є додатковим і, якщо він присутній, полегшує фільтрацію пакетів шляхом їх розбиття на підсегмент або віртуальні локальні мережі відповідно до специфікації IEEE 802.1Q. Ethertype 185 визначає формат даних або як тип даних, або як його довжину в залежності від його формату. Ethertype 185 і VLAN 184 відповідають формату, який запобігає плутанину щодо того, чи введені дані про додаткову VLAN 184 чи ні.

Після того як всі дані заголовка прийняті, корисне навантаження 187 містить фактичні дані, які повинні передаватися пакетом Ethernet. Ці дані можуть відповідати Інтернет-протоколу і можуть містити дані, інкапсулюючи контент рівнів 3-7, як передбачено в моделі OSI. З іншого боку, в спеціально розроблених системах корисне навантаження 187 може містити протоколи, що належать конкретному обладнанню або виробникам. Якщо всі необхідні дані не можуть бути відправлені в пакеті максимального розміру 1500 байт, дозволеному стандартом Ethernet, тоді корисне навантаження може бути розбите на частини або відправлене з використанням альтернативного протоколу, наприклад, Jumbo-кадру, який може передавати до 9000 байт даних - в шість раз більше, ніж стандартний пакет Ethernet. Поле контролю кадру 186 містить просту інформацію, пов'язану з перевіркою помилок для пакета Ethernet рівня 2 189, але не даних пакета Ethernet рівня 1 для преамбули 180 або SFD 181. Перевірка кадру 186 здійснюється за алгоритмом перевірки циклічної надмірності довжиною 32 біта, здатному виявляти ненавмисні зміни в необроблених даних пакета Ethernet рівня 2 189.

Стандартне фізичне середовище для Ethernet - це електричний або волоконно-оптичний кабель, причому електричний кабель в даний час є найбільш поширеним. Швидкість передачі даних з часом змінилася від 10-100 Мбіт/с останнім часом до 1-100 Гбіт/с - це так званий "гігабітний Ethernet". Кабелі Ethernet використовують легко помітні роз'єми RJ-45 для захисту з'єднань між комутаторами LAN та такими пристроями, як сервери, стаціонарні комп'ютери, ноутбуки, приставки та модеми. У деяких випадках, до Ethernet може бути застосована технологія "Живлення через Ethernet" (Power over Ethernet, PoE). Ця технологія дозволяє подавати живлення на пристрій через виту пару в мережі Ethernet.

Wi-Fi (802.11).У багатьох випадках Ethernet використовується для встановлення бездротового з'єднання з мережею з мобільними пристроями з використанням лінії радіозв'язку ближньої дії. Згодом пропріетарні бездротові лінії були замінені стандартизованими системами ближнього зв'язку з протоколом, визначеним стандартом IEEE 802.11, з комерційною назвою Wi-Fi. Часто, поєднуючи функції маршрутизатора і комутатора з радіоприймачами та передавачами, роутери Wi-Fi тепер часто встановлюють в будинках, офісах, підприємствах, кафе і громадських місцях.

Радіолінія, показана на рисунку 10, ілюструє комбінацію двох взаємопов'язаних мереж, одна з яких містить пристрій "MAC-доступу до Ethernet" 200A, а інша містить радіолінію, а саме "точку доступу радіозв'язку" 200B. Інтерфейсна схема та відповідний мікропрограмний блок 202 забезпечують фізичний інтерфейс рівня 1, тобто фізичний міст 204A і 204B між електричною мережею і радіомережею, а також спрощує каналний рівень 2 205A і 205B між протоколом Ethernet та протоколом радіозв'язку, наприклад, Wi-Fi. В процесі роботи дані з Ethernet 201 надходять в стек зв'язку 203A, при цьому фізичні сигнали з'єднуються з інтерфейсом 202 за допомогою бездротової технології фізичного рівня 1 211, а інформація каналного рівня 2 проходить через з'єднання 205A.

Після обробки дані передаються з інтерфейсу 202 в стек зв'язку 203B точки радіодоступу 200B, при цьому фізичні сигнали підключаються за допомогою бездротової технології фізичного рівня 1 204B, а інформація каналного рівня 2 проходить через з'єднання 205B. Потім ця інформація надходить через з'єднання 204 в приймач радіозв'язку та транслюється за допомогою одного з "n" радіоканалів через радіолінії 206A-206N для виходу на радіоантену 207. При прийомі радіосигналів дані проходять по тому ж шляху, але в протилежному напрямку по відношенню до вищезгаданого опису.

Інтерфейс 202 також може діяти як комутатор LAN для підтримки одночасного зв'язку на різних радіоканалах одночасно з різними мережевими пристроями, підключеними до Ethernet, в цьому випадку до пристрою радіозв'язку підключається декілька кабелів Ethernet 201. В альтернативному варіанті кілька розмов можуть бути послідовно відправлені по одному з'єднанню Ethernet з вхідним пристроєм при використанні рівнів 3 і 4 для управління маршрутизацією пакетів для різних одержувачів.

Одним з стандартизованих пристроїв та протоколів ближнього радіозв'язку є бездротова локальна мережа або пристрій WLAN, що працює відповідно до специфікації IEEE 802.11. Такі пристрої з комерційною назвою Wi-Fi використовуються для бездротового доступу в Інтернет і для систем бездротового розподілення (Wireless Distribution Systems), тобто для радіозв'язку, що використовується замість дротових з'єднань в місцях, де прокладати кабелі незручно, складно або дорого. Крім основної специфікації IEEE 802.11, для визначення частот несучих, каналів, схем модуляції, швидкості передачі даних і діапазону радіозв'язку використовуються такі її підверсії, як 802.11a, 802.11n, 802.11ac і т.д. Зведена інформація про підверсії стандарту 802.11, схвалених IEEE на момент оформлення даної заявки, наведена в наступній таблиці:

Версія 802.11	Дата випуску	Частота несучої, ГГц	Смуга пропуску каналу, МГц	Максимальна швидкість передачі даних, Мбіт/с	Максимальна кількість MIMO	Модуляція	Дальність дії в приміщенні, м	Дальність дії поза приміщенням, м
a	Верес. 1999	5	20	6÷54	Немає	OFDM	35	120
		3.7					–	5,000
b	Верес. 1999	2.4	22	1÷11	Немає	DSSS	35	140
g	Черв. 2003	2.4	20	6÷54	Немає	OFDM	38	140
n	Жовт. 2009	2.4 або 5	20	7.2÷72.2	5	OFDM	70	250
			40	15÷150				
ac	Груд. 2013	5	20	7.2÷96.3	8	OFDM	35	–
			40	15÷200				
			80	32.5÷433.3				
			160	65÷866.7				

Версія 802.11	Дата випуску	Частота несучої, ГГц	Смуга пропуску каналу, МГц	Максимальна швидкість передачі даних, Мбіт/с	Максимальна кількість MIMO	Модуляція	Дальність дії в приміщенні, м	Дальність дії поза приміщеннями, м
ad	Груд. 2012	60	2,160	6,912	Немає	OFDM з однією несучою або малою потужністю	–	–

Згідно таблиці, в основному Wi-Fi працює на частотах 2,4 ГГц і 5 ГГц; частота 3,7 ГГц, призначена для маршрутизації систем бездротового розподілення (WDS) на великі відстані, до сих пір прийнята тільки в США. Для гігабітних швидкостей передачі даних недавно прийнята несуча 60 ГГц, сумісна з іншими високошвидкісними мережами, наприклад, з Gigabit Ethernet та волоконно-оптичний кабель з використанням DOCSIS 3.0. Для підтримки паралельної роботи декількох користувачів в кафе і громадських місцях стандарти 802.11n і 802.11g пропонують паралельне 5-канальне і 8-канальне підключення MIMO (багатоканальний вхід - багатоканальний вихід). Для досягнення високої пропускної здатності Wi-Fi в основному використовує OFDM (мультиплексування з ортогональним частотним розділенням каналів) в якості способу кодування цифрових даних на декількох близько розташованих ортогональних каналах піднесучих.

Під час роботи OFDM розподіляє єдиний сигнал по піднесучих, при цьому один надзвичайно швидкий сигнал розділяється на кілька повільних сигналів. Ортогональність в цьому контексті означає, що суміжні канали піднесучих не перекриваються, що виключає плутанину щодо розділення даних по каналах. Потім ці кілька піднесучих збирають в приймачі і рекомбінують для відновлення однієї високошвидкісної передачі. Оскільки швидкість передачі даних в каналах піднесучих нижче, ніж в одному високошвидкісному каналі, чутливість сигналу до спотворень та перешкод знижується, що підвищує надійність радіозв'язку навіть в умовах шуму і на великій відстані. За винятком особливого діапазону 3,7 ГГц, дальність дії Wi-Fi обмежується 70 м в приміщенні та 250 м на відкритому повітрі при підвищеній потужності мовлення. Wi-Fi не має можливості перемикавання, яке властиве стільниковому зв'язку, тому його складно використовувати в мобільному зв'язку на великі відстані, і тому така можливість відноситься до технології LTE, описаної нижче.

У режимі Wi-Fi з використанням OFDM-модуляції дані, що передаються, об'єднуються в "символи", тип представлення даних, який природним чином стискає багатоцифрові стани в меншу кількість символів. Потім ці символи передаються з низькою "швидкістю передачі символів" для забезпечення несприйнятливості до втрати даних, пов'язаної з проблемами перенесення несучої. Цей підхід забезпечує підвищення швидкості передачі даних зі зниженням частоти помилок, підвищенням якості обслуговування та зниженням чутливості до коливань рівня сигналу, радіовідображення та зовнішнього шуму або електромагнітних завад. Символом може бути будь-яка модуляція, наприклад, частота, тон або конкретний шаблон імпульсу, що корелює з кожним конкретним символом, де послідовність символів при фіксованій тривалості може бути перетворена в потік даних зі швидкістю передачі бітів, що перевищує швидкість передачі символів. Цей метод аналогічний прапорам семафора, де прапор може переміщатися в одну з шістнадцяти фіксованих позицій через заданий проміжок часу, наприклад, через одну секунду. Швидкість передачі символів, також відома як швидкість "в бодах", при цьому дорівнює одному символу в секунду або одному боду, де термін "бод" визначається як "кількість змін окремих символів, внесених в середу передачі в секунду". Так як прапор може мати 16 різних значень, у двійковому вигляді вісім станів еквівалентні 4 бітам, оскільки $2^4=16$ станів. Тоді швидкість передачі одного символу в секунду або 1 бод дорівнює швидкості передачі 4 біт/с, що в чотири рази перевищує швидкість передачі символів. Аналогічно, використовуючи 16 різних тонів для представлення символів, швидкість передачі символів в 10 млн. символів в секунду може призводити до швидкості передачі бітів 40 Мбіт/с.

Однак кількість використовуваних символів впливає не тільки на швидкість передачі бітів, але також на частоту помилок і якість обслуговування. Наприклад, якщо використовується занадто багато символів, точне визначення символів в умовах зовнішнього шуму може виявитися складним завданням для процесора цифрової обробки (DSP) радіосигналів, а частота помилок при розпізнаванні даних збільшиться і буде вимагати повторної передачі даних для отримання правильної контрольної суми при динамічному контролі CRC пакету. Використання меншої кількості символів при будь-якій заданій швидкості передачі символів

полегшує їх розпізнавання, але, в свою чергу, знижує швидкість передачі бітів та пропускну здатність зв'язку. За аналогією, якщо прапор семафора може переміщатися тільки в одну з чотирьох позицій, а не шістнадцяти, то його легше побачити під проливним дощем, так що ймовірність помилки зв'язку, тобто його неправильного зчитування, значно зменшується. При використанні тільки однієї з чотирьох позицій прапора швидкість передачі даних в бодах і раніше становить 1 символ в секунду, але швидкість передачі бітів падає до 2 біт/с, тому що $22=4$. Таким чином, існує внутрішній компроміс між швидкістю передачі даних в бітах та частотою помилок при розпізнаванні бітів, який Wi-Fi може регулювати шляхом динамічного коригування швидкості передачі символів. Аналогічний компроміс досягається в системі радіозв'язку стандарту LTE (довготривалого розвитку).

У версіях a, g і n стандарту 802.11 новий символ може передаватися через 4 мкс або з частотою 250 000 бод для кожного каналу піднесучої. Wi-Fi використовує 64 канали піднесучих, тому теоретична максимальна швидкість передачі символів повинна складати 16М бод при максимальній пропускну здатності каналу. Але для захисту від міжканальних перешкод фактично є 48 з 64 каналів піднесучої, що знижує швидкість передачі символів до 12М бод при максимальній пропускну здатності каналу. У сучасному радіозв'язку символи перетворюються в біти на декількох рівнях, ці рівні динамічно змінюються зі зміною умов радіозв'язку за рахунок використання різних схем фазової модуляції, наведених нижче в таблиці:

Фазова модуляція	Умови радіоканалу	Кількість бітів на символ	Швидкість передачі символів на піднесучу	Максимальна швидкість передачі символів в каналі Wi-Fi в багатоканальному режимі	Максимальна швидкість передачі бітів в каналі Wi-Fi
BPSK	Наявність шуму або далекий зв'язок	1	250к бод	12М бод	12 Мбіт/с
QPSK	Хороші, середня дальність	2			24 Мбіт/с
16-QAM	Дуже хороші, мала дальність	4			48 Мбіт/с
64-QAM	Відмінні, дуже мала дальність	6			72 Мбіт/с

де співвідношення між швидкістю передачі символів і швидкістю передачі бітів визначається наступним виразом:

$$(\text{Швидкість передачі бітів})/(\text{Швидкість передачі символів}) = \text{Кількість бітів в символі}$$

де швидкість передачі бітів вимірюється в бітах в секунду або біт/с, а швидкість передачі символів вимірюється в символах в секунду або "бодах". З наведених схем фазової модуляції "двохпозиційна фазова маніпуляція" (BPSK) найкраще працює на великих відстанях та в умовах значного шуму в системі радіозв'язку, але використовує чисто бінарний метод - один біт на символ, тому вона обмежена низькими швидкостями передачі даних. У хороших умовах радіозв'язку швидкість передачі даних перевищує швидкість передачі символів, тобто (Кількість бітів в символі) > 1, а швидкість передачі бітів в радіоканалі може бути збільшена в будь-якому місці 2-6 разів, у порівнянні з BPSK, в залежності від умов радіозв'язку, відсутність електромагнітних перешкод, відстані між приймачами і мовленнєвої потужності радіостанції. Наприклад, в хороших умовах або в системі дальності радіозв'язку методи "квадратурної фазової маніпуляції" (QPSK) забезпечують подвоєну швидкість передачі даних, в порівнянні з BPSK при 2 бітах в символі. За оптимальних умов, обмежених роботою малої дальності, "16-позиційна квадратурна амплітудна модуляція" (16-QAM) може забезпечити чотириразове збільшення швидкості передачі символів при 48 Мбіт/с у системі бездротового зв'язку Wi-Fi. При відмінних умовах в безшумному середовищі швидкість передачі даних може збільшуватися до 6 бітів в символі за допомогою 64-позиційної квадратурної амплітудної модуляції (64-QAM). Схеми фазової модуляції в системах зв'язку добре відомі фахівцям в даній області техніки і не будуть обговорюватися далі в цьому описі.

У системах, що відповідають стандартам 802.11b і 802.11g, використовується інша схема модуляції спектру - широкої смуги з прямим розширенням спектра (DSSS), де термін "розширення" відноситься до того факту, що в DSSS сигнали несучої виникають у всій смузі пропускання, тобто в спектрі частот передачі пристрою радіозв'язку. В системі DSSS модульна схема використовує безперервну послідовність символів псевдошумового коду, який коротше одного інформаційного біта, для фазового зсуву синусоїдальної хвилі псевдовипадковим чином

до передачі та віднімання того ж шуму з сигналу приймача. В результаті фільтрації некоррельований шум повністю видаляється, та зв'язок може здійснюватися надійно навіть при наявності шуму та електромагнітних завад в радіоканалі, навіть при відношенні сигнал-шум нижче одиниці. Оскільки покращений спектр використовує повний діапазон радіозв'язку, такі методи перестають бути кращими для OFDM і не використовуються в новітніх реалізаціях Wi-Fi.

Крім визначення деталей фізичного рівня для діапазонів радіозв'язку та схем модуляції, стандарт 802.11 може вплинути на формат послідовного пакета даних, необхідний для здійснення зв'язку з Wi-Fi радіомодулями. У порівнянні з пакетом Ethernet, заголовок пакета Wi-Fi є більш складним, через те, що він повинен вказувати адреси приймачої та передавальної станції, а також одну або дві мережеві адреси. Структура даних пакета Wi-Fi, наведена на рисунку 11, графічно показана як послідовність даних, що розміщуються зліва направо в напрямку збільшення часу 86. Відповідна таблиця 242 описує функцію кожного блоку та субпакета в пакеті Wi-Fi. Як і пакет Ethernet, кадр даних включає інформацію канального рівня 2, інкапсульовану в кадрі даних рівня 1 з заголовком рівня 1.

Заголовок Рівня 1 містить преамбулу 230 довжиною 10 байт та початковий роздільник кадру (SFD) 231 довжиною 2 байта, а також PLCP (процедуру сходження фізичних рівнів) 232 довжиною 2 байта. Незважаючи на те, що вважається, що PLCP містить дані і рівня 1, і рівня 2, тут будемо вважати, що це дані рівня 1. Тоді можна вважати, що заголовок рівня 1 має довжину 14 байт, а решту пакету Wi-Fi складають дані рівня 2, що змінюються по довжині від 34 байт за відсутності корисного навантаження до 2346 байт при максимальному корисному навантаженні 241, величина якого складає 2312 байта. При максимальній довжині корисного навантаження 2312 байта, Wi-Fi пакет довший, ніж Ethernet пакети, які в стандартній формі обмежені довжиною корисного навантаження 1500 байт. Компоненти рівня 2 Wi-Fi пакету, відповідно до зображення, включають в себе керування кадром 233, тривалість 234, MAC-адреси 1 і 2 базової радіостанції, показані як блоки 235 і 236 відповідно, умовні MAC-адреси 3 і 4, показані як блоки 237 і допоміжний блок 239 відповідно, послідовність 238 і контроль кадру 240.

Під час роботи завдання преамбули 230 як підполя заголовка кадру даних рівня 1 полягає в тому, щоб допомогти обладнанню спочатку ідентифікувати пристрій, що намагається відправити дані. Початковий роздільник кадру (SFD) 231, ще один артефакт рівня 1, використовується для синхронізації даних вхідних пакетів з синхронізуючими таймерами, щоб забезпечити достовірне зчитування даних. Після цих двох блоків процедура сходження фізичних рівнів (PLCP) 232 надає інформацію, що відноситься до довжини пакета, швидкості передачі даних та перевірці помилок заголовка.

Керування кадром 233, перші дані канального Рівня 2 визначають тип версії пакету Wi-Fi, тобто, чи містить він інформацію, що відноситься до управління, команди управління, даних, або до резервованих функцій, включаючи біти управління "до/від розподіленої системи", використовуваних для того, щоб визначити, чи працює радіостанція в якості точки доступу або системи бездротового розподілення. Поле "тривалість" 234, також зване "тривалість і ідентифікатор", визначає тривалість вектора надання мережі (NAV), тобто, як довго радіосередовище буде зайняте до того, як інша станція зможе претендувати на її надання, за винятком режиму енергозбереження, де вона містить інформацію, що ідентифікує свій "ідентифікатор станції", який використовується для розпізнавання своїх маяків при перевірці активності. Крім інформації про тривалість, блоки адреси 1 і адреси 2 235 і 236 визначають адреси базової станції, по суті, це MAC-адреси радіоприйомопередавача.

Зокрема, адреса 1 в блоці 235 містить адресу приймачої станції BSS (від англ. Base Station Subsystem - підсистеми базової станції), а адреса 2 в блоці 236 містить адресу передавальної станції BSS. При здійсненні зв'язку між двома радіостанціями, адреси яких завантажені в Адресу 1 і Адресу 2, залежить від налаштування "до/від розподіленої системи", визначеної у блоці 233, що здійснює управління кадром. Адреса 3, визначена в блоці 237, використовується для зв'язку радіостанції з фізичною мережею, наприклад, використовуючи Ethernet, по суті описуючи, звідки поступають дані, що передаються, або, навпаки, куди отримані дані мають бути направлені. Таким чином, адреса, присутня в адресі 3, також залежить від настройки "до/від розподіленої системи", визначеної в пакеті Wi-Fi. Для забезпечення сумісності з Ethernet-з'єднаннями, адреси Wi-Fi мають довжину 6 байт, таку ж, як MAC-адреси, що використовуються в локальних мережах Ethernet.

Щоб визначити напрямок передачі даних і мати можливість реорганізувати пакети, отримані з порушеним порядком, тобто пошкоджені через вплив фазових затримок в радіоканалі, блок послідовності 238 містить номери послідовностей і фрагментів, що визначають кадр пакету. Якщо пакет Wi-Fi не ідентифікований як пакет WDS (системи бездротового розподілення), то додаткова адреса 239 виключається з пакета Wi-Fi. Після блоків управління адресою та

послідовністю знаходиться корисне навантаження 241 - фактичний вміст, що доставляється пакетом Wi-Fi, включаючи дані рівнів 3-7 моделі OSI. Після цього здійснюється перевірка кадру 240 за алгоритмом перевірки циклічної надмірності довжиною 32 біта для виявлення випадкових змін в необроблених даних пакета Ethernet рівня 2.

5 Як описано вище, коли Wi-Fi радіомодуль використовується в якості "точки доступу", наприклад, щоб забезпечити радіоз'єднання мобільного пристрою до Інтернету, необхідні тільки три MAC-адреси - передавальної радіостанції, приймальної радіостанції і Ethernet-з'єднання. Порядок присвоєння адрес залежить від напрямку потоку даних, що визначається параметром "до/від розподіленої системи". Термін DS (Distribution System) є скороченою назвою розподіленої системи провідної мережі або Ethernet-з'єднання, до якого підключена радіостанція. Порядок адресації в пакеті Wi-Fi при роботі з точкою доступу Wi-Fi показаний на рисунку 12A, де верхній рисунок відповідає випадку, коли мобільна радіостанція, в даному прикладі ноутбук 260, здійснює бездротову передачу даних в точку доступу Wi-Fi 261 і на розподілену систему через Ethernet 265, а нижній рисунок відповідає випадку, коли дані з розподіленої системи направляються в точку доступу Wi-Fi 261 через Ethernet 265, а потім відправляються по бездротовій лінії зв'язку в ноутбук 260.

Як представлено на верхній частині рисунка, під час роботи дані відправляються з Wi-Fi радіомодуля в ноутбук 260 з використанням радіосигналу 264, що передається від антени 262A та приймається антеною 262B підсистеми базової станції (BSS) в точці доступу Wi-Fi 261, яка, в свою чергу, відправляє пакет до розподіленої системи через Ethernet 265. У цьому випадку послідовність 238 містить біти "до/від розподіленої системи", показані в таблиці 263, де для біта "до розподіленої системи" встановлено значення 1, а для біта "від розподіленої системи" встановлюється значення 0. У такому випадку адреса 1 в блоці 235-MAC-адресу одержувача в системі радіозв'язку - містить адресу приймача BSS Wi-Fi; адреса 2 в блоці 236-MAC-адресу відправника в системі радіозв'язку - містить адресу передавального радіомодуля ноутбука і адреса 3 в блоці 237 містить MAC-адресу одержувача будь-якого підключеного до розподіленої системи мережевого пристрою, що використовує Ethernet 265.

Як представлено на нижній частині рисунка, де потік даних спрямований у протилежний бік, MAC-адреси відправника і одержувача в системі радіозв'язку міняються місцями, а адреса Інтернету змінюється з MAC-адреси одержувача на MAC-адресу відправника. У цьому випадку послідовність 238 містить біти "до/від розподіленої системи", показаної в таблиці 263, де біт "до розподіленої системи" встановлюється рівним бінарного 0, а біт "від розподіленої системи" встановлюється рівним двійковій 1, при цьому адреса 1 в блоці 235-MAC-адресу одержувача в системі радіозв'язку - містить адресу радіоприйомного модуля ноутбука, адреса 2 в блоці 236-MAC-адресу відправника в системі радіозв'язку - містить адресу передавача BSS Wi-Fi, а адреса 3 в блоці 237 містить MAC-адресу відправника будь-якого мережевого пристрою, що використовує Ethernet 265. У процесі роботи пакети даних відправляються через розподілену систему від мережевого пристрою через Ethernet 265 в підсистему базових станцій (BSS) до точки доступу Wi-Fi 261, яка, в свою чергу, за допомогою антени 262B передає радіосигнал 264, який приймається антеною 262A в радіомодуль Wi-Fi ноутбука 260.

Специфікація Wi-Fi також передбачає використання радіомодулів Wi-Fi для реалізації систем бездротового розподілення (WDS), відповідно до рисунка 12B. За схемою побудови, WDS-системи - це бездротова реалізація провідної мережі, тобто радіо аналог мережевого кабелю. Однак для реалізації WDS при маршрутизації пакетів є потреба у додатковій адресі - адресі 4, що міститься в блоці 239. У простому поданні маршрутизація пакетів через систему бездротового розподілення Wi-Fi вимагає послідовного використання чотирьох MAC-адрес, за допомогою чого (1) вхідний пакет з мережевої MAC-адреси відправника підключається через Ethernet до (2) MAC-адресу відправника передавальної радіостанції, який, в свою чергу, підключається до (3) MAC-адресу одержувача приймальної радіостанції, з якої, нарешті, пакет відправляється через Ethernet до (4) мережевого MAC-адресу одержувача. Для роботи радіомодуля Wi-Fi в режимі WDS блок послідовності пакетів Wi-Fi 238 містить дані, показані в таблиці 263, де для обох параметрів "до/від розподіленої системи" встановлюється двійковий 1 стан.

Напрямок даних пакета при цьому легко визначається використанням чотирьох MAC-адрес, двох для систем бездротового розподілення та двох для Wi-Fi радіомодуля. Як представлено на верхній частині рисунка 12B, вхідний пакет, прийнятий по Ethernet 269A, приймається базовою станцією Wi-Fi WDS A 268A, транслюється у вигляді радіосигналу 264 антеною 262A передавальної радіостанції, приймається антеною 262B приймаючого радіомодуля базової станції Wi-Fi WDS B 262B та перенаправляється через Ethernet 269B на MAC-адресу одержувача. Для керування маршрутизацією адреса 1 в блоці 235 являє собою MAC-адресу

одержувача радіолінії, тобто адресу Wi-Fi WDS B, адреса 2 в блоці 236 містить адресу відправника радіолінії, тобто адресу Wi-Fi WDS A, адреса 3 в блоці 237 являє собою MAC-адресу одержувача Ethernet, яку перенаправлено на Ethernet 269B, а адреса 4 в блоці 239 містить адресу відправника Ethernet, яку прийнято по Ethernet 269A.

Для передачі даних в зворотному напрямку від базової станції Wi-Fi WDS B 268B до базової станції Wi-Fi WDS A 268A, показаної в нижній частині рисунка 12B, адреси відправника та одержувача просто міняються місцями, в результаті чого адреса 1 в блоці 235 являє собою MAC-адресу одержувача радіолінії, тобто адреса Wi-Fi WDS A, адреса 2 в блоці 236 містить адресу відправника радіолінії, тобто адреса Wi-Fi WDS B, адреса 3 в блоці 237 являє собою MAC-адресу одержувача Ethernet, перенаправлений на Ethernet 269A, і адреса 4 в блоці 239 містить адресу відправника Ethernet, прийняту по Ethernet 269B.

При цьому пакет Wi-Fi є дзеркальним відображенням кадру даних Ethernet, і містить адресу 3 в якості MAC-адреси одержувача і адресу 4 в якості MAC-адреси відправника, як якщо б радіолінія взагалі не брала участі в маршрутизації. Таким чином, при маршрутизації пакетів по мережі з комутацією пакетів система бездротового розподілення, реалізована у вигляді Wi-Fi, веде себе аналогічно провідної мережі. Крім того, функція керуючих бітів "до/від розподіленої системи" дозволяє одному й тому ж Wi-Fi радіомодулю працювати в якості двонаправленого каналу передачі даних, тобто WDS, або в якості двобічної точки доступу до мережі.

Телефонний зв'язок четвертого покоління (4G)/стандарт LTE (Long Term Evolution - довготривалий розвиток) - Аналогічно переходу провідної телефонної системи зв'язку від мереж з комутацією каналів на систему зв'язку з комутацією пакетів, замінюючи традиційну телефонну систему (POTS) і абонентську мережу зв'язку (PSTN) спочатку такими цифровими мережами, як ISDN (англ. Integrated Services Digital Network - цифрова мережа з інтеграцією служб), які в своїй роботі використовують пропріетарне обладнання, а пізніше, IP-мережами з приватною хмарною інфраструктурою, - такий же перехід здійснив і бездротовий зв'язок. Згідно рисунку 13, еволюція цифрового стільникового зв'язку починалася з послуг передачі голосових і коротких повідомлень (системи SMS) 290, переданих по мережах з комутацією каналів, званим GSM - розшифровка цієї аббревіатури, спочатку означала "Groupe Spécial Mobile", згодом була змінена на "Global System for Mobile Communications"(Глобальна система мобільного зв'язку). GSM, яка вважається другим поколінням (2G) бездротового телефонного зв'язку, оптимізованим для повнодуплексного мовленнєвого зв'язку, замінила початкові аналогові стільникові (1G) мережі, побудовані на основі протоколу множинного доступу з розділенням за часом (TDMA). Наступне удосконалення телефонного зв'язку, показане блоком 291, з'явилося з метою розширення можливостей GSM, забезпечуючи більш високу пропускну здатність і додаючи такі функції, як передача мультимедійних повідомлень (MMS). Все ще покладаючись на мережеві технології з комутацією каналів, вдосконалені мережі розглядалися як перехідні, що отримало відображення в їх назві - 2.5G.

Перший крок до мобільного телефонного зв'язку покоління 3G був зроблений з впровадженням "пакетного радіозв'язку загального користування" (GPRS) шляхом переходу як бездротової інфраструктури, так і для програмного забезпечення телефонної мережі зв'язку з комутацією пакетів, поліпшення послуг голосового зв'язку, SMS та MMS за допомогою стандарту PTT (Push To Talk - натисни і говори), постійного доступу до Інтернету, протоколу для додатків бездротового зв'язку (WAP) і т.д., згідно блоку 292. На основі множинного доступу з кодовим розділенням каналів (CDMA) GPRS також покращує якість зв'язку, збільшує пропускну здатність мережі та підвищує продуктивність системи. Наприклад, доставка SMS-повідомлень через GPRS здійснюється з такою швидкістю, яка принаймні, в три рази перевищує швидкість GSM. При швидкості 384 Кбіт/с продуктивність CDMA була в 40 разів вище, ніж для попередніх рішень на основі GSM.

Перехід на CDMA став значною подією, оскільки він включав заміну та перезавантаження інфраструктури мобільного зв'язку всього світу з використанням нових приймачів і антен. Після розгортання технології WCDMA (широкосмугового множинного доступу з кодовим розділенням каналів) був зроблений другий, ще більш важливий крок в 3G-телефонії з використанням UMTS ("універсальної системи мобільного зв'язку") - стандарту, розробленого в рамках проекту партнерства третього покоління (3GPP), реалізує глобальний та всеосяжний підхід до визначення і розгортання справді універсальною мережі та стандартизованого протоколу. Щоб розширити свої можливості та підвищити пропускну здатність мережі, в стандарті UMTS прийнятий новий протокол - широкосмуговий множинний доступ з кодовим розділенням каналів (технологія радіодоступу WCDMA), що пропонує підвищення спектральної ефективності та пропускну здатності операторам мереж мобільного зв'язку, не вимагаючи від них додаткових

інвестицій в обладнання для технології 3G. У перших мережах максимальна швидкість передачі даних в низхідній лінії зв'язку становила 3,6 Мбіт/с.

Одночасно, розробка білого світлодіода і ефективної мініатюрної схеми управління світлодіодом вперше дала можливість використання кольорових дисплеїв в мобільних пристроях та сприяла появі смартфона. Смартфон був критичним каталізатором для комерційного використання пропускної здатності мережі, оскільки кольорові дисплеї підвищеної якості негайно створювали попит на швидкий доступ в Інтернет, завантаження фільмів, фотозйомку з високою роздільною здатністю, мультимедійне мовлення і навіть на обмежене онлайн-відео в реальному часі. Щоб задовольнити цей попит, в оновлених мережах було розгорнуто високошвидкісний пакетний доступ (HSPA), також відомий як 3.5G, що підвищує як швидкість завантаження і швидкість передачі даних в низхідній лінії зв'язку, в той же час, використовуючи методи модуляції WCDMA. Розгортання відбувалося по етапах - спочатку була запущена технологія високошвидкісної пакетної передачі даних від базової станції до мобільного телефону (HSDPA) як версія 3GPP Release 5, а незабаром після цього з'явилася технологія високошвидкісної пакетної передачі даних від мобільного телефону до базової станції (HSUPA) як версія 3GPP Release 6. Максимальна швидкість передачі даних підвищилася приблизно до 14 Мбіт/с в низхідній лінії зв'язку та приблизно до 5,8 Мбіт/с у висхідній лінії зв'язку, але вона суттєво відрізняється в залежності від географічного положення інфраструктури.

Ще до широкого розвороту HSUPA провайдери стільникового зв'язку перейшли на технологію HSPA +, вперше певну та стандартизовану в версії 3GPP Release 8, також відомої як "Довготривалий розвиток 3GPP" (LTE). Ця технологія являє собою мережу з комутацією пакетів, засновану на "мультиплексуванні з ортогональним частотним розділенням каналів множинного доступу" (OFDMA) на основі того ж методу OFDM, який використовується в Wi-Fi і який обговорювався раніше. Хоча OFDM була розроблена для одного користувача з типом зв'язку "один-до-одного", OFDMA можна розглядати як версію для багатьох користувачів, оскільки в ній передбачена можливість динамічно призначати підмножина піднесучих для окремих користувачів.

При первинному розгортанні LTE на базі HSPA + швидкість передачі даних становила 21 Мбіт/с. У 2008 році в секторі радіозв'язку Міжнародного союзу електрозв'язку (ITU-R) був визначений набір вимог до стандартів 4G, названий специфікацією IMT-A (International Mobile Telecommunications Advanced - поліпшеною міжнародного мобільного зв'язку), що встановлює мінімальні вимоги до максимальної швидкості передачі даних для служби 4G на рівні 100 Мбіт/с для зв'язку з високомобільними системами, наприклад, з поїздами і автомобілями, і 1 Гбіт/с для зв'язку з низькомобільними об'єктами, такими як пішоходи та стаціонарні користувачі.

Оскільки ранні системи LTE на основі HSPA + не відповідали специфікації IMT-A по швидкості передачі даних, такі ранні спроби створення систем 4G офіційно не визнавалися як 4G-телефонія, хоча що вони і використовували модуляцію OFDMA і мережі з комутацією пакетів. Внаслідок цього не існує єдиної думки про те, до якої категорії слід відносити технологію HSPA + - до телефонного зв'язку пізнього 3G або раннього 4G з комутацією пакетів. Було навіть запропоновано назву 3.9G. Незалежно від назви, 4G-телефонія, показана в блоці 293, сьогодні вважається зв'язком з комутацією пакетів на основі OFDMA-модуляції і її різних реалізацій. Незважаючи на технічні та історичні зміни протоколів даних та використання неоднорідних бездротових мереж, в середовищі фахівців терміни 4G, LTE і 4G/LTE використовуються неоднозначно і можуть підміняти один одного.

Висока швидкість передачі даних та відносно висока продуктивність 4G/LTE-телефонії багато в чому зобов'язані методам модуляції та структурі кадрів даних. Згідно рисунку 14A, модуляція в системах 4G здійснюється в смузі частот 20 МГц навколо центральної несучої частоти, зазвичай в діапазоні від 700 МГц до 2,6 ГГц, яка розділяється на смуги частот піднесучих, де для низхідної лінії зв'язку передбачено безліч вузьких діапазонів 296a-296N для каналів піднесучих, необхідних OFDMA. Для економії електроенергії в мобільних пристроях для висхідної лінії зв'язку передбачено меншу кількість широких діапазонів 295A-295N та використовується одноканальна версія технології множинного доступу з частотним розділенням каналів (SC-FDMA). Різні частотні діапазони 295A-295N використовуються для одночасної підтримки декількох користувачів, але на відміну від OFDMA, вони не використовуються для розділення одного високошвидкісного потоку даних на кілька. В результаті швидкість передачі даних по висхідній лінії зв'язку для SC-FDMA неминуче менше, ніж швидкість передачі даних по низхідній лінії зв'язку в системі на основі OFDMA.

Ліцензійні несучі частоти, наведені нижче в таблиці, змінюються в залежності від регіону, де телефони з однієї країни можуть не працювати в іншій країні, якщо не використовується багатодіапазонний або загальносвітовий телефон, призначений для глобального роумінгу.

Регіон	Частоти, МГц	Діапазони
Північна Америка	700, 750, 800, 850, 1900, 1700/2100 (AWS – покращені послуги бездротового зв'язку), 2500, 2600	4, 7, 12, 13, 17, 25, 26, 41
Південна Америка	2500	3, 7, 20
Європа	800, 900, 1800, 2600	3, 7, 20
Азія	1800, 2600	1, 3, 5, 7, 8, 11, 13, 40
Австралія/Нова Зеландія	1800, 2300	3, 40

5

Вищезазначені ліцензійні частоти можуть бути змінені рішеннями комісій по зв'язку, керуючих ліцензуванням радіочастот в різних регіонах.

На рисунку 14В, показаний фізичний рівень 4G включає в себе набір радіоімпульсів тривалістю 10 мс, що утворюють 4G-пакет або кадр 300. Кожен кадр 300 ділиться на 20 інтервалів тривалістю 0,5 мс, що містять 7 OFDM-символів 302. Кожен символ 304 відділений від інших циклічним префіксом 303 і містить п'ятдесят ресурсних блоків 305, пронумерованих від 0 до 49, при цьому кожен блок 306 містить 84 ресурсних елементи 307, що включають 7 символів і 12 піднесучих. Ця структура даних підтримує гнучку систему кодування, яка використовується для реалізації високої швидкості передачі даних, забезпечення надмірності та зменшення кількості помилок.

На рисунку 15 показано наповнення контентом каналного рівня 2 в кадрі даних 4G 299 для OFDMA-модуляції, що використовується для завантаження даних 4G. Подібний пакет даних 4G існує для вивантаження даних SC-FDMA, але не включений в даний опис, оскільки він аналогічний наведеному пакету. Згідно рисунку, кожен пакет даних фізичного рівня 1 або "кадр даних" 299 містить кадр 300 тривалістю 10 мс з двадцятьма інтервалами 301 тривалістю 0,5 мс, що утворюють каналний рівень 2. Контент каналу передачі даних рівня 2 в пакеті 4G має три рівні вкладення, що містять:

- підрівень MAC (Media Access Control) управління доступом до мультимедіа;
- підрівень RLC (Radio Link Control) "управління радіоканалом";
- підрівень PDCP (Packet Data Convergence Protocol) "протоколу конвергенції пакетних даних".

Підрівень MAC рівня 2 містить заголовок MAC 303, одиночний кадр блоків службових даних (SDU) MAC 304 і неінформативний часовий інтервал 305, де термін SDU (Service Data Unit) - скорочена назва блоку службових даних. Заголовок MAC 303 включає в себе необхідні MAC-адреси відправника та одержувача для здійснення радіозв'язку. Кожен окремий кадр блоків службових даних SDU MAC 304, в свою чергу, містить блоки даних протоколу управління радіоканалом (RLC PDU) 306 рівня 2, скорочення RLC PDU (Radio Link Control Protocol Data Unit) означає "блок даних протоколу управління радіоканалом", який використовується для управління роботою радіоканалу. Зокрема, PDU RLC 306 містять заголовок RLC 307, який вказує інформацію про роботу радіозв'язку та протоколах, а також вміщують інформацію блоку службових даних управління радіоканалом, тобто одиночний кадр SDU RLC 308 в якості вложеного корисного навантаження. Після завершення передачі SDU RLC 308 в момент часу 309 та після короткочасної затримки 310 починається передача нових даних управління радіоліній з заголовком RLC 311 і іншим набором SDU RLC. В результаті формується послідовний потік даних багатоканальних RLC SDU 319, дані для K-го і (K+1)-го блоків 313 і 314 передаються виключно однокадрово RLC SDU 308 і де (K+2)-й блок 314 складається з обох блоків - 308 з поточного кадру і 312 з наступного.

На підрівні протоколу перетворення даних пакетів рівня 2 кожен блок SDU містить комбінацію заголовка PDCP і PDCP SDU. Наприклад, K-й блок 313 містить заголовок PDCP 312A і PDCP SDU 323, (K+1)-й блок 314 містить заголовок PDCP 321B і PDCP SDU 324, а (K+2)-й блок 315 містить заголовок PDCP 321C і PDCP SDU 325, спільно формуючи PDCP PDU 320. Контент PDCP SDU 323, 324, 325 в свою чергу містить корисне навантаження 330 пакета 4G, а саме блоки даних 333, 334 і 335, включаючи дані мережевого, транспортного та прикладного рівня. Сьогодні вся вищезазначена обробка, необхідна для складання, передачі, прийому і декодування зв'язку 4G/LTE, виконується в одній пропрієтарній комунікаційній мікросхемі або цифровому сигнальному процесорі (DSP).

Використання протоколу рівня 2 дозволяє системі 4G реалізувати численні удосконалення в порівнянні з попередніми мережами та стандартами зв'язку, в тому числі:

- можливість використання технології MIMO (Multiple Input Multiple Output - "багатоканальний вхід - багатоканальний вихід") для максимізації швидкості передачі даних та забезпечення зв'язку з високою якістю обслуговування;

- використання програмно-керованих радіостанцій для одночасного підключення до декількох мереж радіозв'язку з метою динамічного визначення найбільш підходящих параметрів обслуговування, в тому числі, наприклад, вартості, якості обслуговування та пропускну здатності, для цього додатка;

- використання базових станцій, що підтримують внутрішньомережеве та міжмережеве перемикання, що забезпечує безперервність обслуговування з нульовим або мінімальним перериванням без помітної втрати якості обслуговування;

- можливість одночасного доступу до послуг і додатків в різних мобільних і бездротових мережах.

Додатки зв'язку 4G/LTE включають онлайн-відео високої і надвисокої чіткості, хмарні обчислення, зберігання великого обсягу даних в хмарному сховищі і резервне копіювання в режимі онлайн, прискорений доступ до Інтернету, можливість відправки і отримання великих файлів електронної пошти та багато іншого.

DOCSIS3/Кабельні та волоконно-оптичні мережі - До недавнього часу розподілені системи кабельного телебачення та волоконно-оптичного відео з комутацією пакетів відставали від інших систем галузі зв'язку в використанні цифрового мовлення та технології з комутацією пакетів. Однак завдяки швидкому впровадженню третього покоління "специфікації інтерфейсу передачі даних по кабелю" (DOCSIS3), можливості кабельної мережі значно покращилися і надають унікальну можливість одночасно обслуговувати велику кількість клієнтів по декількох каналах зв'язку з високою пропускну спроможністю. DOCSIS3 одночасно забезпечує цифровий повнодуплексний зв'язок з високою пропускну здатністю і доступ в Інтернет, технологія VoIP, а також підтримує багатоканальну передачу потокового відео високої чіткості, включаючи сотні широкомовних телевізійних каналів і каналів класу преміум, одностороннє телебачення з послугою платного телебачення та завантаженням IP-телебачення.

Приклад кабельної та волоконно-оптичної мережі на основі DOCSIS3, що підтримує декількох незалежних користувачів, показаний на рисунку 16. При розділенні кабельних каналів трансляцією контенту і клієнтським зв'язком управляє центральний пристрій кабельного головного вузла, що називається "системою підключення кабельних модемів" (CMTS) 350. В CMTS 350 надходить контент від різних пристроїв, включаючи головний вузол відео 351, що надає інформацію мережевого телебачення; систему IPTV 352, що забезпечує односторонню передачу послуги платного телебачення, а також завантаження IPTV і фільмів; технологію VoIP 353 для телефонного зв'язку та Інтернет 20 для підключення до хмарних систем і мережі Інтернет. Сукупна інформація, що включає високошвидкісні цифрові дані (HSD), технологію VoIP, мовлення і IPTV, відправляється клієнтам як кілька каналів 354 по одному коаксіальному або волоконно-оптичному кабелю.

Пакети даних, що розділяються CMTS 350, потім передаються великій кількості абонентів, а пристрої, що включають кабельний модем, об'єднаний з телевізійною приставкою KM/ТВП 357, підключаються до телевізора високої чіткості 39, або кабельний модем KM 358 передає сигнали голосового зв'язку на телефон 37, та забезпечують високошвидкісну цифрову зв'язність зі стаціонарним комп'ютером 38 і домашнім Wi-Fi передавачем 26. Подібно до того, як це робиться в мережах з шинами і концентраторами, весь сукупний контент, який передається по каналах 354, передається по одному кабелю або волоконно-оптичній лінії зв'язку та приймається всіма мережевими пристроями, підключеними до CMTS.

За допомогою DOCSIS3 система підключення кабельних модемів CMTS 350 стала абонентською мережею, де весь вміст не обов'язково надсилається кожному абоненту. Ця функція, відома як "зв'язування", дозволяє CMTS 350 управляти каналами, які можуть прийматися різними мережевими пристроями абонента. Згідно рисунку, пов'язані канали 355 передають контент для телевізора 39 і IP-телебачення, а пов'язані канали 356 передають високошвидкісні цифрові дані та голос. Об'єднані кабельний модем та телевізійна приставка KM/ТВП 359 мають доступ до двох груп каналів 355 і 356, які можна використовувати в телевізорі 39 - це смарт-телевізор, в той час як кабельний модем KM 360, який використовується для стаціонарного комп'ютера 36, телефону 37 і домашньої мережі Wi-Fi 26, підключений тільки до пов'язаних каналах HSD/технологія VoIP 356, оскільки для нього не потрібне підключення відео.

Як і в попередніх прикладах для Ethernet, Wi-Fi і 4G/LTE, поширення контенту з використанням DOCSIS3 по кабелю та волоконно-оптичній лінії зв'язку забезпечує можливість роботи в повнодуплексному режимі, при цьому реалізується технологія з комутацією пакетів. Завдяки використанню оптичного випромінювання замість електричних або радіорелейних сигналів для передачі інформації на фізичному рівні, волоконно-оптична лінія зв'язку, зокрема, забезпечує значно більшу пропускну здатність у порівнянні з іншими видами зв'язку. Стек зв'язку OSI для DOCSIS3 в кабельній розподіленій системі показаний на рисунку 17 та ілюструє зв'язки фізичного рівня 1, каналний рівень 2 та розташований вище мережевий рівень 3 як для пристрою підключення кабельних модемів CMTS 101, так і для прикладів мережевих пристроїв, наприклад, кабельного модему KM 103 або телевізійної приставки ТВП 102. Зокрема, пристрій підключення кабельних модемів CMTS 101 містить мережевий інтерфейс 361 фізичного рівня 1, підключений до хмарного сервера 22 і Інтернет 20, або, як альтернатива, з головним вузлом 351, системою І Р-телебачення 352 або технологією VoIP 353, показаними на попередньому рисунку. Комбінація мережевого інтерфейсу 361 і каналного рівня 366 утворює стек зв'язку інтерфейсу пристрою для CMTS 101.

На каналному рівні 2 дані передаються з стека зв'язку мережевого інтерфейсу в стек зв'язку кабельного мережевого інтерфейсу за допомогою функції перемикачання 370, зокрема, на підрівень управління логічним зв'язком (LLC) 369 каналного рівня. Підрівень LLC 369 утворює незалежний від устаткування протокол, визначений відповідно до специфікації IEEE 802.2. Потім пакетні дані модифікуються засобами захисту каналу 368, щоб забезпечити обмежений захист пакетів, в першу чергу для запобігання несанкціонованого перегляду такого контенту, як одностороннє мовлення з послугою платного телебачення. Потім пакети даних формуються відповідно до DOCSIS3 для включення кабельних MAC-адрес 367 способом, подібним показаному в прикладі для моста Wi-Fi радіомодуля на рисунку 10. Після цього кабельний інтерфейс 362 фізичного рівня 1 відправляє кадри даних через розподілену мережу 102, що містить або коаксіальний кабель 104 або волоконно-оптичний кабель 91, відповідний кабельний інтерфейс 363 фізичного рівня 1 в кабельному модемі KM 103 або телевізійної приставки ТВП 102. Кабельний інтерфейс 363 являє собою фізичний рівень стека зв'язку кабельного мережевого інтерфейсу кабельного модему KM 103 або телевізійної приставки ТВП 102.

Після прийому пакета даних кабельний MAC інтерфейс 371 інтерпретує кабельні MAC-адреси, передаючи їх корисне навантаження засобів захисту каналу 372 для дешифрування і, нарешті, до незалежного від устаткування інтерфейсу підрівня управління логічним зв'язком (LLC) 372 для інтерпретації. Потім вхідні дані передаються в стек зв'язку кабельної мережі KM або ТВП через прозорий міст 374 в стек зв'язку інтерфейсу пристрою KM або ТВП, зокрема, незалежний від пристрою підрівень LLC 375 відповідно до специфікації IEEE 802.2. Потім пакет передається або в блок MAC HSD і IPTV 376, або в блок MAC Wi-Fi 802.11 377 для оновлення MAC-адрес пакету. У разі Wi-Fi-підключення пакет даних передається з блоку 802.11 MAC 377 на радіоінтерфейс 365 фізичного рівня 1 Wi-Fi для передачі по Wi-Fi радіомодулю 26. У разі дротового підключення пакет даних передається з блоку MAC HSD і IPTV 376 в інтерфейсний блок 364 Ethernet або HDMI для підключення до телевізора 39 або стаціонарного комп'ютера 36.

Подібно OFDM, використовуваному в мережі Wi-Fi або OFDMA, використовуваному в системі зв'язку 4G/LTE, система зв'язку DOCSIS3 використовує кілька ортогональних, тобто тих, що не перекриті, частот в радіорелейних або в оптичному діапазоні електромагнітного випромінювання, в якому вона кодує та передає свою інформацію. Замість того щоб спеціально призначати контент для кожного каналу, DOCSIS3 підтримує "решітчасте кодування" - можливість динамічно розподіляти та перерозподіляти контент, включаючи відео, високошвидкісні дані і голос, по всім доступним частотним каналам. Згідно кількох прикладів кодування на рисунку 18 з використанням 1-6 каналів, пакети даних, що представляють даний тип контенту, можуть бути призначені одному каналу або розділені по декількох каналах. Дані розділяються як по каналах 385, так і по тимчасових інтервалах 386. У прикладі, позначеному $m=1$ (QPSK), тимчасові інтервали t_0 - t_8 кодується в одному каналі для доставки контенту від одного джерела №1. У прикладі, позначеному $m=2$ (8-QAM), два канали, закодовані з використанням алгоритму 8-QAM, використовуються для доставки контенту від двох джерел. Метод модуляції - квадратурна амплітудна модуляція (QAM) - той же, що використовується Wi-Fi, він вже обговорювалося раніше та повторно розглядатися не буде. Джерело №1 передає дані в інтервалі часу від t_0 до t_4 , потім передає дані джерело №2 в інтервалі від t_4 до t_8 . У прикладі, позначеному $m=3$ (16-QAM), три канали, закодовані з використанням алгоритму 16-QAM, використовуються для доставки даних від трьох джерел. Паралельно з джерелом №2, передає контент 390 по каналу $m=1$ в інтервалі часу від t_0 до t_8 , джерело №1 передає контент

391a в інтервалі часу від t_0 до t_4 по каналах $m=2$, а джерело №2 передає контент 391b в інтервалі часу від t_4 до t_8 .

У прикладі, позначеному $m=5$ (64QAM), шість каналів, закодованих з використанням алгоритму 64QAM, використовуються для доставки контенту від п'яти джерел. Наприклад, на двох підканалах $m=5$, позначених $m=2$, контент від джерела №3 передається в інтервалі часу від t_0 до t_4 , а контент від джерела №5 передається в інтервалі часу від t_4 до t_8 . В цей же час по підканалах, позначених $m=4$, контент від джерела №1 передається по чотирьох каналах в інтервалі часу від t_0 до t_2 , а потім тільки по трьох каналах в інтервалі часу від t_2 до t_3 . Передача контенту від джерела №2 починається в момент часу $t=t_2$ тільки на одному з чотирьох каналів, а потім в момент часу t_3 це число каналів збільшується до $m=4$. У прикладі, позначеному $m=6$ (128QAM), контент 389 від джерела №3 передається по двох каналах з шести в інтервалі часу від t_0 до t_4 в той час як інші чотири канали використовуються для передачі контенту 388a від джерела №1 в інтервалі часу від t_0 до t_2 і для передачі контенту 388b від джерела №2 в інтервалі часу від t_2 до t_4 . У наведених прикладах решітчасте кодування надає кабельному провайдеру максимальну гнучкість при управлінні пропускнуою спроможністю і розділенні контенту.

У відповідному пакеті даних, що використовується в DOCSIS3, показаному на рисунку 19, фізичний рівень 1 утворює кадр 390 фізичного носія даних змінної довжини та тривалості, що містить дані MAC канального рівня 2, що включають преамбулу 391, корисне навантаження змінної довжини або кодові слова 392 та захисний тимчасовий інтервал 393. Преамбула 391 містить або преамбулу для висхідного напрямку, або преамбулу для низхідного напрямку, в залежності від напрямку передачі. У преамбулі для висхідного напрямку преамбула 391 містить заголовок фізичного носія (PMD) 398, заголовок MAC 399 та протокол управління радіоканалом (PDU) даних 400. У преамбулі для низхідного напрямку преамбула 391 містить заголовок формату MPEG 401, заголовок MAC 399 і PDU даних 400. контент корисного навантаження 392 змінної довжини може містити коротке кодове слово 394 або довге кодове слово 397.

Коротке кодове слово 394 містить корисне навантаження 395A, що включає дані A і блоки виправлення помилок 396A, що містять FEC A. Для довгого кодового слова 397 корисне навантаження ділиться на кілька блоків корисного навантаження 395A, 395B і 395C, що містять дані A, дані B і дані C, відповідно, при цьому для кожного блоку корисного навантаження передбачені свої власні блоки перевірки помилок 396A, 396B і 396C, що включають відповідні дані FEC A, FEC B і FEC C. Після контролю помилок передані дані DOCSIS3 містять блоки даних 395A, 395B і 395C для довгого кодового слова та тільки блок даних 295A для короткого кодового слова.

Таким чином, DOCSIS3 забезпечує гнучкість передачі даних по кабельній мережі, використовуючи протокол передачі даних з комутацією пакетів.

Рівень 3 моделі OSI - мережевий (Інтернет) рівень

Як було описано раніше, корисне навантаження даних може бути передано з використанням декількох апаратних конфігурацій фізичного рівня 1 та протоколів інтерфейсу канального рівня 2. Незважаючи на те, що рівні 1 і 2 специфічні для пристроїв, рівень 3 - мережевий рівень - забезпечує незалежну від пристрою форму зв'язку, єдину і інваріантну до фізичної мережі, використовуваної для передачі сигналів і даних. Зв'язок рівня 3 ілюструє рисунок 20, на якому три мережевих пристрої 420A, 420B і 420C, що реалізують функції 423A, 423B або 423C обчислень та зберігання даних, здійснюють спільну роботу з Інтернет-підключенням 421. Таким чином, відповідний пристрою стек зв'язку 422A, 422B і 422C з'єднує пристрої один з одним з використанням мережевого рівня 3 421, яким, за винятком пропріетарних систем, зазвичай є Інтернет.

Щоб забезпечити можливість спільної роботи в мережах з комутацією пакетів, які працюють з різними апаратними платформами, мережами та системами, модель OSI наказує чітко визначений протокол, організований на семи рівнях, відповідно до рисунка 21. Як уже згадувалося раніше, пакети даних або "датаграми" для мереж з комутацією пакетів влаштовані аналогічно матрьошці, коли рівень 1 - пакет фізичного рівня або "кадр" - містить всі інші рівні у своєму корисному навантаженні, включаючи канальний рівень 2, до якого, в свою чергу, вкладено корисне навантаження, що міститься на рівнях 3-7, в тому числі пакети мережевого рівня 4 і т.д.

Розглядаючи цю картину більш детально, слід вказати, що кадр рівня 1 430 містить всі дані фізичного (PHY) рівня, в тому числі електричні, радіо і оптичні сигнали. В дані фізичного рівня 430 вкладена інформація управління доступом до середовища передачі даних або інформація канального рівня 2, що містить заголовок MAC 431, корисне навантаження MAC 432 і нижній колонтитул MAC 433. В корисне навантаження MAC 432 вкладений мережевий (Інтернет) рівень

або пакет IP на рівні 3, що містить заголовок інтернет-протоколу (IP) 434 і корисне навантаження IP 435. У корисне навантаження IP 435 вкладена датаграма транспортного рівня або дані рівня 4, що містять заголовок транспортного рівня 436 і корисне навантаження транспортного рівня 437. Далі до корисного навантаження транспортного рівня 437 вкрито всі дані додатків 438 для прикладних рівнів 5-7 відповідно до моделі OSI, показаної раніше на рисунку 4.

В процесі роботи після прийому пакета даних IP, показаного на рисунку 21, мережевий пристрій і його вбудоване програмне забезпечення інтерпретують дані рівня 1 і рівня 2 і ігнорують будь-яку інформацію, що міститься в корисному навантаженні MAC 432. Мережеве програмне забезпечення, в свою чергу, інтерпретує дані IP адрес, маршрутизації та управління, що містяться в IP рівня 3, але ігнорує вміст корисного навантаження IP 435. Потім програмне забезпечення транспортного рівня 4 інтерпретує дані, що містяться в корисних навантаженнях IP 435, як "датаграму" транспортного рівня, що містить заголовок транспортного рівня 436 і корисне навантаження транспортного рівня 437, забезпечуючи необхідне квітування між сторонами, що обмінюються інформацією, щоб забезпечити надійну передачу IP-пакета. В корисне навантаження транспортного рівня 437 вкрито інформація, що містить прикладні дані 438 для інших додатків верхнього рівня, включаючи пакети, що містять дані для сеансового рівня 5, представницького рівня 6 та прикладного рівня 7. Якщо резюмувати викладене вище, рівні 1 і 2 пов'язані з встановленням фізичного з'єднання та правил для мережевих пристроїв, рівні 3 і 4 пов'язані з ідентифікацією одержувача IP-пакета та підтвердженням його доставки, а рівні 5-8 містять фактичні дані, що передаються в якості корисного навантаження. Відповідно, обладнання та вбудоване програмне забезпечення рівня 1 і рівня 2 не цікавляться вмістом даних, що відправляються і їх додатками, мережеве програмне забезпечення рівня 3 і рівня 4 не стосується того, які фізичні пристрої відправляють пакети, і що міститься в цих пакетах, а рівні 5-7 не піклуються про те, як був відправлений пакет та підтверджений його прийом. Таким чином, в мережах з комутацією пакетів маршрутизацією датаграми невідомого контенту можна управляти, не торкаючись питань, пов'язаних з обладнанням, яке використовується при відправленні пакета або призначенням даних пакета.

Щоб забезпечити сумісність, пакети, відправлені по мережах, використовують стандартизований формат, званий інтернет-протоколом (IP), навіть в тих випадках, коли фактична мережа безпосередньо не підключена до Інтернету. Зв'язок рівня 3 може включати в себе будь-який набір мережевих пристроїв, підключених до загальної мережі з комутацією пакетів, що використовує IP-пакети, включаючи зв'язок через (1) хост-сервер або виділений сервер, підключений безпосередньо до Інтернету; (2) приватні закриті мережі або "інтранет", не підключені до Інтернету або (3) закриті мережі, підключення до Інтернету через "перетворення мережевих адрес" (NAT), що розглядаються нижче в цьому документі. У першому випадку будь-яка IP-адреса, що використовується в Інтернеті, повинна бути зареєстрована та ліцензована для клієнта як ексклюзивна і дійсна інтернет-адреса. В останніх двох випадках IP-адреса застосовується тільки в ізольованій мережі, де вона виділена для використання, і не зареєстрована в якості інтернет-адреси. При спробі використовувати незареєстровані IP-адреси в Інтернеті буде фіксуватися помилка підключення.

Згідно рисунку 22, кожен IP-пакет містить два елементи - заголовок IP 434 і корисне навантаження IP 435. Заголовок IP 434 зазвичай містить дані про одну з двох широко використовуваних версій - "інтернет-протоколу версії 4" (IPv4) або "інтернет-протоколу версії 6" (IPv6). Перші 4 біта заголовка IP 434 в преамбулі заголовка 440 або 444 містять двійковий код інтернет-версії пакету, де 0100 - це поле даних 447, що визначає версію 4, а 0110 - поле даних 448, що визначає версію 6. У разі вибору IPv4, преамбула 440 являє собою поле довжиною 12 байтів, що включає в себе біти номера версії 447, за ним слідує адреса відправника 441 довжиною 4 байта, адреса одержувача 442 довжиною 4 байта та поле опцій 443 довжиною 8 байтів. У разі вибору IPv6, преамбула 444 являє собою поле довжиною 8 байтів, що включає в себе біти номера версії 448, за ним слідує адреса відправника 445 довжиною 16 байтів і адреса одержувача 446 довжиною 16 байтів. На відміну від IPv4, версія 6 не має поля опцій.

Важливо відзначити, що преамбула 440 IPv4 та преамбула 444 IPv6 відрізняються по довжині, змісту та формату та повинні розглядатися окремо. Крім того, поле IP-адреси IPv6 має довжину 16 байтів, що дозволяє однозначно вказувати практично незліченну кількість IP-адрес, тобто 2¹²⁸. Для порівняння, в IPv4 його довжина становить 4 байти і дозволяє вказувати тільки 2³² адреси. Через обмежену кількість комбінацій в IPv4 потрібна інша інформація для ідентифікації та відділення мереж від клієнтів, як зазначено в преамбулі 440. Для IPv6 немає необхідності в такому розмежуванні. Більшість сучасних мереж і IP-маршрутизаторів в даний час підтримують як IPv4, так і IPv6.

Інтернет-протокол IPv4 - Розглянемо більш докладно структуру пакета даних датаграми 450 IPv4. На рисунку 23 показано двовимірне графічне представлення, впорядковане за часом послідовно зліва направо по стовпцях та зверху вниз по рядках. Зокрема тут час, що відповідає кожному рядку, представляється байтами або октетами від 0 до 3 (або бітами від 0 до 31), а кожен рядок з верхнього до нижнього відзначений октетом зміщення, при цьому найвищий рядок має позначку "0", за нею слідує рядок з відміткою "4", потім "8", потім "12" і т.д. Щоб правильно прочитати послідовність даних з датаграми 450, потрібно починати зчитувати пакет в рядку октету зміщення з відміткою "0", де зліва направо перші відправлені або отримані дані містять преамбулу 451, що включає вищезгадане поле "Версія", за яким слідує поле "IHL" (Internet Header Length - розміри заголовка), "DSCP" (Differentiated Services Code Point - точка коду диференційованих послуг), "ECN" (Explicit Congestion Notification - явне повідомлення про перевантаження) і "Загальна довжина". Відразу ж після цього зчитуються дані щодо зміщення наступного рядка, зазначені октетним рядком зміщення "4", що містить поля, що позначені "Ідентифікація", "Прапори", "Зсув фрагмента". Нарешті, останній рядок з відміткою "8" в преамбулі 450 містить поля "Час життя", "Протокол" і "Контрольна сума". Після преамбули в датаграму розташовується вихідна IP-адреса відправника довжиною 4 байта, IP-адреса одержувача довжиною 4 байта і рядок, позначений як 20 в стовпці "Зсув, Октет" - поле "Опції". Останнє поле в датаграмі 450 містить пакет корисного навантаження 435 змінної довжини. Незважаючи на те, що в прикладі показана довжина корисного навантаження 4 байта, ця довжина є змінною.

У таблиці 451 представлені короткі зведені дані про інформацію, що міститься в полях датаграми IPv4. Як було зазначено раніше, в чотирьохбітному полі версії встановлюється двійковий код 0100 для версії 4 інтернет-протоколу. Поле IHL вказує кількість 32-бітних слів у заголовку 434 IP, довжину пакета IPv4 450 без урахування корисного навантаження 435 в діапазоні значень від 20 до 62 байтів. DSCP є поле довжиною 6 біт, що визначає диференційовані служби для контролю якості обслуговування. ECN являє собою поле довжиною 4 біта для явних повідомлень про перевантаження (ECN), що описують стан завантаженості мережі. Поле "Загальна довжина" визначає загальну довжину датаграми пакета IPv4, що включає в себе як заголовок IP 434, так і корисне навантаження IP 435, при цьому його мінімальна довжина становить 20 байтів, а максимальна - 65 535 байтів. Максимальна довжина пакета може бути обмежена меншими датаграмами по протоколу каналного рівня 2 для конкретного фізичного середовища. Поле "Ідентифікація" довжиною 2 байта однозначно ідентифікує групу фрагментів однієї IP-датаграми, щоб дозволити повторне складання пакета з прийнятими в неправильному порядку сегментами спільно з полями "Прапори" довжиною 3 біта і "Зміщення прапорів" довжиною 13 бітів, що використовуються для управління фрагментацією пакетів. Поле TTL (Time To Live - Час життя) довжиною 1 байт обмежує час життя датаграм в мережі, щоб уникнути "безсмертя" - наявність пакетів, які не можуть бути доставлені в пункт призначення, але й ніколи не припиняють своє існування. Поле TTL вказує максимальну кількість маршрутизаторів, через які може пройти будь-який конкретний пакет, перш ніж він буде відкинутий через неможливість його доставки. Кожен раз, коли пакет проходить маршрутизатор, лічильник TTL зменшує своє значення на одиницю.

Поле 460 - поле "Протокол" довжиною 1 байт - описує тип даних, що містяться в корисному навантаженні 435 пакета IPv4. У деяких випадках ці дані містять конкретні інструкції, наприклад, для перевірки стану мережі або затримки поширення, які повинні виконуватися як пакет рівня 3, а в інших випадках корисне навантаження може бути ідентифіковано як таке, що містить транспортний протокол рівня 4, який використовується для управління передачею та підтвердження доставки, в тому числі стандартні транспортні протоколи ICMP, IGMP, TCP, UDP або інші пропріетарні формати. По суті, поле протоколу є опис датаграми рівня 4 в пакеті IPv4 рівня 3, тісно прив'язуючи рівень 3 моделі OSI до рівня 4 в інтернет-протоколі. Поле контрольної суми заголовка використовується для забезпечення правильності даних заголовка, щоб пакет не доставлявся не тому одержувачу. Воно містить 16-розрядну контрольну суму, яка використовується для виявлення помилок та відкидання даних. У сукупності вищезазначені поля утворюють преамбулу 440 пакета IPv4.

Наступні два поля - IP-адреса відправника та IP-адреса одержувача - мають довжину 4 байта і можуть бути представлені в декількох форматах. Традиційний формат, званий форматом "точка - десяткове число", містить чотири десяткових числа, між якими ставлять крапку, наприклад, 192.0.2.235 або в шістнадцятковій формі з точкою як роздільник - 0xC0.0 × 00.0 × 02.0xEB, де кожному байту, тобто октету, передує 0x, при цьому він індивідуально перетворюється в шістнадцяткову форму. 32-розрядна адреса також може бути перетворена в десятковий еквівалент 3221226219 або в одне шістнадцяткове число 0xC00002EB шляхом

конкатенації октетів з шістнадцятирічного формату з точкою як роздільник. Додаткову інформацію про формати адрес IPv4 можна отримати, перейшовши за посиланням <http://en.wikipedia.org/wiki/IPv4> або за іншими аналогічними посиланнями. Поле "Опція" довжиною 4 байта (активне лише тоді, коли в полі "IHL" встановлено значення від 6 до 15) використовується рідко, так як створює загрози для безпеки.

Інтернет-протокол IPv6 - Вичерпання адресного простору для існуючих IP-адрес підштовхнуло до створення нового набору IP-адрес, званого версією 6 інтернет-протоколу. Конструкція пакета даних датаграми IPv6 453, показана на рисунку 24, як і її попередниці, версії 4, містить два елементи: заголовок IP 434 і корисне навантаження IP 435, при цьому слід врахувати, що заголовок значно простіше і IP-адреси значно довше. Зокрема, преамбула IPv6 444 має довжину всього 8 байтів, тоді як адреси IPv6 445 і 446 мають довжину 16 байтів.

У таблиці 454 представлені короткі зведені дані про інформацію, що міститься в полях датаграми IPv6. Як було зазначено раніше, в чотирьохбітному полі версії встановлюється двійковий код 0110 для версії 6 інтернет-протоколу. Поле "Клас трафіку" довжиною 1 байт включає в себе підполе довжиною 6 біт, що визначає диференційовані служби, та поле довжиною 2 біта для явних повідомлень про перевантаження (ECN), аналогічні версії 4. Поле "Мітка потоку" довжиною 20 біт мінімізує фрагментацію, підтримуючи шлях даних, щоб уникнути реорганізації в додатках реального часу. Поле "Довжина корисного навантаження" величиною 2 байта визначає довжину корисного навантаження 435 в байтах (октетах). Поле "Наступний заголовок" 460 довжиною 1 байт вказує тип вмісту в корисне навантаження 435. Як і поле "Протокол" в IPv4, поле "Наступний заголовок" в IPv6 по суті надає дані про зміст корисного навантаження IP 435. У деяких випадках цей вміст являє собою дію, наприклад, перевірити затримки в мережі, і містить дані рівня 3. У інших випадках цей вміст являє собою транспортний протокол рівня 4, який використовується для управління передачею та підтвердження доставки, в тому числі стандартні транспортні протоколи ICMP, IGMP, TCP, UDP або інші пропріетарні формати. Як і "Час життя" в IPv4, поле "Ліміт переходів" довжиною 1 байт в пакеті IPv6 вказує максимальну кількість маршрутизаторів, які пакет може пройти, перш ніж він буде відкинтий як "безсмертний". Кожен раз, коли пакет проходить маршрутизатор, цей лічильник зменшує своє значення на одиницю.

Наступні два поля, кожне з яких має довжину 16 байтів, визначають IP-адресу відправника 445 і IP-адресу одержувача 446. Як було зазначено раніше, метою використання більш довгих IP-адрес є подолання нестачі IP-адрес, що виникає в IPv4. Ця проблема ілюструється рисунком 25 для IP-адрес 469 шляхом зіставлення трьох класів 4-байтових адрес IPv4 з адресою IPv6 458 довжиною 16 байтів без розбивки на класи. Оскільки адреса IPv6 здатна утворювати 2^{128} або $3,403 \times 10^{38}$ унікальних комбінацій, немає необхідності ділити адреси на класи, спеціально призначені для мереж і клієнтів. Навпаки, через обмежену кількість комбінацій, доступних в IPv4, адреси були розділені на "класи", та сьогодні вони від класу А до класу С все ще широко поширені.

Згідно рисунку, клас А містить поле мережі 456А довжиною 1 байт та поле клієнта 457А довжиною 3 байта, має адреси IPv4 від 0.0.0.0 до 127.255.255.255, що забезпечують підтримку 128 мереж і 16 777 216 (приблизно 224) клієнтів. Користувачем класу А може бути будь-який великий інтернет-провайдер, телефонна компанія або провайдер відеоконтенту. Адреси класу В містять поле мережі довжиною 2 байта, позначене 456В, та поле клієнта довжиною 2 байта, позначене 457В, що має адреси IPv4 від 128.0.0.0 до 191.255.255.255, що забезпечують підтримку 16 384 (близько 214) мереж і 65 536 (близько 216) клієнтів. Користувачами класу В можуть бути компанії з великою кількістю сайтів. Адреси класу С містять поле мережі довжиною 3 байти, позначене 456С, та поле клієнта довжиною 2 байта, позначене 457С, що має адреси IPv4 від 192.0.0.0 до 223.255.255.255, що забезпечують підтримку 2 097 152 (близько 221) мереж і 256 (т. е. 28) клієнтів. Користувачами класу С зазвичай є суб'єкти малого бізнесу.

Під час маршрутизації пакета через мережу або Інтернет обробка кожного поля в IP-заголовку 434 відбувається на основі необхідного знання. Наприклад, кожен маршрутизатор повинен знати версію IP, довжину пакета і контрольну суму пакета для контролю помилок. Точно так же час переходів або час життя обов'язково обробляються проміжними маршрутизаторами для відкидання "безсмертних" пакетів. Однак проміжні маршрутизатори не повинні інтерпретувати кожне поле заголовка IP 434. Зокрема, поле 460 - поле "Протокол" в IPv4 або "Наступний заголовок" в IPv6 мають значення тільки для IP-адрес відправника і одержувача. Проміжні маршрутизатори не повинні знати зміст корисного навантаження IP 435 і, отже, не повинні обробляти цю інформацію. Тільки тоді, коли пакет, нарешті, досягає IP-адреси свого одержувача, призначений пристрій або сервер одержувача зчитують значення поля 460 у заголовку 434 IP, щоб інтерпретувати, які дані вкладені в корисне навантаження IP 435. Згідно

рисунку 26, будь-яке дійсне значення в полі 460 може призвести до дії, яка може бути застосована до корисного навантаження мережевого рівня - рівня 3 - або, в іншому випадку, до корисного навантаження транспортного рівня - рівня 4. У разі якщо код, що міститься в полі 460, не розпізнає IP-адресою одержувача, сервер чи пристрій одержувача будуть відкидати цей пакет як такий, що містить помилки.

У тих випадках, коли поле 460 містить корисне навантаження мережевого рівня - рівня 3 - як інструкції, що підлягають виконанню, корисне навантаження IP 435 вказує мережі задачу, яка повинна виконуватися. Наприклад, коли поле 460 містить еквівалент десяткових чисел 1 або 2 в полях протоколу або наступного заголовка 461 або 462, корисне навантаження IP 435 буде містити відповідні інструкції для мережевих утиліт ICMP або IGMP (протокол управління групами користувачів в мережі Інтернет), відповідно. Якщо поле 460 замість цього містить еквівалент десяткового числа 6 в поле протоколу або наступного заголовка 463, корисне навантаження IP 435 буде містити дані 475 для корисного навантаження, що використовує транспортний протокол TCP рівня 4. Аналогічно, якщо поле 460 замість цього містить еквівалент десяткового числа 6 в поле протоколу або наступного заголовка 464, корисне навантаження IP 435 буде містити дані 476 для корисного навантаження, що використовує транспортний протокол UDP рівня 4. Корисне навантаження рівня 4 буде розглядатися в наступному розділі цього документа. Існують і інші менш поширені та пропріетарні коди. Якщо поле 460 містить код протоколу або наступного заголовка, який є стандартизованим зареєстрованим кодом, то загальнодоступні мережі, принаймні, теоретично, повинні відповідним чином реагувати на код та правильно інтерпретувати корисне навантаження. У тих випадках, коли код є пропріетарним, тільки пропріетарні мережі і налаштований маршрутизатор можуть інтерпретувати цей код та виконувати відповідні дії належним чином.

У разі, коли поле 460 містить еквівалент десяткового числа 1 в як поле протоколу або наступного заголовка, корисне навантаження IP 435 містить конкретну мережеву утиліту 435, звану ICMP або "протокол управління повідомленнями в мережі Інтернет", яка використовується такими мережевими пристроями, як сервери, маршрутизатори, точки доступу і т.д. для доступу до затримок поширення мережі; для вказівки, що запитувана послуга недоступна, або визначення, що маршрутизатор або хост не можуть бути досягнуті. Його призначений ідентифікатор протоколу або наступного заголовка, десяткове число 1, відрізняється від UDP і TCP в тому, що ICMP зазвичай не використовується для обміну інформацією між системами або додатками кінцевого користувача, за винятком випадку, коли виконується певна мережева діагностика. Згідно рисунку 26 для IP-пакета, що відповідає даним 461, пакет ICMP містить заголовок, що складається з чотирьох частин, якими є тип 465, код 466, контрольна сума 467 і решта заголовку ICMP 468, за якою слідує дані ICMP 469.

Поля "Тип" 465 і "Код" 466 спільно полегшують доставку різних керуючих повідомлень. Можна уточнити, що тип 3 керуючих повідомлень означає, що одержувач IP недоступний, при цьому код вказує, чому він недоступний, наприклад: код 0 означає, що була недоступна мережа одержувача; код 1 означає, що був недоступний хост одержувача; код 3 означає, що був недоступний порт одержувача; а код 9 означає, що доступ до мережі адміністративно заборонений і т.д. Тип 5 означає, що пакет може бути переадресовано, при цьому код 0 означає перенаправлення датаграми для мережі, код 1 означає перенаправлення датаграми для хоста і т.д. Тип 8 - "ехо-запит", за яким слідує тип 0 - "ехо-відповідь", спільно виконують важливу та відому функцію "ping", аналогічну акустичному зондуванню підводного човна гідролокатором, щоб перевірити затримку поширення мережі. Можна вказати ще кілька важливих функцій: "трасування маршруту" (код 30), "запит доменного імені" (код 37), "відповідь на запит доменного імені" (код 38), "запит мітки часу" (код 13), "відповідь на запит мітки часу" (код 14). Для задач доставки код 11 означає, що "час доставки перевищено"; код 12 означає "невірний IP заголовок", а код 4 або "відключення джерела" використовується в разі контролю перевантаження. Контент даних ICMP 469 може містити повідомлення або може використовуватися просто для завантаження мережі більшими пакетами, щоб дослідити, зокрема, чи можуть виникати проблеми при передачі великого обсягу корисного навантаження.

Також на рисунку 26 показано, що, коли поле 460 містить еквівалент десяткового числа 2 в полі протоколу або наступного заголовка, корисне навантаження IP 435 містить конкретну мережеву утиліту 435, звану IGMP (Internet Group Management Protocol - протокол управління групами користувачів в мережі Інтернет). На відміну від протоколу ICMP, використовуваного при діагностиці мереж IPv4 і IPv6, IGMP використовується тільки при мультимовленні IPv4 для мережевих додатків "один-до-багатьох", таких як ігри або онлайн-трансляції. Однак термін IGMPv4 не використовується, оскільки IGMP зобов'язаний своїм походженням більш раннім реалізаціям Інтернету. В даний час підтримуються тільки протоколи IGMPv2 і IGMPv3. Крім того,

в IPv6 мультимовлення здійснюється через вбудований в ICMPv6 протокол визначення одержувачів мультимовленнєвих запитів (Multicast listener discovery), а не безпосередньо через чисте вкладення IGMP. Пакет IGMP містить заголовок, що складається з чотирьох полів, якими є "Тип" 470, "Максимальний час відповіді" 471, "Контрольна сума" 472 і "Адреса групи IGMP" 473, за якими слідують дані IGMP 474.

У IGMP поле "Тип" 470 описує природу пакета як команди "Запит членства", "Звіт про членів групи" або "Покинути групу"; "Максимальний час відповіді" (Maximum Response Time) 471, або максимальний час відповіді, встановлює максимальний час для отримання звіту до 100 мс, а контрольна сума 472 являє собою додаток до одиниці 16-розрядної суми всього пакета IGMP. Для широкомовної передачі IGMPv2 відправляє пакет IGMP і дані IGMP зі своїм корисним навантаженням 474 за адресою групи IGMP 473 відповідно до настройки типу 470, де "загальний запит" здійснює мультимовлення всім хостам, наприклад, 224.0.0.1, а команда "Покинути групу" аналогічно відправляє повідомлення до всіх маршрутизаторів, наприклад, 224.0.0.2. У "запиті певної групи" та "звіті щодо членів групи" IGMPv2 в інформаційному обміні буде задіяна тільки запитувана група. У IGMPv3 можливий більш повний запит членства, визначає всі підключені боку.

Інші датаграми можуть містити не тільки ICMP і IGMP, а й пропрієтарні протоколи, де IP-адреси відправника та одержувача повинні передбачати зв'язок з використанням унікального формату, в іншому випадку корисне навантаження 435, як правило, буде містити дані, відповідні протоколам TCP або UDP транспортного рівня 4.

Рівень 4 моделі OSI - транспортний рівень

Функція транспортного рівня 4 моделі OSI ілюструється рисунком 27, де три мережевих пристрої 480A, 480B і 480C, що містять блоки 483A, 483B і 483C для виконання обчислень та зберігання даних з відповідними стеками зв'язку 482A, 482B і 482C, спільно використовують загальну мережу 481. Транспортний рівень гарантує, що зв'язок 484 здійснюється тільки між стеком зв'язку 482A в пристрої A та стеком зв'язку 482B в пристрої B. Завдання транспортного рівня полягає в управлінні зв'язком між цими двома мережевими пристроями та забезпеченні контексту для типу прикладних даних, які повинні бути доставлені пакетами IP, та послуги, яка повинна бути виконана. Таким чином, по суті мережу 481 рівня 3 моделі OSI дозволяє підключати будь-яку комбінацію пристроїв, а транспортний рівень рівня 4 моделі OSI, забезпечує зв'язок двох конкретних пристроїв.

Двома переважаючими транспортними протоколами, що використовуються в даний час, є TCP і UDP. В "протоколі управління передачею даних" (TCP) встановлення зв'язку між пристроями гарантується процедурою підтвердження встановлення зв'язку, яка підтверджує, що IP-пакет був надійно та точно доставлений по мережі з комутацією пакетів перед відправкою наступного пакета. Завдяки передбаченій в TCP процедурі підтвердження встановлення зв'язку, "підключення" може гарантуватися навіть в системі зв'язку з комутацією пакетів "без організації з'єднання", що містить локальну мережу, інтрамережу або загальнодоступний Інтернет. TCP забезпечує надійну, яка контролює наявність помилок доставку в належному порядку послідовності цифрових байтів з високою точністю, але без гарантії своєчасної доставки. TCP використовується для передачі нечутливої до часу доставки корисного навантаження, в тому числі великої кількості різних комп'ютерних програм, файлів, текстових, відео і голосових повідомлень, включаючи електронну пошту, передачу файлів, веб-браузери, функції віддалених терміналів та захищені оболонки. Для чутливої до часу доставки корисного навантаження кращі інші протоколи, більш підходящі для додатків реального часу, наприклад, UDP.

Протокол управління передачею даних (TCP) - TCP, що працює на транспортному рівні 4 моделі OSI, займає проміжний рівень між мережевим (Інтернет) рівнем 3, та верхніми прикладними рівнями. При передачі IP-пакетів TCP може коригувати непередбачувану поведінку мережі внаслідок перевантаження мережі, відкидання пакетів, врівноваження навантаження трафіку і доставки в неправильному порядку. TCP виявляє ці та інші проблеми, запитує повторну передачу втрачених даних у міру необхідності, відновлює організацію даних, прийнятих в неправильному порядку, і навіть пом'якшує наслідки помірного перевантаження мережі, наскільки це можливо. IP-пакети, що передаються транспортним рівнем TCP, можуть називатися датаграмами TCP/IP. Під час передачі пакета використовується таймер для контролю часу доставки. У разі, коли цей час закінчується до доставки пакета, виконується запит на повторну передачу пакета. Пакети TCP вкладені в корисне навантаження IP-пакетів. Отримані пакети TCP буферизуються та повторно збираються для доставки додатків.

Щоб ідентифікувати додаток або послугу, для яких призначений пакет TCP, TCP використовує цифрову ідентифікацію під назвою "порт". Порт - це номер, який використовується для унікальної ідентифікації передачі по мережі, шляхом вказівки як хоста, так і задіяної служби.

Порти використовуються протоколом TCP або UDP для розпізнавання різних IP-служб і додатків, таких як веб-служба (HTTP), поштова служба (SMTP) та служба передача файлів (FTP). Пристрої зв'язку використовують комбінацію IP-адрес рівня 3 та портів рівня 4 для управління обміном інформацією з фізичної мережі, що містить фізичний рівень 1 і каналний рівень 2, з верхніми прикладними рівнями моделі OSI, починаючи з рівня 5 та вище.

Кожен пакет TCP 500, показаний на рисунку 28А, містить заголовок TCP 506 і корисне навантаження TCP 507. Докладні дані про функції заголовка TCP 506 зведені в таблицю 508, показану на рисунку 28В, де заголовок TCP 506 містить порт відправника 501, порт одержувача 502, порядковий номер 503, номер підтвердження 504, а також поля "Зміщення", "Резервування", "Прапори", "Розмір вікна", "Показчик терміновості" і "Опції". Він також містить контрольну суму 505 для підтвердження цілісності пакета. Порядковий номер 503 використовується для відстеження порядку надходження декількох пакетів та залежить від стану прапора SYN в полі "Прапори" заголовка TCP 506. Поле "Підтвердження" використовується в процесі підтвердження встановлення зв'язку. Якщо для прапора ACK в полі "Прапори" заголовка TCP 506 встановлено значення двійковій одиниці, поле підтвердження містить черговий порядковий номер, очікуваний приймачем, після прийому якого підтверджується отримання всіх наступних пакетів.

Дані поля "Зміщення" визначають розмір заголовка TCP 506, тобто довжину заголовка від початку датаграми TCP 500 до початку корисного навантаження TCP 507, яке вказується кількістю 2-байтових (32-розрядних) слів в діапазоні від 5 до 15 слів двобайтової довжини. Зарезервовані біти в даний час не використовуються. Поле прапорів містить дев'ять довічних прапорів, пов'язаних з приховуванням інформації, перевантаженням, терміновістю, підтвердженням пакета, функцією "push-функцією", скиданням з'єднання, управлінням послідовністю дій та відсутністю даних від відправника. Розмір вікна визначає максимальну кількість байтів, яке відправник наказує отримати в одному пакеті. Контрольна сума являє собою контрольну суму довжиною 2 байта (16 біт) для контролю помилок заголовка TCP 506 і корисного навантаження TCP 507. Якщо для прапора URG встановлено значення двійковій одиниці, в полі "Показчик терміновості" вказується останній байт даних, що підлягає терміновій відправці.

У зв'язку з комутацією пакетів на основі TCP/IP підтвердження встановлення зв'язку є ключовою функцією для забезпечення цілісності даних. Згідно рисунку 29 в момент часу $t=0$ ноутбук 510 відправляє пакет TCP/IP веб-сервера 531, відправляючи заголовок TCP 512А, корисне навантаження TCP 513А та час переміщення 514А, для яких потрібна загальна тривалість Δt_a , за ними слідує підтвердження від веб-сервера 511 ноутбуку 510, що містить заголовок TCP 512В і нульове поле 513В, для яких потрібна загальна тривалість Δt_b . Загальна величина об'єднаного інтервалу $t_1 = \Delta t_a + \Delta t_b$ є мінімально необхідний час для відправки та підтвердження пакета TCP/IP, що приблизно в два рази перевищує час початкової доставки пакета. Після цього, та тільки після цього може передаватися другий пакет, що містить заголовок TCP 512С і корисне навантаження TCP 513С. У разі якщо пакет пошкоджений або втрачений, цей пакет повинен бути відправлений та підтверджений повторно, що збільшує тривалість доставки з t_1 до $2t_1$. Якщо пакет потребує повторення передачі "n" раз, тривалість передачі тільки одного пакета складе nt_1 . Мінлива тимчасова затримка при використанні транспортного протоколу TCP створює серйозні проблеми при передачі чутливих до часу доставки пакетів, наприклад, відео або VoIP.

Таким чином, пакети TCP/IP мають такі характеристики:

- Надійність - TCP/IP гарантує доставку за рахунок керування підтвердженням, контролю помилок, запитів на повторну передачу і функцій перерви.

- "Ваговитість" - TCP/IP використовує великий пакет транспортного рівня з довгим складним заголовком та вимагає щонайменше трьох пакетів тільки для того, щоб встановити з'єднання між хостом і клієнтом.

- Мінлива/низька швидкість передачі даних - Через підтвердження встановлення зв'язку швидкість передачі даних TCP/IP змінюється та стає значно менше, ніж для UDP, що робить протокол TCP непривабливим для таких додатків реального часу, як передача відеоконтенту та VoIP.

- Правильний порядок - TCP буферизує та реорганізовує всі пакети, прийняті в неправильному порядку.

- Контроль перевантаження - TCP надає кілька функцій управління перевантаженням, відсутніх в UDP.

- Контроль помилок - Пакети TCP/IP перевіряються на цілісність, якщо вони приймаються та передаються повторно, якщо будь-які пакети відкидаються або надходять в пошкодженному вигляді.

Протокол датаграм користувача(UDP) - В якості альтернативи протоколу управління передачею даних (TCP), (UDP) використовує режим передачі без встановлення з'єднання, з мінімальним складом протоколу: без встановлення з'єднання і без підтвердження доставки пакета. Володіючи чутливістю до нестабільності, властивій мережі, UDP не пропонує ніяких підтверджень доставки, а також ніякого упорядкування пакетів і дублюючого захисту. Тим не менш, він використовує контрольні суми для підтвердження цілісності даних. UDP найкраще підходить для додатків, чутливих до часу передачі, або для цілей, коли контроль та виправлення помилок або не є необхідними, або виконуються постфактум в додатку, виключаючи витрати ресурсів на таку обробку на мережевому рівні.

Пакет UDP 529, показаний на рисунку 30, містить заголовок UDP 520 і корисне навантаження UDP 524. Заголовок UDP 520, описаний в таблиці 525, містить лише чотири поля: адреса порту відправника 521 довжиною 2 байта, адреса одержувача 521 довжиною 2 байта, поле "Довжина" 523 і "Контрольна сума" 523. Адреси портів UDP використовують той же формат, що і пакети TCP/IP. Поле довжини пакета UDP 523 в IPv6 змінюється по довжині від мінімального значення 8 байтів до максимального значення 65 535 байтів. З практичних причин найбільша довжина контрольної суми в протоколі IPv4 обмежена трохи меншим значенням 65 507 байтів.

Контрольна сума 523 довжиною 2 байта використовується для виявлення помилок загальної довжини корисного навантаження UDP 524 разом з даними від заголовка UDP 520, алгоритмічно перетвореного в псевдозаголовок для включення IP-адрес і інших полів, запозичених з заголовка IP. Псевдозаголовок ніколи не існує в датаграмі в явному вигляді, але створюється, тобто алгоритмічно синтезується з даних, наявних в заголовку IP та заголовку UDP, тільки для цілей контролю помилок. Формат псевдозаголовка та значення контрольної суми відрізняються для пакетів UDP на основі IPv4 і IPv6. У IPv4 функція контрольної суми є необов'язковою, тоді як в IPv6 її використання є обов'язковим. Коли вона не використовується, це поле заповнюється цифровим значенням 0. Після заголовка UDP 520 слідує корисне навантаження UDP 524, довжина якої в IPv4 є змінною від 0 до 65 507 байтів.

Таким чином, на транспортному рівні 4 для передачі IP-пакета по мережі зв'язку з комутацією пакетів можуть використовуватися як UDP, так і TCP/IP. Пакети UDP мають наступні характеристики:

- Ненадійність - UDP не гарантує доставку і нечутливий до втрати пакетів. У UDP відсутні механічні засоби для ідентифікації втрати пакетів, для запиту повторної передачі або для контролю стану перерви під час передачі.

- Легковажність - UDP використовує невеликий транспортний рівень з заголовком мінімального розміру, в якому відсутні багато функцій TCP та пов'язані з ними властивості пакета.

- Висока швидкість - Внаслідок їх невеликого розміру, пакети UDP можуть передаватися швидко і не вимагають квітування доставки або повторної передачі втрачених або пошкоджених пакетів. Мінімальна швидкість передачі даних удвічі вище, ніж у TCP, та в чотири рази вище, ніж у випадках, пов'язаних з повторною передачею пакетів TCP. У нестабільних мережах запит на повторну передачу може повністю зупинити доставку TCP-пакетів.

- Невпорядкованість - порядок, в якому приймаються пакети, може не відповідати порядку, в якому вони були відправлені. Додаток має бути досить інтелектуальним, щоб забезпечити правильний порядок проходження пакетів.

- Відсутність контролю перевантаження - навіть якщо не враховувати, що це є наслідком невеликих ресурсів, що виділяються на пакет, UDP не виключає можливість перевантаження, якщо тільки функція контролю перевантаження не реалізована на прикладному рівні.

- Контроль помилок - Пакети UDP перевіряються на цілісність тільки після їх отримання. При виявленні помилок ці пакети відкидаються без будь-якого запиту на повторну передачу.

Використання портів рівня 4 - Порти грають важливу роль в реалізації рівня 4 - транспортного рівня - в мережі зв'язку з комутацією пакетів. Серед інших переваг, порти допомагають ідентифікувати додатки або служби, що надаються сервером або пристроєм, вони допомагають вирішити питання взаємодії декількох користувачів з одним та тим же сервером, не змішуючи комунікації окремого клієнта, вони забезпечують засоби підтримки повнодуплексного зв'язку з використанням різних пар портів для обміну "хост-клієнт" та "клієнт-хост", і допомагають полегшити роботу NAT, щоб збільшити кількість доступних IP-адрес для

користувачів, одночасно обмежуючи вартість і кількість необхідних підключень безпосередньо в Інтернеті.

Приклад обміну датаграмами хоста і клієнта ілюструється рисунком 31А, де клієнтський пристрій 526В - планшет або ноутбук - запитує веб-сторінку з хоста 526А, яким, як правило, є веб-сервер. В процесі обміну клієнт 526В відправляє ІР-датаграму, що містить заголовок ІР рівня 3, що має ІР-адресу 527В з числовим значенням "ІР-адреси В", на хост-сервер за ІР-адресою 527А, має числове значення "ІР-адреси А". Вкладений в корисне навантаження датаграми рівня 3 клієнт також відправляє заголовок транспортного рівня 4 530, що містить свій власний номер порту відправника 528А зі спеціальним значенням 9999. Запит порту відправляється на порт 80 хоста - зарезервований HTTP-порт 528А, який використовується для завантаження веб-сторінок в веб-браузер. Тому, незважаючи на те, що номер запитуваного порту 9999 призначається довільно спеціальним чином як номер наступного відкритого порту, порт одержувача 80 має конкретне значення для запитуваної послуги, а саме, запиту веб-сторінки.

Проста версія ІР-датаграми, використовуваної для цього запиту веб-сторінки, ілюструється в нижній частині рисунка 31А і містить заголовок ІР рівня 3 529, заголовок транспортного рівня 4 530 і корисне навантаження 536 ІР-пакета. У заголовку 529 ІР рівня 3 ІР-адреса відправника 531 має числове значення "ІР-адреси В", а ІР-адреса одержувача 532 має значення "ІР-адреси А". У заголовку 530 транспортного рівня 4 числове значення номера порту відправника 533 одне "9999", а числове значення номера порту одержувача 534 інше "80". Корисне навантаження 536 ІР-пакета включає поле корисного навантаження (даних) 535, що містить дані прикладних рівнів 5-7.

На рисунку 31В показана відповідь на запит клієнта на надання послуг. Відповідно до рисунку, всі напрямки стрілок змінюються на протилежні, а всі ІР-адреси відправника і одержувача та номери портів змінюються один з одним, в порівнянні з попереднім рисунком. В процесі обміну ІР-датаграма, що містить заголовок ІР 537 рівня 3, відправляється з ІР-адреси відправника 531 з числовим значенням "ІР-адреси А" на ІР-адресу одержувача 532, що має числове значення "ІР-адреси В". Вкладений в датаграму рівня 3 заголовок транспортного рівня 4 538 включає в себе порт відправника 533, що має числове значення номера порту "80" та порт одержувача 534, що має числове значення номера порту "9999". Вбудований в корисне навантаження 539 ІР-пакета відповідь на запит надання послуг являє собою корисне навантаження (дані) 536, яка може містити HTML-код для створення веб-сторінки.

Таким чином, незважаючи на те, що деякі номери портів вільні та призначаються в міру необхідності при виборі сервера, інші номери зарезервовані для використання в пакетах UDP, в пакетах TCP або в обох з них. Список зазвичай офіційно зарезервованих номерів портів наведено на рисунку 31С, в їх число входять відомий порт 80 для перегляду веб-сторінок HTTP з використанням тільки TCP, порт 20 для передачі файлів, порт 23 для роботи з telnet, порт 110 для отримання електронної пошти по протоколу POP3 тільки для TCP, порт 220 для отримання електронної пошти по протоколу IMAP3, а також порти для великої кількості захищених версій протоколів, таких як HTTPS, IMAPS, FTP через TLS/SSL і т.д. Однак недавно було виявлено, що безпека SSL - внутрішнього методу захисту транспортного рівня - має вразливі місця для певних видів атак, як зазначено в одному з заголовків на початку цього документа. Порт 7, який використовується для функцій "echo" і "ping" рівня 4, був в значній мірі замінено функцією ICMP рівня 3.

У таблиці, наведеної на рисунку 31D, показані діапазони номерів портів і дані про їх використання. Згідно рисунку, номери зарезервованих портів зазвичай знаходяться в діапазоні від 0 до 1023 для "системних портів", а номери портів більше 49 152 зазвичай вільні і доступні. У проміжному діапазоні номерів портів від 1024 до 49 151 великі блоки вільні і доступні для динамічного розділення портів, але деякі зарезервовані порти там також присутні. Широко поширена практика, коли великі корпорації можуть повідомляти про вибір виділених портів у своєму програмному забезпеченні, але не реєструвати ці номери портів офіційно. Незважаючи на це, "офіційні" та зарезервовані номери портів, хоча вони і не строго контролюються, знаходять широку підтримку, тому що компанії хочуть забезпечити сумісність своїх систем та програмного забезпечення з Інтернетом та з продукцією інших підприємств.

Порти також використовуються для полегшення роботи "брандмауера", запобігання або, принаймні, заборони несанкціонованого доступу до комп'ютера, сервера або пристрою для конкретної послуги. Наприклад, будь-який сервер, розташований в інтрамережі, тобто в приватній мережі, розташованій за перетворенням мережевих адрес (NAT) або захищений виділеним блоком мережевої безпеки, може бути обмежений конкретними типами запитів на обслуговування, ініційованих з Інтернету. Наприклад, міжмережевий екран може бути

налаштований на блокування запитів через порт 80, відключення запитів HTTP-сервісу та запобігання завантаження веб-сторінок з Інтернету. В якості альтернативи, міжмережевий екран може бути налаштований на дозвіл запитів на надання послуг з Інтернету тільки через порт 25, без надання таких прав для інших портів. У таких випадках міжмережевий екран дозволяє

5 запити на обслуговування для протоколу SMTP (простий протокол передачі пошти), дозволяючи відправляти електронну пошту з інтрамережі в Інтернет і назад, але блокує всі інші типи транзакцій. Такі суворі заходи з боку брандмауера створюють проблеми, тому що додатковий захист, блокує багато інших видів транзакцій, не дозволяючи співробітникам та провайдерам на місцях отримувати доступ до важливої інформації, необхідної для виконання

10 своєї роботи.

Інший напрямок використання портів - допомогти затримати процес утворення нестачі IP-адрес в IPv4. Замість того щоб призначати для кожного персонального пристрою кілька виділених IP-адрес, Інтернет-провайдери(ISP), такі як провайдери кабельних мереж, загальнодоступних мереж Wi-Fi, стільникового телефонного зв'язку та інші, мають можливість

15 організувати динамічне циклічне використання IP-адрес та використовувати IP-адреси приватних мереж для зв'язку між своїм інтернет-шлюзом та своїми приватними клієнтами. Таким чином, одна IP-адреса в Інтернеті може обслуговувати до 65 534 користувачів підмережі класу B або 254 користувачів підмережі класу C за умови, що пропускна здатність висхідного каналу досить висока для підтримки трафіку.

Пристрій, який виконує двонаправлене перетворення і інформаційний обмін через цю одну IP-адресу з багатьма IP-адресами, називається "перетворювачем мережевих адрес" (NAT). Показаний на рисунку 32A перетворювач мережевих адрес (NAT) 550 містить блок перетворення IP-адреси і номера порту 554 і два стеки зв'язку, в тому числі підключений до Інтернету стек зв'язку 553A та стек зв'язку підмережі класу C 553B. Підключений до Інтернету

25 стек зв'язку 553A встановлює зв'язок з усіма іншими мережевими пристроями, підключеними до Інтернету, такими як сервер 22A, маршрутизатор 27 та веб-сервер 511, через загальнодоступну мережу 531. На транспортному рівні 4 стек зв'язку 553A управляє паралельним зв'язком з декількома пристроями, наприклад, 557A і 557B. У наведеному прикладі локальна мережа 552 підключає різні домашні пристрої, наприклад, ноутбук 35, холодильник 34, стаціонарний комп'ютер 35 і маршрутизатор домашньої мережі Wi-Fi 62A, до стека зв'язку підмережі класу C 553B. У цій приватній мережі транспортні протоколи рівня 4 керують зв'язком між стеком зв'язку 553B і мережевими пристроями, наприклад, сполуками 556A і 556B рівня 4. За підтримки обміну інформацією між приватною та загальнодоступною мережами блок перетворення IP-адреси та порту 554 динамічно будує спеціальну таблицю пересилання 555 для перенаправлення

30 передачі кожного пакета приватної мережі в загальнодоступну мережу і навпаки.

Робота NAT ілюструється рисунком 32B, на якому стаціонарний комп'ютер 36 і ноутбук 35, підключені до приватної мережі "за NAT", намагаються одночасно обмінюватися даними з підключеними до Інтернету веб-сервером 21A та сервером електронної пошти 27 тільки через один загальнодоступний IP-адрес для підключення до Інтернету. У наведеному прикладі

40 ноутбука 35 виділена IP-адреса, позначена "NB" та призначений динамічний порт, стаціонарного комп'ютера 36 виділена IP-адреса, позначена "DT" та призначений динамічний порт, веб-сервера 21A виділена IP-адреса, позначена "S1" і надано порт 80 для сервісу веб-сторінок на основі HTTP, а сервера електронної пошти 27 виділена IP-адреса, позначена "S2", і надано порт 110 для послуг електронної пошти на основі IMAP. NAT 550 має загальнодоступна IP-адреса "N" в Інтернеті та використовує призначений динамічний порт.

45

Під час роботи ноутбук 35 ініціює запит веб-сторінки 560A IP-пакетом з IP-адреси відправника NB і довільного номера порту 9999 веб-сервера 21A на IP-адресу одержувача S1 і номер порту 80. Одночасно стаціонарний комп'ютер 36 ініціює запит електронної пошти 561A IP-пакети з IP-адреси відправника DT і довільного номера порту 10200 поштового сервера 27 на IP-адресу одержувача S2 і номер порту 110. Після прийому цих запитів NAT 550 перенаправляє

50 вхідні повідомлення на вихідне Інтернет-з'єднання, здійснюючи трансляцію адреси відповідно до таблиці трансляції 555. Потім NAT відправляє запит від ноутбука 35, зберігаючи IP-адресу одержувача S1 і номер порту 9999, але замінюючи інформацію про відправника від ноутбука 35 на NAT 550 транслявати IP-адреса відправника "N" і номером порту відправника 20000 для створення IP-пакети Інтернету 560B.

55

Аналогічним чином NAT 550 транслює запит від стаціонарного комп'ютера 36 сервера електронної пошти 27, зберігаючи IP-адресу одержувача S2 і номер порту 9999, але замінюючи інформацію про відправника від стаціонарного комп'ютера 36 на NAT 550 з транслюваною IP-адресою відправника "N" і номером порту відправника 20400 для створення IP-пакета Інтернету

60 561B. Таким чином, веб-сервер 21A та сервер електронної пошти 27 вважають, що вони

обмінюються даними з NAT 550 та поняття не мають про запит, що надходить від ноутбука 35 та стаціонарного комп'ютера 36. Взагалі-то, такі IP-адреси, як "NB" або "DT", що використовуються пристроями, підключеними до підмережі NAT, не є допустимими адресами в Інтернеті і не можуть бути використані для підключення безпосередньо без втручання NAT 550.

Після прийому веб-сервером 21A запитуваного 560B IP-пакета, він відправляє у відповідь HTML-код для побудови веб-сторінки по маршруту 560C IP-пакета з IP-адреси відправника "S1" та порту "80" на IP-адресу одержувача "N" і номер порту 20000. Звертаючись до таблиці трансляції 555, NAT знає, що відповіді на номер порту 20000 відповідають запиту з ноутбука 35 та перенаправляє повідомлення, замінюючи IP-адресу одержувача і номер порту на дані ноутбука, а саме IP-адресу "NB" і номер порту 9999, для створення відповідного 560D IP-пакета.

Паралельно з цією транзакцією після отримання запиту у вигляді 560B IP-пакета від NAT 550 сервер електронної пошти 27 відправляє у відповідь IMAP-код, що містить адресу електронної пошти, за маршрутом 561C IP-пакета з IP-адреси відправника "S2" і номера порту 110 на IP-адресу одержувача "N" і номер порту 20400. Звертаючись до таблиці трансляції 555, NAT знає, що відповіді на номер порту 20400 відповідають запиту від стаціонарного комп'ютера 36 та перенаправляє повідомлення, замінюючи IP-адресу одержувача і номер порту на дані стаціонарного комп'ютера, а саме IP-адресу "DT" і номер порту 10200 для створення відповідного 561D IP-пакета. Таким чином, кілька користувачів можуть окремо звертатися до кількох мережевих пристроїв, підключених до Інтернету та сайтам через одну IP-адресу.

Інші протоколи транспортного рівня 4 - Щодо інших відомих протоколів транспортного рівня, крім TCP і UDP, немає єдиної думки про те, чи працюють вони як унікальні і незалежні протоколи рівня 4, або вони працюють як поліпшена версія TCP і UDP рівня 4, або це просто прикладні програми верхнього рівня, що працюють поверх UDP і TCP.

Один з таких протоколів - протокол DCCP (Datagram Congestion Control Protocol - протокол управління перевантаженням датаграми) - є орієнтованим на повідомлення протоколом транспортного рівня для управління контролем перевантаження, який корисний для додатків з обмеженнями часу доставки даних, таких як онлайн-мультимедіа, що розраховані на багатьох користувачів онлайн-ігор, але він не має заходів упорядкування пакетів, прийнятих в неправильному порядку, які є в TCP. Незважаючи на можливість незалежного використання, протокол DCCP застосовується за іншим призначенням - для виконання функцій управління перевантаженням для додатків на основі UDP. Крім трафіку переданих даних, DCCP містить трафік підтвердження, інформує відправника, коли пакет прибув, та чи були відзначені "явні повідомлення про перевантаження" (ECN).

Ще однією спробою керувати своєчасною доставкою пакетів, зокрема тексту, є технологія LCM (Lightweight Communication and Marshaling - легкий зв'язок і маршалінг) на основі опції під LGPL UDP. У порівнянні з одноадресною розсилкою UDP, багатоадресна розсилка UDP має одну перевагу, яка полягає в тому, що кілька додатків можуть стабільно працювати на одному хості або розподілятися по декількох платформах. Крім прагнення мінімізувати затримку мережі, деякі протоколи рівня 4 використовуються для "тунелювання" даних при створенні віртуальних приватних мереж (VPN), які працюють в Інтернеті та через Інтернет. Такими протоколами на основі UDP є протокол GRE (Generic Routing Encapsulation - загальна інкапсуляція маршрутів), протокол PPTP (Point-to-Point Tunneling Protocol - протокол тунелювання точка-точка), протокол SSTM (Secure Socket Tunneling Mechanism - захищений механізм тунелювання сокетів), протокол SSH (Secure Shell - захищена оболонка) та інші. Деякі реалізації VPN призначені для підвищення безпеки, однак в дійсності вони збільшують мережеву затримку.

Крім вищезгаданих стандартизованих протоколів UDP і TCP транспортного рівня 4, неясно, яка ступінь сприйняття пропріетарних протоколів і які компроміси вони пропонують для зниження мережевої затримки за рахунок пошкодження IP-пакетів або підвищення безпеки за рахунок збільшення затримки.

Рівні 5, 6 і 7 моделі OSI - прикладні рівні

У той час як порт # визначає тип запитуваної служби, додаток повинен розуміти природу даних, інкапсульованих як корисне навантаження рівня 4. Вживання заходів на основі вмісту доставленого пакета є завданням верхніх прикладних рівнів моделі OSI - рівнів 5, 6 і 7. Взаємозв'язок декількох пристроїв на прикладному рівні графічно ілюструється на блок-схемі, наведеній на рисунку 33, де три пристрої 570A, 570B і 570C, кожне з яких має свої засоби виконання обчислень та зберігання даних 573A, 573B і 573C, з'єднані відповідними стеками зв'язку 572A, 572B і 572C, спільно використовують зв'язок на прикладному рівні 571. Насправді ці пристрої мають зв'язок на всіх рівнях моделі OSI, але для простоти показується тільки з'єднання прикладного рівня.

Крім підключення до мережі з комутацією пакетів, основне правило для встановлення зв'язку пристроїв на прикладному рівні полягає в тому, що на всіх комунікаційних пристроях повинна існувати одна й та сама сумісна програма. Наприклад, банківська програма не може зрозуміти програму відеоігор, програма САПР не може інтерпретувати онлайн-відео високої чіткості, музичний програвач не може займатися торгівлею на фондовому ринку і т.д. У той час як багато прикладних програм є пропрієтарними і належать одній компанії або провайдеру, деякі додатки та сервіси є широко поширеними, а в деяких випадках навіть мають урядові повноваження для роботи в середовищі з відкритим вихідним кодом. Наприклад, коли компанія Microsoft спробувала прив'язати свій поштовий сервер Outlook тільки та виключно до операційної системи Microsoft Windows, в Європейському Союзі на підставі судових рішень ухвалили, що такі дії порушують антимонопольні закони та змусили Microsoft опублікувати свій поштовий додаток як окрему програму з чітко визначеними підключеннями до операційного середовища, в якій воно працює. Незабаром після цього на кількох обчислювальних платформах з'явилося безліч конкуруючих поштових програм, які використовують поштові протоколи і функції Microsoft.

Різниця між прикладними рівнями 5, 6 і 7 – малопомітна. Як наслідок, багато хто називає ці рівні в семирівневій моделі OSI одним загальним терміном - "прикладні рівні", "верхні рівні" або навіть просто "рівень 7". В останній інтерпретації рівень 7 розглядають як справжній додаток, а рівні 5 і 6 вважають допоміжними рівнями, використовуваними для його обслуговування, аналогічно виклику підпрограм в комп'ютерній програмі. Щоб ще більше заплутати ситуацію, альтернативний п'ятирівневий опис мереж з комутацією пакетів, конкуруючий з семирівневою моделлю OSI, об'єднує всі три прикладні рівні в один рівень, що називається рівнем 5, хоча за структурою він ближче до рівня 7 в моделі OSI.

Сеансовий рівень 5 - У семирівневій моделі OSI рівень 5 називається "сеансовим рівнем" і координує діалоги між додатками та всередині них, включаючи управління дуплексним, напівдуплексним та симплексним зв'язком, а також забезпечення перевірки контрольних точок, відновлення та правильного завершення сеансів TCP. Він також встановлює, управляє та завершує підключення для віддалених додатків безпосередньо в середовищі додатків, які використовують "віддалені виклики процедур" або RPC (від англ. Remote Procedure Call). Рівень 5 також пов'язаний з управлінням сеансами крос-додатків, коли один додаток запитує доступ до процесу іншої програми, наприклад, імпортує діаграму з Excel в PowerPoint. Ще один додаток рівня 5 - протокол SOCKS (SOCKEt Sequire - "захищений сокет") - це інтернет-протокол, який використовується для маршрутизації IP-пакетів між сервером і клієнтом через проксі-сервер і для виконання "перевірки автентичності", щоб обмежити доступ до сервера тільки авторизованим користувачам. Спираючись на ідентифікацію користувача для надання або заборони доступу та привілеїв, SOCKS забезпечує настільки ж надійний захист, як і використовувані процеси автентифікації.

В процесі роботи SOCKS діє як проксі-сервер, здійснюючи маршрутизацію TCP-з'єднання через довільну IP-адресу і надаючи послуги пересилання для UDP-пакетів. У випадках, коли клієнту блокується доступ до сервера через міжмережевий екран, за допомогою SOCKS клієнт може зв'язатися з проксі-сервером SOCKS в мережі клієнта та запросити з'єднання, яке клієнт хоче встановити, щоб зв'язатися з сервером. Після отримання доступу до сервера проксі-сервер SOCKS відкриває з'єднання через міжмережевий екран та здійснює обмін даними між сервером і клієнтом так, як ніби міжмережевий екран відсутній. Діючи на більш низькому рівні, ніж проксі-сервери на основі HTTP, SOCKS використовує метод квітування для інформування програмного забезпечення проксі-сервера про з'єднання, яке клієнт намагається встановити без інтерпретації або перезапису заголовків пакетів. Після отримання інформації щодо під'єднання SOCKS працює непомітно для користувачів мережі. У більш новій версії SOCKS, званої SOCKS4, розширені можливості програмного забезпечення, при цьому клієнти можуть вказувати доменне ім'я одержувача, а не необхідну IP-адресу.

Оскільки SOCKS не більш надійний, чим процес автентифікації, який використовується для ідентифікації авторизованого користувача, він може бути перетворений хакерами та злочинцями в засіб для подолання захисту мережевого доступу. Для боротьби з цим впливом була розроблена версія SOCKS5, пропонує більшу кількість варіантів автентифікації, а також додатково надає підтримку перенаправлення UDP з використанням запитів DNS (сервера доменних імен). SOCKS5 також був оновлений для підтримки IP-адрес IPv4 і IPv6. Під час підтвердження встановлення зв'язку і узгодження сеансу клієнт та сервер ідентифікуються за номером методу, застосовуваного для автентифікації, а саме:

- 0 × 00: Автентифікація не виконується
- 0 × 01: Методи GSSAPI

- 0 × 02: Ім'я користувача/пароль
- 0 × 03-0 × 7F: Методи, призначені IANA (Управлінням по присвоєнню інтернет-номерів)
- 0 × 80-0xFE: Методи, зарезервовані для приватного використання

Після завершення переговорів та вибору методу автентифікації можна починати інформаційний обмін. Доведено, що найпростіша процедура автентифікації "ім'я користувача/пароль" за своєю суттю небезпечна та легко зламуються, особливо при чотирьохсимвольному паролі. В якості альтернативи застосовується "універсальний прикладний програмний інтерфейс служби безпеки" (GSSAPI), який сам по собі є не методом захисту, а стандартизованим інтерфейсом IETF, що викликає бібліотеку програмного забезпечення, що містить код безпеки і методи перевірки автентичності, в основному написані провайдерами послуг безпеки. Використовуючи GSSAPI, користувачі можуть змінювати свої методи захисту, не переписуючи текст прикладної програми. Виклики процедури включають в себе отримання доказу ідентичності користувача або секретного криптографічного ключа, генерування токена клієнта або відправки виклику на сервер і отримання відповідного токена, перетворення даних програми в захищений або зашифрований токен і його відновлення і т.д. В якості альтернативи, Управління з присвоєння інтернет-номерів (IANA), що є підрозділом некомерційної інтернет-корпорації з присвоєння імен і номерів (ICANN), відповідно до свого статуту, призначає певні методи для забезпечення стабільності і безпеки мережі.

Представницький рівень 6 - Рівень 6 управляє синтаксичним поданням даних і об'єктів, включаючи підтримку угоди з кодування символів, аудіо, відео і графічним форматам. По суті, представницький рівень, іноді званий синтаксичним рівнем, готує або перетворює файли та вбудовані об'єкти в форму, яку буде використовувати додаток, і "представляє" дані на прикладний рівень 7. Наприклад, якщо графічний об'єкт прийнятий в форматі, незрозумілому даному додатку, програмне забезпечення представницького рівня, по можливості перетворює або трансформує цей формат в прийнятний для цього додатка. І навпаки, рівень 6 може перетворювати спеціально відформатовані об'єкти в стандартні формати та вкладати їх перед передачею на сеансовий рівень 5. Таким чином, рівень 6 встановлює синтаксичний контекст між різними додатками для переміщення даних вгору та вниз по стеку зв'язку та протоколу. Наприклад, графічний об'єкт, створений в Adobe Illustrator або AutoCAD, може бути імпортований та вбудований в презентацію PowerPoint або в документ електронної пошти на основі HTTP.

Рівень 6 також відповідає за шифрування, тобто форматування і шифрування даних перед відправкою по мережі, і, навпаки, за дешифрування даних і їх переформатування перед поданням на прикладний рівень. Наприклад, після отримання файлу з роздільниками позицій табуляції, відправленого в зашифрованому форматі через Інтернет, рівень 6, після розшифровки файлу відповідно до встановлених ключами дешифрування, може переформатувати дані для імпорту в електронну таблицю на основі рядків та стовпців, наприклад, Excel, або такої реляційної бази даних, як Oracle. Для підвищення безпеки шифрування і дешифрування на рівні 6 може бути обмежена авторизованими відправниками та одержувачами, чия ідентичність підтверджена апіорі за допомогою процедури автентифікації рівня 5. Безпека такого зв'язку не вище, ніж при шифруванні, використовуваному для забезпечення конфіденційності файла даних, та при автентифікації, що використовується для підтвердження права користувача на доступ до файлу даних.

Незважаючи на те, що програмне забезпечення представницького рівня може бути розроблено по замовленню для конкретного пристрою або операційної системи, для забезпечення транспортабельності та функціональної сумісності програма може бути побудована з використанням базових правил кодування ASN.1 (Abstract Syntax Notation, version 1 - абстрактна синтаксична нотація, версія 1), в тому числі таких можливостей, як перетворення текстового файлу з кодуванням EBCDIC в файл з кодуванням ASCII або серіалізація об'єктів та інших структур даних з XML та в XML. Як протокол представницького рівня 5 ASN.1 зіставляє структуровані дані з конкретними правилами кодування, наприклад, перетворюючи ціле число в бітовий рядок, що підлягає передачі, і аналогічно декодує бітовий рядок, використовуючи "правила кодування XML", також відомі як XER. Приклади різних форматів, які охоплюються операціями рівня 6:

- текст, в тому числі в форматах ASCII і EBCDIC;
- графічна інформація, в тому числі в форматах PNG, JPG, GIF, BMP, EPS;
- звук та відео, в тому числі в форматах MP4, WMV, MOV, AVI, MIDI;
- документи, в тому числі в форматах PDF, DOC, PPT, HTML, XML, MIME, архіви (наприклад, ZIP);
- потокові дані, в тому числі в форматах RTP, RTSP, RTMP;

- зашифровані дані, в тому числі в форматах TLS/SSL, SSH.

Прикладний рівень 7 - У семирівневій моделі OSI "прикладний рівень 7 спрощує інтерфейс між користувачем, клієнтом або пристроєм з хостом, сервером або системою. Оскільки прикладний рівень найближче до користувача, він спрощує інтерфейс між користувачем і хостом. У разі, коли користувач - це людина, а хост - електронний пристрій, наприклад, стільниковий телефон або комп'ютер, цей інтерфейс спрощується за рахунок натискання клавіш, натискання або жестів з використанням клавіатури або сенсорного екрану або іноді за допомогою голосу. Інтерфейси сенсорного екрану, спочатку звані GUI (Graphical User Interface - графічний інтерфейс користувача), в значній мірі поступилися місцем пристроїв, званим терміном UI/UX (User-Interface/User-Experience - інтерфейс користувача/взаємодія з користувачем) - конструкції інтерфейсу, заснованої на вивченні взаємодії людини і машини. У M2M-підключення і машинно-інфраструктурних (M2X) системах інтерфейс людини замінюється різнорідними апаратними пристроями, що говорять на різних машинних мовах.

Незалежно від цих відмінностей, прикладний рівень повинен дозволяти людині і машині або декільком машинам обмінюватись між собою інформацією у зрозумілій їм формі. Оскільки модель OSI працює зі стеком зв'язку та протоколу, ці інтерфейси виходять за рамки моделі OSI, але все ж грають важливу роль в обговоренні предмета переговорів, включаючи ідентифікацію партнерів по зв'язку, визначення доступності ресурсів та синхронізацію зв'язку. При ідентифікації партнерів по зв'язку рівень 7 повинен визначити, чи має інша сторона встановлене програмне забезпечення, чи дозволено встановлювати зв'язок, та вводить правильні облікові дані.

У деяких випадках може знадобитися, щоб рівень 5 спочатку підтвердив ідентичність іншого боку, перш ніж ініціювати обмін даними. Це підтвердження може бути виконано під час запиту на обмін інформацією або встановлено апіорі за допомогою процесу прив'язки або з використанням перевірки AAA - трьохетапної процедури, що означає автентифікацію, авторизацію і адміністрування. У комунікаційних додатках, таких як стільниковий телефонний зв'язок з використанням технології VoIP, прикладне програмне забезпечення також має перевірити наявність підтвердження в мережі і достатню стабільність для розміщення виклику, тобто встановити, що послідовність IP-пакетів відправлена і отримана з досить малою затримкою, щоб говорити про прийнятний рівень якості обслуговування. При синхронізації зв'язку весь інформаційний обмін між додатками вимагає спільних дій, керованих прикладним рівнем.

Прикладами реалізації завдань на прикладному рівні є емуляція терміналів, поштові служби, управління мережею, веб-браузери, управління файлами, резервне копіювання і хмарні сервіси зберігання, драйвери периферійних пристроїв, в тому числі:

- управління файлами, в тому числі FTP, FTAM, SFTP, NNTP, IRC, SIP, ZIP;
- веб-браузери, в тому числі HTTP (наприклад, Safari, Firefox, Chrome, Outlook, Netscape і т.д.);
- поштові служби, в тому числі SMTP, IMAP, POP3, а також Microsoft Outlook, Apple Mail, Google Gmail, Yahoo, Hotmail і т.д.;
- служби зв'язку та мовлення, в тому числі SIP, NNTP, IRC і OTT ("over-the-top") призначені для користувача реалізації;
- управління мережею, в тому числі DNS, SNMP, DHCP, SNMP, BGP, LDAP, CMIP;
- емуляція терміналів, в тому числі Telnet;
- служби резервного копіювання та хмарного зберігання, в тому числі NFS і комерційні версії Android, iOS, Apple Time Machine, Apple iCloud, Carbonite, Barracuda, Dropbox, Google Drive, Microsoft One Drive, Box;
- драйвери периферійних пристроїв, в тому числі принтерів, сканерів, камер, флеш-карт;
- додатки безпеки, наприклад, Symantec, Norton, AVG.

Підкреслені приклади найбільш поширених додатків для комп'ютерів та смартфонів включають передачу файлів, передачу гіпертексту для перегляду веб-сторінок, служби електронної пошти та пошукові запити DNS для перетворення доменних імен в IP-адреси. Завдяки їх широкому поширенню у цих загальновідомих додатків є виділені порти, призначені для таких служб.

Додатки управління файлами - Один загальновідомий додаток рівня 7 - це програма передачі файлів (FTP), яка використовується для відправки файлів або завантаження даних. Файли після завантаження "записуються" в енергонезалежний накопичувач для подальшого використання. Якщо в цих файлах міститься виконуваний код, програма завантаження і установки разом з операційною системою пристрою відкриває та встановлює програмне забезпечення в каталог додатків на комп'ютері або мобільному пристрої.

Цей процес ілюструє рисунок 34, де ноутбук 35, якому виділено числову IP-адресу "NB" та призначений динамічний порт, запрошує файл з файлового сервера 21A шляхом відправки 580 IP-пакета як запит FTP з використанням транспортного протоколу TCP на номер порту 21 - керуючого порту FTP файлового сервера. Результуючий 580 IP-пакет включає в себе IP-адресу одержувача "S1", номер порту одержувача 21, а також свою IP-адресу відправника "NB" і номер свого спеціального порту 9999. Оскільки порт №21 - це порт управління для запиту послуг передачі файлів, файловий сервер 21A знає, що ноутбук 35 запитує файл та чекає реєстраційну інформацію для підтвердження IP-адреси і номера порту одержувача пакета.

Потім під час активного сеансу FTP ноутбук 35 відправляє адресу одержувача і номер порту одержувача для запитаного файлу, аналогічно платіжними реквізитами для банківського переказу, що містять SWIFT-код і номер рахунку. Результуючий 581 IP-пакет включає в себе IP-адресу ноутбука "NB" і номер його порту 9999 як інформацію про відправника, і IP-адресу сервера "S1" як одержувача. Номер порту одержувача пакета змінюється на порт №20 для встановлення каналу даних FTP окремо від підключення команд.

Потім в якості відповіді файловий сервер 21A відкриває корисне навантаження IP-пакета для визначення імені файлу і, необов'язково, шляхи до запитуваного файлу, та після визначення місцезнаходження файлу 583 вкладає його в відповідь 582 IP-пакет та відправляє пакет назад за вказаними даними ноутбука 35 шляхом заміни місцями IP-адрес та портів, тобто одержувачем стає IP-адресою "NB" з номером порту 9999, а відправником стає IP-адреса "S1" та порт №20. Як і в попередніх двох транзакціях, IP-пакет використовує TCP як транспортний механізм.

Як тільки ноутбук 35 отримує файл, його витягують з корисного навантаження пакета 582 і, можливо, перетворюють з використанням представницького рівня 6 в файл даних 583 для зберігання або для завантаження в операційну систему 585 ноутбука. В останньому випадку ця програма або інша програма - утиліта в операційній системі - завантажує виконуваний код файлу 583 для створення прикладної програми 586.

Для первісної реалізації активної передачі файлів FTP зберігаються дві проблеми. По-перше, оскільки порт команд FTP (порт №21) є відкритим стандартом, хакери часто використовують його, щоб спробувати підставити свої ідентифікаційні дані та завантажити файли, доступ до яких для них є несанкціонованим, або іншим чином здійснити атаку типу "відмова в обслуговуванні", яка позбавляє пристрій можливості працювати. Іншою проблемою активної передачі FTP є те, що 582 IP-пакет, що відправляється з файлового сервера, може блокуватися NAT або мережевим екраном, які перехоплюють його доставку ноутбука 35. Варіант цієї процедури, званий пасивним FTP, може обійти проблему брандмауера, але в даний час більшість маршрутизаторів з NAT сумісні з FTP та підтримують передачу файлів з належними обліковими даними або автентифікацією.

На додаток до FTP-сервісів, доступним для порту №20, або замість них застосовується "захищений протокол передачі файлів", також відомий як протокол передачі файлів SSH. Ця передача використовує захищену оболонку або порт SSH (порт №22) - це той же порт, який використовується для захищених логінів та захищеного перенаправлення портів. До альтернативних додатків для передачі файлів ставиться менш популярний протокол FTAM (File Transfer, Access and Management - передача, доступ і управління файлами), а також процедури стиснення даних з використанням ZIP і інших алгоритмів.

Веб-браузери та веб-сервери - Ще один широкий клас додатків рівня 7 включає програми, які використовують спеціальну технологію форматування під назвою "гіпертекст". Ці програми включають "веб-сервери", які зберігають гіпертекстові документи; "веб-браузери", які читають та відображають їх; а також спеціальний протокол передачі повідомлень з призначеними виділеними зареєстрованими портами для більш швидкісного доступу. Ключовим компонентом є веб-браузер - графічно орієнтована комунікаційна програма, призначена для завантаження та відображення гіпертекстових документів з Інтернету, інтрамережі або інших мереж з комутацією пакетів. Мережевим компаньйоном браузера є веб-сервер - швидкодіючий комп'ютер, який розподіляє гіпертекстові документи згідно запиту браузера на їх доступ. Гіпертекст також може використовуватися для відображення електронних листів з вбудованим форматуванням, недоступним для простих пристроїв перегляду електронної пошти.

В процесі роботи браузер не встановлюють пряме з'єднання один з одним, а обмінюються інформацією через посередників, що містять один або кілька веб-серверів, доступних для обох. Щоб опублікувати документ, користувач просто "відправляє" документ або зображення на "веб-сторінку", розміщену на будь-якому сервері, підключеному до Інтернету або до будь-якої іншої приватної або загальнодоступної мережі або хмари. Користувач, який ініціює передачу документа, вирішує, хто має доступ до опублікованих файлів, та чи мають вони права на

читання або редагування. Веб-сервер, на якому розміщуються документи, може належати видавцеві документа, або знаходитися під його управлінням, або може представляти незацікавлену сторону, не пов'язану з розміщеним контентом і дизайном веб-сторінки.

У документах на основі гіпертексту використовується спеціальна мова форматування документа, яка називається HTML (HyperText Markup Language - мова розмітки гіпертексту) для відображення текстового, графічного та відеоконтенту способом, який динамічно налаштовується так, щоб найкращим чином розміститись у вікні, в якому він буде відображатися. Функція HTML полягає в завантаженні матеріалу для відображення і його динамічному форматуванні на кожній сторінці. Кожна сторінка може містити як статичні, так і динамічно змінні розмірні поля з текстом, завантаженим з жорстко запрограмованого програмного забезпечення або завантажуваних з файлу або бази даних. Незважаючи на складність розробки та запису, перевага використання бази даних для вмісту HTML-сторінки полягає в тому, що база даних може оновлюватися часто або регулярно, при цьому веб-сторінка буде налаштована автоматично. В іншому випадку кожен веб-сторінку потрібно буде переробляти при зміні контенту. HTML також визначає місце розташування об'єктів, в тому числі нижніх та верхніх колонтитулів, бічних панелей та полів з фіксованим місцем розташування, а також плаваючих об'єктів, які динамічно обтікаються текстом.

Самі об'єкти можуть являти собою статичні графічні об'єкти або фотографії, анімовану графіку, флеш-відео, аудіофайли, відеоролики і фільми з високою чіткістю зображення і багато іншого. Як і текст, форматування може бути жорстко закодовано або динамічно пов'язане. Пов'язані об'єкти можуть бути динамічно перетворені з одного формату або типу об'єкта в інші за допомогою функцій представницького рівня 5. Наприклад, задане поле в електронній таблиці може бути перетворено в статичний знімок або графічний об'єкт під час малювання сторінки. Інші об'єкти також можуть містити прямі посилання на інші сервери та веб-сайти, а при натисканні можна передавати інформацію про комп'ютер користувача, що переглядає веб-сторінку, особисту і контактну інформацію, переваги та інтереси з попередніми схваленням або без попереднього схвалення користувача, що переглядає веб сторінку. По суті, клацання користувача на посиланні вважається мовчазною згодою з умовами хоста зв'язаної веб-сторінки. Наприклад, клацання на рекламному оголошенні нового автомобіля може відправити інформацію в базу даних людей, зацікавлених у купівлі нових автомобілів, та привести до отримання небажаного "спаму" по електронній пошті про нові рекламні акції, що відправляються на особисту адресу електронної пошти користувача. З цього часу на динамічних веб-сторінках може автоматично запускатися вміст полів банерної реклами, що відображає автомобільну рекламу - все це наслідки однієї єдиної дії - клацання користувача на посиланні та перегляду реклами. Компанії інтернет-маркетингу продають таку інформацію про користувачів торговцям і рекламодавцям, навіть не знаючи, чи відображають їх дані про поведінку користувача його дійсні наміри або вони є випадковими.

Важливо відзначити, що в документах на основі гіпертексту велика частина тексту і майже всі об'єкти, що використовуються для створення запитуваної веб-сторінки, не включаються до початкового HTML завантаження веб-сторінки, а завантажуються після завантаження початкової HTML-сторінки. Документи і об'єкти не завантажуються з використанням протоколу FTP, але замість цього використовують більш динамічний процес, званий HTTP (HyperText Transfer Protocol - протокол передачі гіпертексту). HTTP являє собою додаток і формат даних, що працюють на Рівні представлення 6 і на Прикладному рівні 7, обробляючи запити додатків, наприклад, веб-браузерів.

На рівні 4 - транспортному рівні - для доступу до мережі HTTP користується своїм власним зарезервованим номером порту, а саме - портом №80. Оскільки порт №80 часто авторизований і розблокований міжмережевими екранами або програмним забезпеченням безпеки, наприклад, портом 21 FTP, порт 80 є улюбленою мішенню для хакерів, які бажають отримати несанкціонований доступ або документи, або щоб ініціювати атаки типу "відмова в обслуговуванні", шкідливої атаки на сервер, щоб усунути його від підтримки звичайних функцій, змушуючи його обслуговувати беззмістовні FTP- або HTTP-запити від хакера або зловмисника.

Процедуру завантаження веб-сторінки по протоколу HTTP ілюструє рисунок 35А, де ноутбук 35, якому виділено IP-адресу "NB" та призначений спеціальний порт №9999, запитує HTML-документ з веб-сервера 21А за IP-адресою "S1" за допомогою 590 IP-пакета. Для запиту веб-сторінки 590 IP-пакет вказує порт №80 веб-сервера. Потім в якості відповіді веб-сервер 21А прикріплює корисне навантаження HTML та повертає 591 IP-пакет, замінюючи адреси і номери портів на дані пакета 591, а саме, відправником тепер є порт №80 на IP-адресу 9999, а одержувачем тепер є порт № 9999 на IP-адресу "NB". Дані HTML передаються з використанням

з'єднання на основі TCP для забезпечення високої надійності доставки корисного навантаження.

Після отримання коду HTML браузер в ноутбучі зчитує HTML-файл та по черзі ідентифікує IP-виклики для завантаження контенту на веб-сторінку. У показаному прикладі перший виклик для графічної інформації полягає в завантаженні контенту з того ж веб-сервера 21А, що і при першому завантаженні, тому ноутбук 35 знову готує 592 IP-пакет для IP-адреси одержувача "S1" та порту №80. Оскільки порт ноутбука призначається динамічно, для відправника 592 IP-пакета він змінюється на спеціальний порт №10001, але зберігається IP-адреса "NB". Як відповідь веб-сервер 21А вкладає JPEG-файли в корисне навантаження 539 IP-пакета, замінюючи адреси відправника і одержувача так, щоб відправником був порт №80 з IP-адреси "S1", а одержувачем - порт 10001 на IP-адресу "NB". Отримавши 593 IP-пакет, браузер в ноутбучі розгортає корисне навантаження, перетворює графічний формат з використанням представницького рівня 6 в сумісний з браузером формат, потім визначає і розміщає розміри зображення на сторінку браузера, тобто на прикладний рівень 7.

Відповідно до рисунку, наступний запит на завантаження об'єкта на сторінці HTML надходить не з веб-сервера S1, а з зовсім іншого сервера, зокрема з медіасервера 511, що має IP-адресу "S5". Таким чином, веб-браузер у ноутбучі 35 готує IP-пакет 594 як інший HTTP-запит до порту одержувача №80, на цей раз за IP-адресою одержувача "S5". Незважаючи на те, що IP-адреса відправника "S1" залишається такою ж самою, при призначенні динамічного порту номер порту відправника знову змінюється, на цей раз, на порт №10020. У відповідь медіасервер 511 готує IP-пакет 595 від відправника з IP-адресою "S5" і номером порту 80 на останню IP-адресу ноутбука "NB" та порт №10030. Прикріплене корисне навантаження, вкладене в IP-пакет 595, містить дані у форматі MPEG. Після їх прийому рівень представлення 6 готує файли, доставляє їх на прикладний рівень 7, де браузер встановлює їх, продовжує зчитувати HTML-код та збирати веб-сторінку до тих пір, поки вона не буде заповнена.

Тому, при використанні HTML, контент веб-сторінки не створюється з одного завантаження, як файл, відправлений з використанням FTP, а створюється з використанням послідовності викликів на різні сервери, кожен з яких приносить конкретний контент. Ця концепція ілюструється графічно на рисунку 35В, де HTML-сторінка 591, текст і дані в форматі JPEG 593 завантажуються з порту №80 веб-сервера "S1", відео у форматі MPEG 595 завантажується з порту №80 медіасервера 511, а фотографії в форматі PNG 596 і дані в форматі JPEG 597 надходять з порту 80 файлового сервера 27. Таким чином, веб-сторінка будується за даними з декількох джерел. Крім HTML, код запитує різні текстові, графічні та аудіо-відео елементи; в такому випадку, немає центрального командного пункту або системи управління, відповідальних за створення документа. Якщо, наприклад, один сервер відповідає повільно через власну завантаженість або через перевантаженість трафіку, картинка веб-сторінки 591 може зависати, зупиняючись на якийсь час до її заповнення. Це переривання може не мати нічого спільного з хостом веб-сторінки, наприклад, Yahoo, але може бути викликано пов'язаними серверами, що викликаються веб-сторінками HTML, наприклад, серверами новин CNN або Fox.

Одне з джерел небезпеки, пов'язаної з веб-сторінками HTML - це можливість для хакерів і шкідливого ПЗ збирати інформацію про користувача, зокрема, якщо з'єднання перенаправляється на фішинговий піратський сайт для збору персональної інформації від імені дійсно етичного бізнесу, для ведення якого потрібна дійсна домашня адреса користувача, номер кредитної картки, PIN-код, номер страхового свідоцтва тощо.

Всесвітнє павутиння - Один надзвичайно популярний, якщо не універсальний, додаток HTML являє собою перегляд веб-сторінок з документами, доступними у Всесвітньому павутинні, зокрема, по веб-адресам, набір яких у браузері починається з букв "www". Кожен раз, коли користувач вводить веб-адресу, також відому як URL (Uniform Resource Locator - уніфікований локатор ресурсу), в адресний рядок браузера, наприклад, "http://www.yahoo.com", браузер відправляє запит до маршрутизатора, розташованого безпосередньо над ним, для визначення цільової IP-адреси. У цьому процесі, проілюстрованому раніше на рисунку 3, бере участь ноутбук 60, що відправляє IP-пакет до маршрутизатора 62А із запитом порту №53. Це номер порту, що ідентифікує запит послуг для пошуку сервера доменних імен (DNS). Маршрутизатор 62А перенаправляє запит DNS на маршрутизатор 62А сервера доменних імен, який, в свою чергу, надає числове значення IP-адреси цільового домену. Якщо, наприклад, сервер 66А є веб-сервером Yahoo з числовим значенням IP-адреси "S11", то сервер доменних імен (DNS) 71 поверне IP-адресу маршрутизатора 62А, а IP-пакет буде побудований з IP-адресою "S11" та портом одержувача веб-сторінки №80.

Слід зазначити, що, хоча у Всесвітньому павутинні доступні багато документів, не всі інтернет-документи розміщуються в ній. Деякі веб-сторінки, наприклад, незважаючи на те, що

вони доступні в мережах загального користування, які не використовують префікс www, перш за все, щоб створити труднощі для хакерів при їх пошуку. Інші веб-сервери використовують приватні мережі або інтрамережі, приховані і доступні тільки за мережевим екраном або через доступ, що використовує зашифрований канал або тунель, званий VPN (Virtual Private Network - Віртуальна приватна мережа). Щоб зрозуміти унікальну властивість Всесвітнього павутиння, важливо зрозуміти його розвиток і еволюцію, які визначають, як його переваги та сильні сторони, так і його недоліки та вразливі місця.

Історично, до винаходу Всесвітнього павутиння і браузера, зв'язок через Інтернет головним чином ґрунтувався на електронній пошті і на передачі файлів з використанням протоколу FTP. Потім в 1989 році Тім Бернерс-Лі продемонстрував перший успішний інтернет-зв'язок між клієнтом та сервером, використовуючи протокол HTTP (HyperText Transfer Protocol - протокол передачі гіпертексту). Після цього в Національному центрі суперкомп'ютерних додатків в Університеті штату Іллінойс в Урбана-Шампейн Марк Андріс (Marc Andreessen) розробив перший повнофункціональний браузер під назвою Mosaic, відомий своїм новаторським інтуїтивним інтерфейсом, підтримкою декількох інтернет-протоколів, сумісністю з середовищами Macintosh і Microsoft Windows, забезпечення сумісності та підтримки більш ранніх протоколів, таких як FTP, NNTP і gopher, а також простотою установки, стабільністю та високою надійністю. Важливим є те, що Mosaic був першим браузером для відображення зображень та тексту на одній сторінці, а не відкривав графіку в окремому вікні.

Mosaic був швидко комерціалізований в Netscape Navigator і багато в чому відповідальний за прискорення інтернет-революції і широке використання веб-сайтів для особистих і бізнес-додатків. Незважаючи на те, що в даний час існує безліч браузерів, Firefox - прямий нащадок Mosaic і Netscape, а також Microsoft Explorer, Apple Safari і Google Chrome сьогодні є найбільш широко використовуваними браузерами. Для полегшення пошуку документів і контенту у Всесвітньому павутинні одночасно з'явився ще один клас додатків - пошукова система веб. Такі пошукові системи, як Google і Yahoo, сьогодні домінують на ринку.

У міру проникнення бізнесу в Інтернет зароджувалася електронна комерція на основі продажів та покупок через Інтернет, здійснюваних на спеціальних сайтах, таких як Amazon, eBay, Barnes & Noble, Best Buy і віднедавна Alibaba. Незабаром сталася фрагментація ринку, при цьому провайдери стали спеціалізуватися на певному типі продуктів або послуг, а не пропонувати послуги спеціального веб-сайту електронної комерції. Наприклад, швидко з'явилися торгові компанії, що спеціалізуються на порівнянні цін при поїздках та перевезеннях, такі як Priceline, Expedia, Orbitz і Sabre, разом з власними пропріетарними електронними базами. Для користувачів, що бажають придбати музику, відео, електронні книги, ігри та програмне забезпечення, пропонують свої онлайн сервіси такі провайдери, як iTunes і AppStore компанії Apple, Walmart, Amazon MP3, Google Play, Sony Unlimited Music, Kindle Fire і Windows Store. Служби онлайн-відео і аудіо, наприклад, iTunes, Google Play, Netflix, Hulu Plus, Amazon Prime, а також провайдери радіомовних і кабельних програм iHeart, наприклад, Comcast Xfinity, стають все більш популярними, особливо завдяки послугам Wi-Fi, пропонованим в літаках, автобусах, лімузинах, а також на вокзалах та в кафе по всьому світу.

Незважаючи на побоювання з приводу конфіденційності та безпеки, діти і молодь в даний час публікують величезну кількість особистої інформації на загальнодоступних веб-сайтах. Індустрія, так звані "соціальні мережі", починалася з веб-сайтів, які забезпечували зручність публікації, оновлення та редагування документів, в яких люди публікували свої особисті думки, ділилися досвідом, вели веб-щоденники або "блоги". Потім YouTube надав творчим особистостям можливість публікувати та поширювати домашні відео. Facebook розвинув цю тенденцію, пропонуючи функції блогу, хронологічно об'єднані з фото- та відеоматеріалами в інтерактивному форматі, де відвідувачі вашої "домашньої сторінки" публікують коментарі та ставлять "лайки", коли їм подобається те, що вони читають або бачать. Facebook також розширив можливості управління контактами, перегляду списків контактів для друзів і їх додавання в Facebook, і дозволив власнику облікового запису "дружити" з іншими користувачами, запросивши доступ до їх домашньої сторінки, або ігнорувати їх. Завдяки розширенню можливостей управління персональними контактами людей, число користувачів Facebook зросло в багато разів, що дозволило людям із застарілою контактною інформацією знайти один одного в соціальних мережах. Потім ті ж методи соціальних мереж були адаптовані для знайомств, пошуку партнерів або отримання сексуальних послуг (легальних або нелегальних), а в світі професіоналів для контактів з колегами по галузі, наприклад, використовуючи LinkedIn.

Оскільки Всесвітнє павутиння засновано на тій же філософії з відкритим вихідним кодом, що і мережі Internet і OSI з комутацією пакетів, в ній немає центрального командного пункту або

системи управління, і, як така, вона залишається нерегульованою, що ускладнює будь-яким урядовим чи регулюючим організаціям здійснювати контроль, вводити обмеження або піддавати цензурі її контент. Більш того, завдяки публікації особистої інформації, злочинцям стало легше "стежити" за людьми, збираючи їх публічну інформацію, щоб було легше підбирати їх паролі, стежити за їх діями та навіть відстежувати їх місцезнаходження, де б вони не знаходилися, за даними GPS і інформації про транзакції. У деяких випадках, наприклад, в контактній і довідковій службі з відкритим вихідним кодом, званої списком Крейга, сексуальні маніяки та вбивці приховують свою особистість і наміри, щоб знайти жертв своїх злочинів. Крім злочинців і хакерів, що використовують Всесвітнє павутиння та соціальні мережі для спостереження за своїми цілями, недавні новинні одкровення показали, що уряди також відстежують і контролюють електронні листи громадян, телефонні розмови, веб-сайти, блоги і навіть щоденне переміщення без видимої причини і дозволу для виконання цих дій. Один з аргументів, які використовуються для виправдання такого вторгнення, полягає в тому, що використання інформації, вільно розповсюджуваної на публічному сайті або в загальнодоступній мережі, є "чесним заняттям" і що необхідно заздалегідь запобігати злочинним та терористичним діям до того, як вони відбудуться, що багато в чому подібно "майбутнього злочину" в популярному фільмі "Особлива думка", та саме по собі є виправданням такого агресивного спостереження і шпигунства.

Як реакція на крадіжку ідентифікаційної інформації та такого небажаного вторгнення з боку урядових організацій, споживачі переходять на такі сайти, як Snapchat, та телефонні служби, що гарантують підвищену безпеку і конфіденційність, що вимагають підтвердження або "автентифікації" з іншого боку того, кого ви знаєте і кому довіряєте. Проте, такі "зони довіри", як їх тепер називають, залежать від методів захисту, доступних для мереж зв'язку з комутацією пакетів. Як видно з даних наведених у вступному розділі даного документа, ці мережі, протоколи зв'язку, веб-сайти та сховище даних все-таки не є абсолютно безпечними, інакше в пресі сьогодні не було б так багато повідомлень про кіберзлочинність.

Додатки електронної пошти - Одним з найбільш поширених і найстаріших програм в мережах з комутацією пакетів є електронна пошта або "email". Цей процес ілюструє рисунок 36, де ноутбук 35, якому виділено числова IP-адреса "NB" та призначений динамічний порт, завантажує IP-пакет електронної пошти 601 на сервер електронної пошти 600. На додаток до вкладеного корисного навантаження електронної пошти SMTP, IP-пакет електронної пошти на основі протоколу TCP 601 включає в себе IP-адресу одержувача "S9", порт одержувача №21 або, як варіант, порт №465 зі своєю IP-адресою відправника "NB" і його пропріетарним портом №10500. Порт №21 представляє поштові служби, що використовують простий протокол передачі пошти (SMTP), а порт №465 є "захищеною" версією SMTPS на основі технології SSL. Однак в недавніх новинах повідомили, що SSL визнаний вразливим і не повністю захищений від хакерів.

У відповідь на отримання IP-пакета електронної пошти 601 сервер електронної пошти 600 підтверджує його прийом, повертаючи IP-пакет 602, що містить підтвердження SMTP, що відправляється на IP-адресу одержувача "NB" та порт 10500 з сервера електронної пошти 600 з IP-адресою відправника "S9" з використанням порту №21 або порту SSL №46. Тим часом сервер електронної пошти 600 одночасно передає електронного листа як повідомлення IMAP в IP-пакеті 605 з IP-адреси відправника "S9" та порту IMAP №220 стаціонарного комп'ютера 36 на IP-адресу одержувача "DT" та спеціальний порт №12000. Після отримання повідомлення електронної пошти стаціонарний комп'ютер 36 підтверджує повідомлення IMAP сервера електронної пошти 600 IP-пакетом 604 з IP-адреси відправника "DT" та порту №12000 на IP-адресу одержувача "S9" та порт 220. Таким чином, в транзакції по доставці електронної пошти задіяні три сторони, в тому числі відправник - ноутбук 35, сервер електронної пошти 600 і одержувач - стаціонарний комп'ютер 36. В ході інформаційного обміну відправник використовує протокол SMTP, а одержувач повідомлення використовує протокол IMAP для підтвердження повідомлення. Обмін по протоколу IMAP оновлює базу даних на сервері і на стаціонарному комп'ютері, щоб забезпечити відповідність записів їх файлів. Оскільки сервер електронної пошти виступає в якості посередника, є можливість перехопити повідомлення або шляхом перехоплення 601 IP-пакета між ноутбуком та сервером, або 605 IP-пакета між сервером та стаціонарним комп'ютером, або шляхом злому файлу, що зберігається на сервері електронної пошти 600. В альтернативному варіанті, для доставки пошти також може використовуватися додаток POP3 (Поштовий Офісний Протокол), але без синхронізації файлового сервера.

Інші додатки рівня 7 - Крім системи управління файлами, веб-браузерів, серверів DNS і функцій електронної пошти, існує безліч інших додатків, в тому числі емуляція терміналів з використанням Telnet, управління мережею, драйвери периферійних пристроїв, утиліти

резервного копіювання, програми забезпечення безпеки, а також додатки для зв'язку і мовлення. Наприклад, додатки резервного копіювання включають в себе "мережеву файлову систему" (NFS) на основі TCP, тепер у своїй четвертій версії, а також комерційне програмне забезпечення резервного копіювання, включаючи користувацькі версії для Android, iOS, Apple Time Machine, Apple iCloud, Carbonite, Barracuda, Dropbox, Google Drive, Microsoft One Drive, Box. Під час роботи хмарне сховище зберігає дані на мережевому накопичувачі аналогічно сервера електронної пошти. Ці дані можуть бути отримані власником файлу або, якщо дозволяють надані права, третьою стороною. Аналогічно транзакціях електронної пошти, існує безліч можливостей злому даних під час транспортування та при їх зберіганні на сервері.

Додатки передачі даних і широкомовні додатки включають SIP (англ. Session Initiation Protocol - протокол встановлення сеансу) - широкомовний протокол сигналізації, який використовується для управління сеансами передачі мультимедійних повідомлень, таких як голос та технологія VoIP (англ. Voice-over-Internet Protocol - передача голосу за допомогою інтернет-протоколу); IRC (англ. Internet Relay Chat - ретранслюючий інтернет-чат) - протокол прикладного рівня для передачі повідомлень у вигляді тексту, а також NNTP (англ. Network News Transfer Protocol - протокол передачі мережевих новин) - прикладний протокол, який використовується для передачі новин між новинними серверами і для публікації статей. OTT-провайдери, такі як Skype, Line, KakaoTalk, Viber, WhatsApp і інші, використовують налаштування додатки для доставки тексту, зображень та голосу через Інтернет з використанням технології VoIP.

Інші програми включають налаштування драйвери периферійних пристроїв для принтерів, сканерів, камер і т.д. Мережеві програми включають SNMP (англ. Simple Network Management Protocol - простий протокол мережевого управління) - протокол інтернет-стандарту для управління пристроями в IP-мережах, в тому числі маршрутизаторами, комутаторами, групою модемів та серверами; додатки BGP (англ. Border Gateway Protocol - протокол граничного шлюзу) в якості стандартизованих зовнішніх шлюзів для обміну інформацією про маршрутизацію і досяжності між автономними інтернет-системами, а також LDAP (англ. Lightweight Directory Access Protocol - полегшений протокол доступу до каталогів) для управління каталогами, що дозволяє обмінюватися інформацією про послуги, користувачів, системах, мережах та додатках, доступних у всіх приватних мережах і інтрамережі. Одна з особливостей додатків, пов'язаних з LDAP, полягає в тому, що один логін забезпечує доступ до декількох пристроїв, підключених через одну інтрамережу. До інших мережевих додатків відноситься CMIP (англ. Common Management Information Protocol - протокол загальної керуючої інформації).

Ще одним важливим мережевим додатком є DHCP (англ. Dynamic Host Configuration Protocol - протокол динамічної настройки вузла). DHCP використовується для запиту IP-адрес з мережевого сервера, починаючи від домашніх мереж і маршрутизаторів Wi-Fi до корпоративних мереж, університетських мереж і регіональних ISP (англ. Internet Service Providers - Інтернет-провайдерів). DHCP використовується для протоколів IPv4 і IPv6.

Якість обслуговування

Розглядаючи якість роботи мережі, оцінюють кілька факторів, а саме:

- швидкість передачі даних, тобто пропускну здатність;
- якість обслуговування;
- захищеність мережі і даних;
- конфіденційність користувачів.

На підставі наведених вище міркувань швидкість передачі даних оцінюється мільйонами біт в секунду (Мбіт/с). Крім того, QoS враховує ще кілька чинників, в тому числі затримку, якість звуку, стабільність мережі, уривчастість роботи або часті переривання для обслуговування, порушення синхронізації або підключення, низьку потужність сигналу, зниження швидкості роботи додатків і функціональну мережеву надмірність в умовах надзвичайної ситуації.

Для програм, файлів та перевірок, пов'язаних з безпекою, визначальним фактором є точність даних. Важливість факторів залежить від характеру корисного навантаження, що передається по мережі з комутацією пакетів. Навпаки, для голосу та відео, що включають додатки реального часу, ключовими є фактори, що впливають на час доставки пакета. Показники якості та їх вплив на різні додатки, такі як відео, голос, дані та текст, в якісному вигляді наведені в таблиці, показаної на рисунку 37. Хороший стан мережі, що характеризується прийнятною формою сигналу 610A IP-пакета з високою швидкістю передачі даних - це стан, що відрізняється мінімальними тимчасовими затримками, чистим сильним сигналом, відсутністю спотворень сигналу, стабільною роботою та відсутністю втрат передачі пакетів. Переривчасті мережі, що характеризуються формою сигналу 610B пакета зі зниженою швидкістю передачі даних з випадковими перериваннями, найбільш суттєво впливають на функції відео,

викликаючи повільне завантаження звичайного відео і неприйнятну якість онлайн-відео. Перевантажені мережі, що працюють зі зниженою ефективною швидкістю передачі даних з регулярними короткочасними перериваннями, для яких приклад форми сигналу 610C IP-пакета показаний на рисунку, не тільки серйозно погіршують якість відео через рвані переривчасті рухи, розмите зображення і неправильну передачу кольору і яскравості, але й починають також погіршувати якість звукового або голосового зв'язку через перекручування, відлуння і навіть випадання речень з бесіди або саундтрека. Однак в перевантажених мережах дані все-таки можуть бути доставлені при використанні протоколу TCP за допомогою повторних запитів на ретрансляцію.

Нестабільні мережі, що ілюструються формою сигналу IP-пакета 610D, демонструють низьку пропускну здатність з численними зупинками і непередбачуваною тривалістю передачі даних. Нестабільні мережі також містять пошкоджені IP-пакети, заштриховані темним кольором на рисунку із зображенням форми сигналу 610D, які при передачі на основі протоколу TCP повинні бути відправлені повторно, а при передачі на основі протоколу UDP просто відкидаються як пошкоджені або неправильні дані. При зниженні якості мережі до певного рівня навіть електронні листи стають переривчастими, та порушується синхронізація IMAP (англ. Internet Message Access Protocol - протокол доступу до електронної пошти Інтернету). Завдяки полегшеному формату даних, більшість SMS та текстових повідомлень будуть доставлені, хоча і з деякою затримкою, навіть при серйозному перевантаженні мережі, але вкладення завантажуватися не будуть. У нестабільних мережах може статися порушення роботи усіх програм і навіть припинення нормальної роботи комп'ютера або мобільного телефону в очікуванні доставки файлу. У таких випадках відео зависає, звук настільки змінюється, що стає нерозбірливим, VoIP-з'єднання багаторазово відкидаються, іноді навіть більше десятка разів протягом декількох хвилин виклику, а в деяких випадках не вдається підключитися взагалі. Аналогічно, передача електронних листів зупиняється або зависає, а круглі значки на екрані комп'ютера обертаються нескінченно. Індикатори виконання завдання зупиняються взагалі. Навіть текстові повідомлення відкидаються і приймають статус "не доставлених".

Незважаючи на те, що нестабільність мережі може виникати під впливом великої кількості факторів, в тому числі несправностей електроживлення серверів управління ключами і "суперточок присутності", перевантаження по числу викликів, передачі величезних файлів даних або UHD-фільмів, та під час активних DoS-атак (англ. Denial of Service - "відмова в обслуговуванні") на вибрані сервери або мережі, ключовими факторами, що використовуються для відстеження QoS мережі, є частота відкидання пакетів та затримка пакетів. Відкидання пакетів відбувається тоді, коли IP-пакет не може бути доставлений та по перевищенню ліміту часу визначається як "безсмертний", або, коли маршрутизатор або сервер виявляє помилку контрольної суми в заголовку IP-пакета. Якщо при передачі пакета використовується протокол UDP, цей пакет втрачається, при цьому прикладний рівень 7 повинен бути досить інтелектуальним, щоб дізнатися, що щось було втрачено. Якщо при передачі пакета на транспортному рівні 4 використовується протокол TCP, буде сформований запит для повторної передачі пакета, що збільшить навантаження на вже потенційно перевантажену мережу.

Інший фактор, що визначає QoS - затримку поширення - можна кількісно виміряти кількома способами: або як затримку передачі IP-пакета від вузла до вузла або, при односпрямованій передачі, від відправника до одержувача, або, в альтернативному варіанті, як час прийому-передачі від відправника до одержувача і назад до відправника. Порівняння впливу затримки поширення на доставку пакетів при використанні протоколів передачі даних UDP і TCP показано на рисунку 38. У міру збільшення міжмодової затримки поширення в мережі збільшується необхідний час прийому-передачі, наприклад, під час розмови з використанням технології VoIP. У разі UDP-передачі 621 час прийому-передачі збільшується лінійно з затримкою поширення. Оскільки великі затримки поширення пов'язані з більш високою частотою помилок по бітам, кількість втрачених UDP-пакетів збільшується, але оскільки UDP запитує повторну передачу відкинутих пакетів, лінійна залежність часу прийому-передачі від затримки зберігається при збільшенні затримки. TCP-передача 620 показує значно більш тривалий час прийому-передачі для кожного відправленого пакету, ніж UDP, через необхідність підтвердження встановлення зв'язку для підтвердження доставки пакета. Якщо частота помилок по бітам залишається низькою, і більшість пакетів не вимагають повторної відправки, то затримка поширення TCP збільшується лінійно з міжмодовою затримкою поширення, але з більшою швидкістю, тобто крутизою лінії TCP 620. Якщо, проте, мережа зв'язку стає нестабільною, в міру збільшення затримки поширення, то час прийому-передачі, пов'язаний з передачею по протоколу TCP, показаний лінією 622, експоненціально зростає через передбачену протоколом необхідність

повторної передачі відкинутих пакетів. Таким чином, застосування TCP протипоказано для додатків, чутливих до часу передачі, таких як технологія VoIP і онлайн-відео.

Оскільки весь пакетний зв'язок має статистичний характер, та в ньому немає двох пакетів з однаковим часом поширення, найкращим способом оцінки мережевої затримки для односпрямованої передачі є вимірювання часу прийому-передачі великої кількості IP-пакетів однакового розміру з подальшим розділенням навіпіл для оцінки затримки односпрямованої передачі. Затримка до 100 мс вважається відмінною, до 200 мс - дуже хорошою і до 300 мс - ще допустимою. Затримки поширення величиною 500 мс, що є характерною для ОТТ додатків в Інтернеті, створюють некомфортні умови для користувачів та заважають нормальній розмові. Зокрема, в процесі голосового зв'язку такі тривалі затримки поширення створюють "погане" звучання і можуть призводити до виникнення реверберації, створюючи "дзвінкий" або металевий звук, перериваючи розмову, коли інша сторона чекає відповіді на свій останній коментар, та в результаті мова стає спотвореною або нерозбірливою.

Для більшої ясності, затримка при односпрямованій передачі даних відрізняється від результату ring-тесту, що виконується за допомогою утиліти ICMP рівня 3 (наприклад, програми безкоштовного тестування мережі на сайті <http://www.speedtest.net>), частково через те, що ICMP - пакети зазвичай менш об'ємні, в порівнянні з реальними IP-пакетами, оскільки в ring-тесті не використовується функція TCP "запит на повторну відправку", а також тому, що немає ніякої гарантії, що при передачі по загальнодоступній мережі Інтернету маршрут ring-тесту буде відповідати фактичному маршруту пакета. По суті, виходить, що якщо сигнал ring-тесту приходить з великою затримкою, то це означає, що щось не так з мережею або з будь-яким зв'язком між пристроєм і мережею, наприклад, в маршрутизаторі Wi-Fi або на ділянці останньої милі, але хороший результат ring-тесту сам по собі не може гарантувати малу величину затримки поширення реального пакета.

З метою підвищення захищеності мережі часто використовуються методи шифрування та верифікації для запобігання злому, перехоплення або шпигунства. Але складне шифрування та протоколи шифрування з декількома ключами постійно підтверджують ідентичність співрозмовників, створюють додаткові затримки та тим самим збільшують ефективну затримку мережі, знижуючи якість обслуговування за рахунок підвищення захищеності.

Інформаційна безпека та конфіденційність інформації

До основних понять в галузі зв'язку відносяться інформаційна безпека і конфіденційність інформації. Ці два поняття, хоча вони і пов'язані, дещо відрізняються один від одного. "Інформаційна безпека, що включає мережеву безпеку, комп'ютерну безпеку та захищений зв'язок, охоплює методи, використовувані для контролю, перехоплення та запобігання несанкціонованого доступу, неправильного використання, модифікації або відмови від обслуговування комп'ютера або мережі зв'язку, доступних для мережі ресурсів або даних, що містяться в мережевих пристроях. Такі дані можуть включати в себе особисту інформацію, біометричні дані, фінансові записи, записи про здоров'я, особисті повідомлення та записи, а також приватні фотографії та відеозаписи. До мережевих пристроїв відносяться стільникові телефони, планшети, ноутбуки, стаціонарні комп'ютери, файлові сервери, сервери електронної пошти, веб-сервери, бази даних, сховища персональних даних, хмарні сховища, підключені до Інтернету побутові прилади, автомобілі з мережевими можливостями, а також загальнодоступні пристрої, що використовуються окремими особами, такі як касові апарати або POS-термінали, бензоколонки, банкомати і т.д.

Очевидно, що кіберзлочинці і комп'ютерні хакери, які намагаються отримати несанкціонований доступ до захищеної інформації, скоюють злочин. Якщо незаконно отримані дані містять особисту інформацію, атака також є порушенням недоторканності особистого життя жертви. Однак, з іншого боку, порушення конфіденційності можуть відбуватися без порушення закону і, фактично, не можуть бути попереджені. У сьогодиньньому мережевому світі несанкціоноване використання особистої інформації людини може здійснюватися без порушення вимог до інформаційної безпеки. У ряді випадків компанії, що збирають дані для будь-якої однієї мети, можуть продати свою базу даних іншим клієнтам, зацікавленим у використанні цих даних для іншої мети. Навіть коли Microsoft купила Hotmail, було добре відомо, що список розсилки був проданий рекламодавцям, зацікавленим в розсилці інформації потенційним клієнтам. Чи вважаються такі дії порушенням конфіденційності інформації, залишається спірним питанням.

"Конфіденційність інформації", що включає конфіденційність в Інтернеті, конфіденційність комп'ютера та приватне спілкування, має на увазі персональне право особистості або повноваження розпоряджатися своєю особистою і конфіденційною інформацією і її використанням, в тому числі на збір, зберігання, спільне використання або обмін інформацією з

іншими. Конфіденційна інформація може включати інформацію, що ідентифікує особу, в тому числі зріст, вагу, вік, відбитки пальців, групу крові, номер посвідчення водія, номер паспорту, номер карти соціального страхування або будь-яку особисту інформацію, корисну для ідентифікації людини без знання його імені. В майбутньому навіть карта ДНК людини може стати предметом юридичної реєстрації. Крім особистої ідентифікаційної інформації, існує неособиста конфіденційна інформація, яка може включати бренди одягу, які ми купуємо; веб-сайти, які ми часто відвідуємо; чи куримо ми; вживаємо алкоголь; чи володіємо пістолетом; на якому автомобілі їздимо; якими захворюваннями перехворіли протягом життя; чи є в нашій родині спадкові захворювання або недуги, і навіть яким людям ми подобаємося.

Ця конфіденційна інформація в поєднанні з публічними звітами, що стосуються особистого доходу, податків, майнових документів, судимості, порушень правил дорожнього руху, а також будь-яка інформація, розміщена на сайтах соціальних мереж, утворюють потужний набір даних для зацікавлених сторін. Технологія навмисного збору великих наборів даних, які фіксують демографічну, особисту, фінансову, біомедичну та поведінкову інформацію, а також аналіз цих даних за шаблонами, тенденціями та статистичній кореляції, сьогодні відома як "великі дані". Система охорони здоров'я, в тому числі страхові компанії, медичні працівники, фармацевтичні компанії та навіть адвокати підозрюваних у злочинній недбалості лікарів, виявляють дуже великий інтерес до особистої інформації, що зберігається у вигляді великих даних. Автомобільні компанії та виробники товарів широкого споживання також хочуть отримати доступ до таких баз даних, щоб направляти свою ринкову стратегію і рекламний бюджет. Навіть політики у ході недавніх виборів почали шукати великі дані, щоб краще зрозуміти думки виборців та предмети політичних суперечок, яких слід уникати.

Проблема конфіденційності інформації полягає не в тому, що сьогодні великі дані фіксують особисту інформацію (це вже стандартна процедура), а в тому, чи зберігає цей набір даних ваше ім'я або достатню особисту ідентифікаційну інформацію, щоб ідентифікувати вас, навіть не знаючи вашого імені. Наприклад, спочатку уряд США заявив, що особиста інформація, зібрана веб-сайтом healthcare.gov, використовувана для підписання Закону про доступне медичне обслуговування, буде знищена після створення приватних медичних рахунків. Потім недавно він відверто визнав, що стороння корпорація, яка допомагала уряду США виконувати збір даних, раніше підписала урядовий контракт, що дає їй право зберегти та використовувати зібрані дані, а це означає, що конфіденційні особисті дані, що стали відомими уряду США, в дійсності не є конфіденційними.

І нарешті, слід зазначити, що спостереження практикується як урядами, так і злочинними синдикатами, що використовують аналогічні технологічні методи. Звичайно, злочинці явно не мають законного права на збір таких даних, але випадки несанкціонованого спостереження з боку держави є більш сумнівною практикою, що істотно відрізняється в різних країнах. Наприклад, Агентство національної безпеки США неодноразово чинило тиск на Apple, Google, Microsoft та інші компанії, вимагаючи забезпечити доступ до їх хмар і баз даних. Прослуховувалися та перехоплювалися навіть розмови та заяви урядових офіційних осіб. На питання, чи відслідковує Skype, який є підрозділом Microsoft, вміст передач своїх абонентів, IT-директор Skype несподівано відповів: "Без коментарів".

Методи кіберзлочинності і кіберспостереження - Зосередившись на темі кібербезпеки, слід зазначити, що існує безліч засобів для отримання несанкціонованого доступу до даних пристрою, мережі і комп'ютера. Як приклад на рисунку 39 показано велику кількість шкідливих і хакерських технологій, що використовуються для здійснення кіберзлочинів і несанкціонованого проникнення в нібито захищені мережі.

Наприклад, людина з планшетом 33, підключеним до Інтернету, може захотіти зателефонувати на телефон 9 бізнес-офісу, відправити повідомлення на телевізор 36, подзвонити другу за місто по традиційній телефонній мережі з комутацією каналів на телефон 6, або завантажити файли з веб-сховища 20, або відправити електронну пошту через поштовий сервер 21A. Незважаючи на те, що всі ці програми - це звичайні додатки Інтернету і глобальних систем зв'язку, на різних ділянках мережі існує велика кількість можливостей для спостереження, кіберзлочинності, шахрайства та крадіжки особистих даних.

Наприклад, для планшета 33, підключеного до мережі через антену 18 стільникової радіостанції та базову станцію 17 LTE або через антену 26 радіосистеми ближньої дії та загальнодоступну базову станцію 100 мережі Wi-Fi, зловмисник може контролювати радіолінію. Аналогічно, виклик 28 LTE можна контролювати або аналізувати радіоприймачем-перехоплювачем або сніффером 632. Той же перехоплювач 632 може бути налаштований для контролю передачі даних по каналу Wi-Fi 29 і на приймаючій стороні по кабелю 105 між кабелем термінальної системи кабельних модемів (CMTS) 101 і кабельним модемом 103.

У деяких випадках виклик LTE також може бути перехоплений шахрайською піратською вишкою 638 шляхом перенаправлення каналу зв'язку 639 між планшетом 38 та вишкою стільникового зв'язку 18. Передача даних по мережі з комутацією пакетів в маршрутизатор 27, сервер 21A та сервер 21B, а також хмарне сховище 20, - також вразливості для атаки "людина посередині" 630. Підключення до дротової лінії 637 можуть перехоплювати виклики на лінії традиційної телефонної мережі (POTS) від шлюзу 3 абонентської мережі зв'язку (PSTN) до телефону 6, а також на лінії корпоративної АТС від сервера АТС 8 до офісного телефону 9.

Завдяки ряду вразливостей в захисті, шпигунське ПЗ 631 може автоматично встановитись на планшет 33, на маршрутизатор 27, на PSTN-міст 3, на хмарне сховище 20, на термінальну систему кабельних модемів (CMTS) 101 або на стаціонарний комп'ютер 36. Троянська програма 634 може встановитися на планшет 33 або стаціонарний комп'ютер 36 для фішингу паролів. Шкідлива програма 636 також може використовуватися для атаки на стаціонарний комп'ютер 36, особливо якщо на комп'ютері встановлена операційна система Microsoft з ActiveX керуючими елементами. Нарешті, для запуску атак типу "відмова в обслуговуванні" вірус 633 може атакувати будь-яку кількість мережевих пристроїв, включаючи сервери з номерами 21A, 21B і 21C, стаціонарний комп'ютер 36 та планшет 33.

На рисунку 40 графічна частина спрощена та відображається стосовно того, в якій частині мережі зв'язку та інфраструктури працює кожен вид шкідливого ПЗ. У показаному хмарі 22, що містить сервер 21A, волоконно-оптичну лінію зв'язку 23 та сервер 21B, кібератаки можуть включати вірус 633, атаку "людина посередині" 630, урядовий нагляд 640 і атаку "відмова в обслуговуванні" 641. "Остання миля" мережі зв'язку надає ще більш великі можливості для шкідливих програм і кібератак і ділиться на три секції: місцева телефонна компанія/мережа, "остання миля" та пристрій. Місцева телефонна компанія/мережа, згідно рисунку, містить високошвидкісний волоконно-оптичний кабель 24, маршрутизатор 27, термінальну систему кабельних модемів (CMTS) 101, коаксіальний/волоконно-оптичний кабель 105, кабельний модем 103, антену Wi-Fi 26 та вишку радіозв'язку LTE 25. У цій частині мережі може застосовуватися перехоплення радіоканалу 632, шпигунське ПЗ 631, вірус 633 і атака "людина посередині" 630.

В останній ланці - місцевому підключенні до пристрою - підключення до мережі включає дротову лінію 104, з'єднання Wi-Fi 29 та з'єднання LTE/радіо 28, що піддається впливу шпигунського ПЗ 631, перехоплення радіоканалу 632, прослуховування 637 та піратської вишки 638. Сам пристрій, що включає, наприклад, планшет 33, ноутбук 35, стаціонарний комп'ютер 36, а можливо також смартфони, смарт-телевізори, POS-термінали і т.д., піддається безлічі атак, включаючи шпигунське ПЗ 631, троянську 634, вірусну 633 та шкідливу 636 програми.

Такі методи спостереження і шпигунські пристрої легкодоступні на комерційному та онлайн-ринку. На рисунку 41A показані два таких пристрої: пристрій 650, що використовується для контролю трафіку в локальних мережах Ethernet, та пристрій 651, що забезпечує ті ж функції для контролю даних Wi-Fi. Два доступних для придбання пристроїв, 652 і 653, які використовуються для контролю стільникового зв'язку, показані на рисунку 41B. Незважаючи на те, що на графіку мережі на рисунку 39 аналіз пакетів 632 хмарних з'єднань 23 волоконно-оптичним кабелем не ідентифікований як загроза, під час досліджень стало очевидним, що неінвазійне перехоплення даних для оптичної лінії зв'язку, тобто таке, де волоконно-оптичний кабель не потрібно обрізати або хоча б тимчасово порушувати його нормальну роботу, вже існує. Згідно рисунку 41C, пристрій 655 виконує аналіз трафіку волоконно-оптичної лінії зв'язку шляхом захоплення світлового витоку на різкому вигині волоконно-оптичного кабелю 656. Якщо захисна оболонка видалена заздалегідь, затиск волоконно-оптичного кабелю 656 клемою в пристрої 655, забезпечує малий радіус розвороту волоконно-оптичного кабелю 656, де відбувається витік світла 657 в фотодатчик 659, що передається по електронному кабелю 660 в ноутбук 661 для аналізу.

Крім використання методів злому та спостереження, існує велика кількість комерційних легкодоступних шпигунських програм для контролю розмов по стільниковому телефону та передачі даних через Інтернет. У таблиці, наведеної на рисунку 42, порівнюються властивості 10 кращих шпигунських програм, які рекламують здатність успішно стежити за своїми співробітниками, дітьми та чоловіком (дружиною). На подив широкий вибір функцій, що включає відстеження дзвінків, фотографій та відео, текстових повідомлень SMS/MMS, сторонніх миттєвих повідомлень, електронних листів, відстеження координат GPS, використання Інтернету, адресної книги, подій календаря, прослуховування, управління додатками і навіть функції віддаленого управління, в сукупності містить переконливу кількість способів порушення конфіденційності інформації.

Насправді кібератаки стали настільки частими, що їх відстежують щодня. Один з таких сайтів відстеження, показаний на рисунку 43, відображає прогалини в захисті і кібератаки на карті світу, в тому числі місце, тривалість та тип атаки. Для запуску кібератаки зазвичай задіюється кілька етапів або поєднань методів, у тому числі:

- 5 - Аналіз IP-пакетів;
- Опитування портів;
- Профілювання;
- Самозванці;
- Перехоплення пакетів;
- 10 - Кіберінфекції;
- Спостереження;
- Піратське адміністрування.

Аналіз IP-пакетів - Використовуючи пристрої радіоконтролю, кіберзлочинець може отримати значний обсяг інформації про користувача, його транзакції і його облікових записів. Згідно рисунку 44, вміст IP-пакета можна отримати або проаналізувати в будь-якій точці шляху між двома користувачами. Наприклад, коли користувач 675A відправляє файл, наприклад, фотографію або текст в 670 IP-пакеті зі свого ноутбука 35 на мобільний телефон 32 свого друга 675B, кіберпірат 630 може виявити IP-пакет в одному з декількох місць, перехопивши останнє посилання 673A відправника, перехопивши локальну мережу 672A відправника, контролюючи хмару 671, перехопивши дзвінок одержувача по місцевому телефонному дзвінку 672B або перехопивши останнє повідомлення 673B одержувача. У перехопленому 670 IP-пакеті доступні для спостереження наступні дані: MAC-адреси рівня 2 задіяні в інформаційному обміні пристроїв, адреси рівня 3 відправника приймаючої сторони, тобто одержувача пакета, включаючи використовуваний протокол передачі, наприклад, UDP, TCP і т.д. IP-пакет також містить номери портів рівня 4 відправляючих та приймаючих пристроїв, що потенційно дозволяють визначити тип запитуваної послуги, а також сам файл даних. Якщо файл не зашифрований, дані, що містяться в файлі, також можуть бути легко прочитані кіберпіратом 630.

Якщо корисне навантаження не зашифроване, текстова інформація, наприклад, номери облікових записів, логіни та паролі, можуть бути прочитані і, якщо вони представляють цінність, вкрадені та змінені в злочинних цілях. Якщо корисне навантаження містить відео або піктографічну інформацію, потрібно виконати деяку додаткову роботу, щоб визначити, який формат прикладного рівня 6 використовується для передачі вмісту, але як тільки він буде ідентифікований, вміст можна буде переглянути, опублікувати або навіть використовувати для шантажу однієї або обох сторін інформаційного обміну. Така кібератака називається атакою "людина посередині", тому що кіберпірат особисто не знайомий ні з однією з сторін інформаційного обміну.

Як було зазначено раніше, маршрутизація IP-пакетів в хмарі непередбачувана, тому контроль хмари 671 є більш складним, тому що кіберпірат 630 повинен захопити важливу інформацію IP-пакета при першій зустрічі з ним, оскільки наступні пакети можуть не слідувати по тому ж маршруту, та проаналізувати пакет. Перехоплення даних на останній милі підвищує ймовірність спостереження послідовності пов'язаних пакетів з однієї і тієї ж розмови, оскільки місцеві маршрутизатори зазвичай слідують продиктованій таблиці маршрутизації, у крайньому випадку, до тих пір, поки пакети не досягнуть точки присутності (POP) за межами області обслуговування власного оператора клієнта. Наприклад, клієнт Comcast, швидше за все, буде передавати IP-пакети в ланцюжок маршрутизації доти, повністю використовуючи Comcast мережу, поки пакет не буде географічно переміщений за межі доступу і області обслуговування клієнтів Comcast.

Якщо послідовність пакетів, що передаються між одними та тими ж двома IP-адресами, спостерігається протягом досить тривалого часу, всю розмову можна відтворити по частинах. Наприклад, якщо текстові SMS-повідомлення передаються по одній і тій же мережі на ділянці останньої милі, кіберпірат 630 може ідентифікувати через IP-адреси і номери портів, що кілька IP-пакетів, що передають текст, являють собою розмову між одними та тими ж двома пристроями, тобто між стільниковим телефоном 32 і ноутбуком 35. Тому навіть якщо номер облікового запису та пароль були в тексті різних повідомлень або були відправлені в повному обсязі та поширилися по багатьом пакетам, збіг ідентифікаторів пакетів як і раніше дозволяє кіберпірату відтворити розмову та вкрати інформацію про обліковий запис. Після того, як дані облікового запису будуть вкрадені, зловмисники можуть або перевести гроші в офшорний банк, або навіть узурпувати повноваження облікового запису, змінивши пароль облікового запису та параметри безпеки, тобто використовуючи крадіжку особистих даних на тимчасовій основі.

Навіть якщо корисне навантаження зашифроване, інша частина 670 IP-пакета, включаючи IP-адреси і номери портів, не шифрується. Після багаторазового аналізу великої кількості IP-пакетів кіберпірат, що володіє достатньою обчислювальною потужністю, може методом "грубої сили" систематично перевіряти кожну комбінацію, поки не зламає пароль шифрування. Як тільки ключ буде зламаний, цей пакет та всі наступні пакети можуть бути розшифровані та використані кіберпіратом 630. Імовірність злому пароля входу в систему шляхом "вгадування пароля" значно підвищується, якщо аналіз пакетів поєднується з описаним нижче "профілюванням" користувача і облікового запису. Слід звернути увагу на те, що атаки "людина посередині" на пристрої зв'язку зазвичай не задіюються, тому що кіберпірат не має до них прямого доступу.

Опитування портів - Ще один спосіб злому пристрою полягає в тому, що за його IP-адресою проводиться опитування великої кількості портів рівня 4 та перевірка наявності відповіді на направлені запити. Згідно рисунку 45, коли кіберпірат 680 визначає за результатами аналізу пакету або іншими засобами, що мобільний телефон 32 з IP-адресою "CP" є цільовим пристроєм, кіберпірат 680 запускає послідовність опитувань портів на стільниковому телефоні 32 в пошуках незахищеного або відкритого порту, порту сервісного та технічного обслуговування або шляхів обходу захисту. Поки хакерська програма опитування систематично перевіряє кожен номер порту, атаки зазвичай зосереджені на таких сумнозвісних портах, як порт №7 для ping-тесту, порт №21 для FTP, порт №23 для емуляції терміналу telnet, порт №25 для простої електронної пошти і т.д. Відповідно до рисунку, послідовно відправляючи пакети 680A, 680B, 680C і 680D, кіберпірат 660 очікує відповіді стільникового телефону 32, який в даному прикладі стався за запитом 680D. Кожен раз, коли відправляється відповідь, пірат дізнається більше про операційну систему цільового пристрою.

В процесі опитування портів кіберпірат 630 не хоче розкривати свою реальну ідентифікацію, тому він використовує приховану псевдо-адресу, символічно позначену тут як "PA", для прийому повідомлень, що не простежується до його персонального пристрою. В якості альтернативи, кіберзлочинці можуть використовувати крадений комп'ютер і обліковий запис, при цьому ситуація виглядає так, ніби хтось намагається зламати цільовий пристрій, але, якщо його відстежити, це приведе слідство до невинної людини, а не до зловмисника.

Профілювання - Профілювання користувачів і облікових записів - це процес, в ході якого кіберпірат проводить дослідження з використанням загальнодоступної інформації, щоб більше дізнатися про цільовий об'єкт, його облікових записів і його особистої історії, щоб зламати його паролі, ідентифікувати облікові записи та визначити активи. Як тільки хакер отримує IP-адресу цілі за результатами аналізу трафіку або іншими засобами, він може скористатися утилітою traceroute для пошуку DNS-сервера облікового запису пристрою. Потім, використовуючи функцію "Who is" (Хто це) в Інтернеті, можна дізнатися ім'я власника облікового запису. Потім в процесі профілювання кіберзлочинець здійснює пошук в Інтернеті для збору всієї наявної інформації про власника облікового запису. В якості джерел інформації використовуються такі загальнодоступні записи, як документи про власність, дані про реєстрацію автомобілів, шлюбні і розлучення, заставних правах на майно для забезпечення сплати податку, паркувальних квитках, порушеннях правил дорожнього руху, судимості і т.д. У багатьох випадках веб-сайти університетів та професійних товариств також містять домашню адресу, адреси електронної пошти, номери телефонів і дату народження людини. Вивчаючи сайти соціальних мереж, такі як Facebook, LinkedIn, Twitter і інші, кіберзлочинець може зібрати значну докладну інформацію, в тому числі про родину і друзів, імена домашніх тварин, адреси колишнього місця проживання, відомості про однокласників, головні події в житті, а також фотографії та відеофайли, в тому числі що містять делікатні подробиці, сімейні таємниці і дані про особистих ворогів.

Наступним кроком кіберпірата є використання даних цього профілю для "вгадування" паролів користувача на основі його профілю для злому цільового пристрою та інших облікових записів цієї людини. Після того, як кіберзлочинець підбере пароль одного пристрою, велика ймовірність того, що він зможе підійти і до інших облікових записів, тому що люди схильні використовувати одні й ті ж самі паролі для зручності їх запам'ятовування. У цей момент може з'явитися можливість вкрати ідентифікаційні дані людини, перевести гроші, зробити його мішенню поліцейських розслідувань та по суті зруйнувати чиєсь життя, викравши все його майно. Наприклад, як описано у вступній частині цього документа, зібравши довгий список паролів з вкрадених облікових записів, кіберзлочинці використовували ці паролі для незаконної покупки преміум квитків на концерти та спортивні події на кілька мільйонів доларів з використанням тих же паролів та логінів.

Самозванці - Коли кіберпірат видає себе за когось іншого, він не використовує або використовує незаконно отримані захищені облікові дані, щоб отримати доступ до повідомлень

або файлів під хибним приводом того, що він є уповноваженим агентом або пристроєм, кіберпірат діє як "самозванець". Кібератака типу "самозванець" може виникати, коли у кіберзлочинця є достатня інформація або доступ до облікового запису користувача, щоб скомпрометувати обліковий запис жертви, відправляти від її імені повідомлення і навмисно спотворювати їх зміст від імені власника зламаного облікового запису. Нещодавно, наприклад, у близької знайомої одного з винахідників було зламано особистий обліковий запис месенджера "Line". Отримавши контроль над обліковим записом, кіберзлочинець відправив повідомлення її друзям з неправдивою інформацією про те, що "вона потрапила в автомобільну аварію, і їй терміново потрібні гроші в борг", вказавши платіжні реквізити для відправки грошей. Не знаючи, що обліковий запис був зламаний, її друзі подумали, що запит реальний, погодилися заплатити викуп. Щоб уникнути підозр, у кожному відправленому запиті були вказані суми менше 1000 доларів США. На щастя, перш ніж відправляти гроші, один з друзів подзвонив їй, щоб перевірити ще раз платіжні реквізити, і, таким чином, виявив шахрайство. Якби не цей дзвінок, ніхто ніколи не дізнався б, що запити були від самозванця, а власник облікового запису Line ніколи не дізнався б про те, що гроші були відправлені або навіть запитані.

Інша форма введення в оману виникає, коли пристрою надані привілеї безпеки та воно має можливість обмінюватися інформацією з сервером або іншим мережевим пристроєм, та при цьому якимось чином пристрою кіберпірата вдається видавати себе за авторизований сервер, в результаті чого пристрій жертви охоче передає файли і інформацію на піратський сервер, не усвідомлюючи, що цей сервер є самозванцем. За наявними даними, цей метод використовувався, щоб схилити знаменитостей до резервного копіювання особистих файлів зображень за допомогою iCloud, промовчавши, що ця резервна хмара була самозванцем.

Інша форма самозванця виникає, коли хтось, маючи фізичний доступ до особистого телефону або відкритого браузеру людини, виконує такі шахрайські операції, як відправка електронної пошти, відповідь на телефонний дзвінок, текстове повідомлення, надіслане з облікового запису або пристрою іншої особи. Одержувач вважає, що він підключений до відомого пристрою або облікового запису, і що особа, яка керує цим пристроєм або обліковим записом, є його власником. Самозванець може виявитися жартівником, який, наприклад, виставляє делікатні коментарі в Facebook, або його дії можуть носити більш особистий характер, коли чийсь чоловік відповідає на особисті виклики або перехоплює конфіденційні текстові повідомлення особистого характеру. Результатом цього несанкціонованого доступу можуть бути ревності, розлучення та судове переслідування. Залишаючи пристрій тимчасово неконтрольованим в офісі або кафе, наприклад, виходячи в туалет, ви піддаєтеся додатковому ризику, надаючи самозванцю можливість швидкого доступу до особистої або корпоративної інформації, відправки несанкціонованих повідомлень електронної пошти, передачі файлів або завантаження будь-якої шкідливої програми в пристрій, як описано в наступному розділі, що називається "Кіберінфекції".

Кібератака з боку самозванця також має велике значення при крадіжці пристрою. У таких випадках, навіть незважаючи на те, що пристрій вийшов з системи, у злодія достатньо часу, щоб підібрати логін. Функція "find my computer" (знайти мій комп'ютер), яка повинна знайти вкрадений пристрій в мережі та стерти файли комп'ютера при першому вході кіберпірата в систему на цьому пристрої, більше не працює, тому що високотехнологічні злочинці сьогодні знають, як активувати тільки саме пристрій без стільникового або Wi-Fi-з'єднання. Цей ризик особливо великий для стільникових телефонів, де безпеку входу забезпечує простий чотиризначний ідентифікаційний номер або PIN-код. Підбір PIN-коду - це всього лише питання часу, оскільки є тільки 9999 можливих комбінацій.

Ключовою проблемою для захисту будь-якого пристрою є запобігання доступу для самозванців. Запобігання втручання самозванців вимагає надійних засобів автентифікації особистості користувача через певні проміжки часу та підтвердження того, що тільки вони мають право на доступ до інформації та привілеїв, яких вони потребують. Захист пристрою часто є найслабшою ланкою в цьому ланцюжку. Після подолання захисту пристрою необхідність в надійній мережевій безпеці є спірною.

Перехоплення пакетів - Перехоплення пакетів - це кібератака, в ході якої нормальний потік пакетів через мережу перенаправляється через шкідливий пристрій. Цей приклад показаний на рисунку 46, де ноутбук 35 з IP-адресою "NB" і ад-нос-портом №9999 відправляє файл у вигляді 670 IP-пакета на стільниковий телефон (не показаний), який має IP-адресу "CP" і FTP- порт даних №20. При нормальних умовах 670 IP-пакет буде передаватися по маршруту від ноутбука 35 до маршрутизатора 26 Wi-Fi і до маршрутизатора 27 з високошвидкісним дротовим підключенням 24 до сервера 22A в хмарі.

Якщо, проте, цілісність маршрутизатора 27 була порушена кібератакою з боку кіберпірата 630, IP-пакет 670 може бути перезаписаний в IP-пакет 686A, який для ясності показаний в скороченому вигляді із зазначенням тільки IP-адрес і номерів портів. Для перенаправлення IP-пакета адреса і номер порту одержувача стільникового телефону змінюються на відповідні дані кіберпіратського пристрою 630, а саме - на IP-адресу "РА" та порт №20000. Після цього кіберпіратський пристрій 630 отримує будь-яку необхідну йому інформацію з корисного навантаження IP-пакета і, можливо, змінює вміст корисного навантаження IP-пакета. Шахрайське корисне навантаження може використовуватися для здійснення ряду шахрайських злочинів, для збору інформації або для завантаження шкідливого ПЗ в стільниковий телефон, як описано нижче в розділі "Кіберінфекції".

Потім зламаний пакет IP-пакет 686B перевизначається так, щоб він виглядав як вихідний IP-пакет 670 з IP-адресою відправника "NB" з порту №9999, відправлений на IP-адресу стільникового телефону "CP" в порт №20, за винятком того, що пакет передається по провідному з'єднанню 685B замість провідного з'єднання 24. В якості альтернативи, перехоплений IP-пакет може бути повернутий на зламаний маршрутизатор 27, а потім відправлений у хмару через дротове з'єднання 24. Щоб максимізувати злочинну вигоду від перехоплення пакету, кіберпірату 630 необхідно приховати свої ідентифікаційні дані при перехопленні пакетів, та з цієї причини вони маскують справжню маршрутизацію IP-пакета, при цьому навіть функція "traceroute" ICMP рівня 3 буде зазнавати труднощів при ідентифікації істинного шляху передачі даних. Якщо, проте, це перехоплення помітно збільшує затримку при маршрутизації пакетів, ця незвичайна затримка може бути приводом для дослідження з боку оператора мережі.

Кіберінфекції - Однією з найпідступніших категорій кібератак є "кіберінфекції" - установка на цільові пристрої або в мережу шкідливого ПЗ, за допомогою якого можна збирати інформацію, здійснювати шахрайські дії, перенаправляти трафік, інфікувати інші пристрої, порушувати роботу або вимикати систему або викликати відмову в обслуговуванні. Кіберінфекція може поширюватися через електронні листи, файли, веб-сайти, системні розширення, прикладні програми або по мережах. Один з основних класів шкідливих програм - "шпигунське ПЗ" - описаний в таблиці на рисунку 42, збирає всі види інформації про транзакції і направляє кіберпірату. У разі "фішингу" веб-сторінка або оболонка додатку, яка відображається як знайома сторінка входу в систему, запитує логін облікового запису або особисту інформацію, а потім передає цю інформацію кіберпірату. Інші шкідливі інфекції можуть управляти обладнанням, наприклад, управляти маршрутизатором для виконання вищезазначеного перехоплення пакетів. У цих випадках кіберпірат намагається отримати інформацію або отримати вигоду для досягнення своїх цілей.

Інший клас кіберінфекцій, що містить віруси, шкідливі та троянські програми, призначений для перезапису критичних файлів або для повторного виконання беззмістовних функцій, щоб позбавити пристрій можливості виконання своїх звичайних завдань. В основному, щоб заборонити обслуговування, погіршити якість роботи або повністю знищити пристрій. Ці зловмисні інфекції за своєю суттю руйнівні та використовуються з метою помсти, щоб порушити нормальну роботу бізнесу конкурента або просто заради забави хакером, охочим побачити, чи можливо це.

Спостереження - Підслуховування та стеження виходять за рамки кіберзлочинності. У таких випадках приватного детектива або знайомого наймають або примушують встановити пристрій або програму в особисті пристрої суб'єкта, що представляє інтерес, для здійснення контролю над його розмовами, обміном даними та місцем розташування. Ризик бути спійманим збільшується, тому що детектив повинен отримати тимчасовий доступ до цільового пристрою так, щоб суб'єкт про це не дізнався. Наприклад, у продажу є SIM-карти, які можуть копіювати права доступу до мережі телефону, але одночасно передається інформація кіберзлочинця, який контролює виклики та трафік даних жертви.

До інших форм спостереження відноситься використання підпільних відеокamer для спостереження за кожною дією та телефонним дзвінком людини, багато в чому аналогічно казино. За допомогою відеоконтролю, пароль або PIN-код пристрою можна дізнатися, просто спостерігаючи за натисканнями клавіш користувачем під час процесу входу в систему. При досить великій кількості камер на місці спостереження, в кінцевому рахунку, рано чи пізно процес входу в систему буде записаний. Щоб отримати доступ до мережі камер, не викликаючи підозр, кіберпірат може зламати існуючу систему відеоспостереження на будівлях, в магазинах або на вулицях, а також, через доступ до чужої мережі, стежити за поведінкою нічого непідозрюючих жертв. Об'єднання відеоспостереження з аналізом пакетів надає ще більш повний набір даних для подальшого запуску кібератак.

Піратське адміністрування (несанкціоноване проникнення) - Ще один засіб, завдяки якому кіберпірати можуть отримати інформацію - це злом і отримання доступу до прав системного адміністратора пристрою, сервера або мережі. Таким чином, замість отримання несанкціонованого доступу до облікового запису одного користувача, зламавши логін адміністратора системи, кіберпірат отримує набагато ширший доступ та привілеї без відома тих, хто використовує систему. Оскільки системний адміністратор виступає системою в якості поліцейського, ніхто не може припинити його злочинну діяльність - по суті, в системі або мережі з корумпованою адміністрацією ніхто не може контролювати поліцію.

Висновок - Повсюдне поширення та сумісність, які Інтернет, мережі з комутацією пакетів і майже загальне визнання семирівневої моделі OSI (Open Source Initiative), за останні двадцять років дозволили розширити глобальний зв'язок в небувалому масштабі, з'єднавши широкий діапазон пристроїв від смартфона до планшетів, комп'ютерів, інтелектуальних телевізорів, автомобілів і навіть побутової техніки та лампочок. Глобальне визнання інтернет-протоколу (IP) в якості основи для підключення до Ethernet, стільникового зв'язку, Wi-Fi і кабельного телебачення не тільки уніфікував зв'язок, але й значно спростили проблему для хакерів і кіберзлочинців, які намагаються проникнути в максимальну можливу кількість пристроїв та систем. З огляду на безліч програмних і апаратних методів, доступних зараз для атаки сучасних мереж зв'язку, очевидно, що одного методу захисту недостатньо для повної безпеки. Замість цього необхідний системний підхід до захисту кожного пристрою, останньої ланки, місцевої телефонної компанії/мережі та хмарної мережі, щоб забезпечити їх захист від складних кібератак. Використовувані методи повинні забезпечувати внутрішню інформаційну безпеку і конфіденційність інформації без шкоди для якості обслуговування, затримки мережі, якості відео або звуку. Незважаючи на те, що шифрування має залишатися важливим елементом розробки наступного покоління безпечної передачі та зберігання даних, безпека мережі не повинна спиратися виключно на методи шифрування.

Короткий виклад суті винаходу

Відповідно до цього винаходу дані (які в широкому сенсі включають в себе текстові, аудіо-, відео-, графічні та всі інші види цифрової інформації або файлів) передаються через динамічну захищену комунікаційну мережу та протокол (SDNP) або "хмару". Хмара SDNP включає в себе безліч "вузлів", іноді званих "медіа-вузлами", які індивідуально розміщуються на серверах або інших типах комп'ютерів або цифрового обладнання (в сукупності іменованих тут як "сервери"), розташованих в будь-якій точці світу. Можливе розміщення двох і більше вузлів на одному сервері. Як правило, дані передаються між медіа-вузлами світлом по волоконно-оптичним кабелям, радіохвилями в високочастотному або надвисокочастотному діапазоні, електричними сигналами по мідних дротах або коаксіальному кабелю або по каналах супутникового зв'язку, але в широкому сенсі винахід включає в себе будь-які засоби, за допомогою яких цифрові дані можуть передаватися з однієї точки в іншу. Мережа SDNP включає хмару SDNP, а також канали "останньої милі" між хмарию SDNP і клієнтськими пристроями, такими як мобільні телефони, планшети, ноутбуки та стаціонарні комп'ютери, мобільні споживчі електронні пристрої, а також пристрої та побутові прилади Інтернету речей, автомобілі та інші транспортні засоби. Системи зв'язку останньої милі також включають вежі стільникового телефонного зв'язку, кабель або оптоволокну, прокладені в будинку, та загальнодоступні маршрутизатори Wi-Fi.

При передачі між медіа-вузлами в хмарі SDNP дані представлені в вигляді "пакетів", дискретних ланцюжків цифрових бітів, які можуть мати фіксовану або змінну довжину, при цьому дані замасковані з використанням таких методів: скремблювання, шифрування або розділення - або відповідних зворотних процесів: дескремблювання, дешифрування та змішування. (Примітка. У даному документі, якщо із контексту не слідує іншого, слово "або" використовується в його кон'юнктивному (та/або) сенсі.)

Скремблювання має на увазі зміну порядку проходження даних в пакеті. Наприклад, для сегментів даних А, В і С, які в початковому пакеті слідує в порядку АВС, порядок проходження зміниться на САВ. Зворотна по відношенню до скремблювання операція називається "дескремблювання" і має на увазі зміну порядку проходження даних в пакеті на той, в якому він спочатку з'явився - АВС в наведеному вище прикладі. Об'єднана операція дескремблювання, а потім скремблювання пакета даних називається "повторним скремблюванням". При повторному скремблюванні пакета, який раніше був скремблований, цей пакет може бути скремблований тим же самим або іншим способом по відношенню до попередньої операції скремблювання.

Друга операція - "шифрування" - це кодування даних в пакеті і їх подання у формі, званої зашифрованим текстом, яка може бути зрозуміла тільки відправнику та всім іншим авторизованим сторонам, а також того, хто повинен виконувати зворотну операцію -

"дешифрування". Об'єднана операція дешифрування пакета даних із зашифрованим текстом і його подальшого шифрування, як правило, але не обов'язково з використанням методу, який відрізняється від методу, який використовується при його попередньому шифруванні, в даному документі називається "повторним шифруванням".

Третя операція - "розділення", згідно з назвою, означає поділ пакета на два або більше пакети меншого розміру. Зворотна операція - "змішування" - визначається як об'єднання двох або більше розділених пакетів назад в один пакет. Розділення пакета, який раніше був розділений, а потім змішаний, може бути виконано таким же чином або може відрізнитися від попередньої операції розділення. Порядок виконання операцій є оборотним, при цьому розділення може бути скасовано шляхом змішування, і, навпаки, змішування декількох вхідних елементів в один вихідний може бути скасовано шляхом розділення для відновлення складових компонентів. (Примітка. Оскільки скремблювання і дескремблювання, шифрування і дешифрування, а також розділення та змішування є зворотними процесами, для виконання зворотної операції необхідно тільки знання алгоритму або методу, який використовувався для прямої операції. Отже, коли мова йде про конкретний алгоритм скремблювання, шифрування або розділення, має бути зрозуміло, що знання цього алгоритму дозволяє виконати зворотний процес).

У відповідності до даного винаходу пакет даних, який проходить через хмару SDNP, скремблюється або шифрується, або ж над ним виконується одна або обидві ці операції в поєднанні з розділенням. Крім того, в пакет можуть додаватися "сміттєві" (тобто беззмстовні) дані, щоб зробити пакет більш складним для дешифрування, або привести довжину пакета у відповідність до встановлених вимог. Крім того, пакет може бути підданий статистичному аналізу, тобто розділений на окремі частини. Професійною мовою комп'ютерних фахівців проводити синтаксичний аналіз - це значить розділити оператор комп'ютерної мови, комп'ютерну команду або файл даних на частини, які можуть бути корисні для комп'ютера. Синтаксичний аналіз може також використовуватися для ускладнення розуміння мети команди або пакета даних, або для упаковки даних в пакети, що мають певну довжину.

Незважаючи на те, що формат пакетів даних відповідає інтернет-протоколу, в хмарі SDNP адреси медіа-вузлів не є стандартними інтернет-адресами, тобто вони не можуть бути ідентифіковані будь-яким сервером доменних імен Інтернету. Отже, незважаючи на те, що медіа-вузли можуть технічно отримувати пакети даних через Інтернет, ці медіа-вузли не будуть розпізнавати адреси або відповідати на запити. Більш того, навіть якщо б користувачам Інтернету потрібно було зв'язатися з медіа-вузлом, вони не змогли б отримати доступ або переглянути дані всередині медіа-вузла, тому що медіа-вузол може розпізнати їх як самозванців, які не мають необхідних ідентифікаційних облікових даних як медіа-вузла SDNP. Зокрема, якщо медіа-вузол не зареєстрований як діючий вузол SDNP, що працює на сервері, що відповідає вимогам сервера імен SDNP або його еквівалентної функції, пакети даних, що відправляються з цього вузла на інші медіа-вузли SDNP, будуть ігноруватися та відкидатися. Аналогічним чином, тільки клієнти, зареєстровані на сервері імен SDNP, можуть звертатися до медіа-вузла SDNP. Подібно незареєстрованим серверам, пакети даних, отримані з джерел, які не є зареєстрованими клієнтами SDNP, будуть ігноруватися і негайно відкидатися.

В порівняно простому варіанті здійснення, званому "одномаршрутний", пакет даних проходить по єдиному шляху через ряд медіа-вузлів у хмарі SDNP, при цьому він скремблюється в медіа-вузлі, в якому він входить в хмару і дескремблюється в медіа-вузлі, в якому він виходить з хмари (ці два вузли називаються "шлюзовими вузлами" або "шлюзовими медіа-вузлами"). У кілька більш складному варіанті здійснення пакет повторно скремблюється на кожному медіа-вузлі з використанням методу скремблювання, відмінного від того, який використовувався на попередньому медіа-вузлі. В інших варіантах здійснення пакет також шифрується в шлюзовому вузлі, в якому він входить в хмару і дешифрується в шлюзовому вузлі, в якому він виходить з хмари, і, крім того, пакет може бути повторно зашифрований в кожному медіа-вузлі, через який він проходить в хмарі. Оскільки даний вузол при скремблюванні або шифруванні пакету кожен раз використовує один та той же алгоритм, цей варіант здійснення називається "статичним" скремблюванням і шифруванням.

У разі, коли виконується дві і більше операції перетворення пакета, наприклад, він скремблюється і шифрується, зворотні операції рекомендується виконувати в протилежному порядку, тобто в зворотній послідовності. Наприклад, якщо пакет скремблюється, а потім шифрується до виходу з медіа-вузла, то при надходженні на наступний медіа-вузол, він спочатку дешифрується, а потім дескремблюється. Пакет відтворюється в своїй вихідній формі тільки в тому випадку, якщо він знаходиться в медіа-вузлі. Поки пакет передається між медіа-

вузлами, він знаходиться в скрембльованому, розділеному або змішаному або зашифрованому вигляді.

В іншому варіанті здійснення, званому "мультимаршрутною" передачею даних, в шлюзовому вузлі проводиться розділення пакета, після чого декілька пакетів, що утворилися, перетинають хмара по ряду "паралельних" маршрутів, причому жоден із шляхів не має загальних медіа-вузлів з іншими, за винятком шлюзових вузлів. Потім ці кілька пакетів змішуються для відтворення вихідного пакета, як правило, в вихідному шлюзовому вузлі. Таким чином, навіть якщо хакеру вдалося зрозуміти сенс одного пакета, у нього буде тільки частина всього повідомлення. Пакет також може бути скрембльований та зашифрований в шлюзовому вузлі до або після його розділення, а декілька пакетів, що утворилися, можуть бути повторно скрембльовані або повторно зашифровані в кожному медіа-вузлі, через який вони проходять.

У наступному варіанті здійснення винаходу пакети передаються не тільки по одному шляху або ряду паралельних шляхів в хмарі SDNP, а можуть передаватися різними шляхами, багато з яких перетинаються один з одним. Оскільки в цьому варіанті здійснення винаходу зображення можливих шляхів нагадує решітку, цей процес називається "решітчастою передачею". Як і в описаних вище варіантах здійснення винаходу, пакети можуть бути скрембльовані, зашифровані і розділені або змішані при проходженні через окремі медіа-вузли в хмарі SDNP.

Маршрути пакетів через мережу SDNP визначаються сигнальною функцією, яка може виконуватися або сегментами самих медіа-вузлів, або переважно в "двоканальному" або "триканальному" варіантах окремими сигнальними функціями, які працюють на виділених сигнальних серверах. Сигнальна функція визначає маршрут кожного пакета, коли він залишає клієнтський пристрій (наприклад, стільниковий телефон), на основі стану (наприклад, затримки поширення) мережі, пріоритету та терміновості виклику, і інформує кожен з медіа-вузлів на маршруті про те, що він повинен отримати пакет, та вказує вузол, якому його потрібно відправити. Кожен пакет ідентифікується міткою, а сигнальна функція вказує кожному медіа-вузлу, яку мітку застосувати для кожного з пакетів, які він відправляє. В одному варіанті здійснення винаходу мітка даних включається в заголовок або підзаголовок SDNP, поле даних, прикріплене до кожного субпакета даних, використовується для ідентифікації субпакета. Кожен субпакет може містити сегменти даних з одного або декількох джерел, що зберігаються в спеціальних "слотах" даних в пакеті. Кілька субпакетів можуть бути присутніми в одному більшому пакеті даних при передачі даних між будь-якими двома медіа-вузлами.

Функція маршрутизації узгоджується з функціями розділення та змішування, так як після розділення пакета необхідно визначити відповідні маршрути кожного з субпакетів, на які він розділений, а вузлу, в якому ці субпакети повинні бути відновлені (змішані), має бути дано вказівку провести їх змішування. Пакет може бути розділений, а потім змішаний один раз, як в мультимаршрутному варіанті, або він може бути розділений та змішаний кілька разів у міру проходження по мережі SDNP до вихідного шлюзового вузла. Визначення вузла, в якому буде проводитися розділення пакета, на скільки субпакетів він буде розділений, відповідні маршрути субпакетів та вузли, в яких ці субпакети будуть змішані, щоб відтворити початковий пакет - всі ці питання знаходяться під контролем сигнальної функції, незалежно від того, виконується вона окремими сигнальними серверами чи ні. Алгоритм розділення може визначати, які сегменти даних при передачі повинні бути включені в кожен з субпакетів, а також порядок та положення сегментів даних в цих субпакетах. Алгоритм змішування визначає зворотний процес в вузлі, де субпакети змішуються, щоб відтворити початковий пакет. Зрозуміло, що при наявності відповідної команди від сигнальної функції, в цьому вузлі пакет може бути знову розділений відповідно до іншого алгоритму розділення, який відповідав би часові або стану у момент виникнення розділеного процесу.

Коли медіа-вузол отримує команду від сигнальної функції відправити кілька пакетів в конкретний цільовий медіа-вузол при "наступному переході" по мережі, навіть якщо ці пакети є розділеними пакетами (субпакетами) або відносяться до різних повідомлень, цей медіа-вузол може об'єднувати пакети в один більший пакет, особливо коли кілька субпакетів повинні бути відправлені в один та той же медіа-вузол для наступного переходу (аналогічно поштовому відділенню, в якому поміщають пачку листів з однією і тією же адресою одержувача в ящик та відправляють цей ящик за адресою).

В "динамічних" варіантах здійснення винаходу окремі медіа-вузли в хмарі SDNP не користуються одними та тими ж методами і алгоритмами скремблювання, шифрування або розділення для пакетів, що по проходять по них по черзі. Наприклад, даний медіа-вузол може скремблювати, шифрувати або розділяти будь-який пакет з використанням одного конкретного алгоритму скремблювання, шифрування або розділення, а потім скремблювати, шифрувати або розділяти наступний пакет з використанням іншого алгоритму скремблювання, шифрування або

розділення. Робота в "динамічному" режимі значно збільшує труднощі, з якими стикаються потенційні хакери, тому що у них залишається короткий проміжок часу (наприклад, 100 мсек), щоб зрозуміти смисловий зміст пакета, і навіть якщо їм це вдасться, користь від цих знань буде короткочасною.

5 У динамічних варіантах здійснення винаходу кожен медіа-вузол пов'язаний з так званим "сервером DMZ (англ. DeMilitarized Zone - демілітаризована зона)", який можна розглядати як частину вузла, яка ізольована від частини передачі даних, та в якій є база даних, яка містить списки або таблиці ("селектори") можливих алгоритмів скремблювання, шифрування та розділення, які цей медіа-вузол може застосовувати до вихідних пакетів. Селектор є частиною
10 обсягу інформації, яка називається "спільними секретами", оскільки ця інформація невідома навіть медіа-вузлам і оскільки всі сервери DMZ мають однакові селектори в даний момент часу.

Коли медіа-вузол приймає пакет, який був скремблований, в динамічних варіантах здійснення винаходу, він також отримує "початковий стан", який використовується для вказівки приймаючому вузлу, який алгоритм повинен використовуватися при дескремблюванні пакета.
15 Початковий стан являє собою приховане числове значення, яке саме по собі не має сенсу, але засноване на стані, який постійно змінюється, наприклад, на моменті часу, в який пакет був скремблований попереднім медіа-вузлом. Коли попередній вузол виконав скремблювання пакета, пов'язаний з ним DMZ-сервер сформував початковий стан на основі стану. Відповідно, цей стан також використовувався відповідним сервером DMZ при виборі алгоритму, який
20 повинен застосовуватися при скремблюванні пакета, який був переданий на відправляючий медіа-вузол у вигляді вказівки про те, як скремблювати пакет. Таким чином, вузол, що відправив, отримав вказівки про скремблований пакет та початковий стан, які повинні бути передані на наступний медіа-вузол. Генератор початкових станів, що працює на сервері DMZ, генерує початковий стан відповідно до алгоритму, заснованому на стані під час виконання процесу. Незважаючи на те, що генератор початкових станів і його алгоритми є частиною
25 спільних секретів медіа-вузла, згенерований початковий стан не є секретним, оскільки без доступу до алгоритмів числове початкове значення не має сенсу.

Таким чином, наступний медіа-вузол на маршруті пакету отримує скремблований пакет та початковий стан, заснований на стані, який пов'язаний з пакетом (наприклад, моменту часу, в
30 який було виконано скремблювання). Початковий стан може бути включено в сам пакет або може бути відправлено на приймаючий вузол до відправки пакета, або за тим же маршрутом, що і пакет, або по якомусь іншому маршруту, наприклад, через сигнальний сервер.

Незалежно від того, як він отримує початковий стан, приймаючий вузол відправляє цей початковий стан на свій сервер DMZ. Оскільки на цьому сервері DMZ є селектор або таблиця
35 алгоритмів скремблювання, які є частиною спільних секретів та тому є такими ж, як і селектор на сервері DMZ відправляючого вузла, він може використати початковий стан для ідентифікації алгоритму, який використовувався при скремблюванні пакета і може вказати приймаючому вузлу, як дескремблювати пакет. Таким чином, приймаючий вузол відтворює пакет в дескремблованому вигляді, тим самим відновлюючи вихідні дані. Як правило, до передачі на
40 наступний вузол пакет знову скремблюється відповідно до іншого алгоритму скремблювання. В цьому випадку приймаючий вузол підключається до свого сервера DMZ, щоб отримати алгоритм скремблювання та початковий стан, і процес повторюється.

Таким чином, коли пакет проходить свій шлях по мережі SDNP, він скремблюється кожним вузлом відповідно до різних алгоритмів скремблювання, і на кожному вузлі створюється новий
45 початковий стан, який дозволяє наступному вузлу дескремблювати пакет.

В альтернативному варіанті здійснення даного винаходу фактичний стан (наприклад, час) може передаватися між вузлами (тобто, відправляючому вузлу не потрібно посилати початковий стан приймаючому вузлу). Сервери DMZ, зв'язані як з відправляючим, так і з приймаючим медіа-вузлами, містять генератори прихованих чисел (знову ж таки, що є частиною
50 спільних секретів), які містять однакові алгоритми в будь-який момент часу. Сервер DMZ, зв'язаний з відправляючим вузлом, використовує стан для генерації прихованого числа, а приховане число - для визначення алгоритму скремблювання з селектора або таблиці можливих алгоритмів скремблювання. Відправляючий вузол передає стан приймаючому вузлу. На відміну від початкових станів, приховані номери ніколи не передаються по мережі, а
55 передаються виключно по конфіденційній лінії зв'язку між медіа-вузлом і сервером DMZ. Коли приймаючий медіа-вузол приймає стан для вхідного пакета даних, генератор прихованих чисел у зв'язаному з ним сервері DMZ, використовує стан для створення ідентичного прихованого номера, який згодом використовується селектором або таблицею для ідентифікації алгоритму, відповідно до якого буде дескремблюватися пакет. Цей стан може бути включено в пакет або

може бути передано з відправляючого вузла в приймаючий вузол до передачі пакета або будь-яким іншим маршрутом.

Методи, які використовуються для динамічного шифрування та розділення, аналогічні методам, використовуваним при динамічному скремблюванні, але замість "початкового стану" для динамічного шифрування використовуються "ключі". Спільні секрети, що зберігаються на серверах DMZ, включають в себе селектори або таблиці алгоритмів шифрування та розділення, а також генератори ключів. У разі шифрування з симетричним ключем, відправляючий вузол передає ключ приймаючому медіа-вузлу, який може використовуватися сервером DMZ приймаючого вузла для ідентифікації алгоритму, який використовується при шифруванні пакета, та тим самим дешифрувати файл. У разі шифрування з асиметричним ключем медіа-вузол запитує інформацію, тобто приймаючий вузол спочатку відправляє ключ шифрування вузлу, який містить пакет даних для передачі. Потім відправляючий медіа-вузол зашифровує дані відповідним ключем шифрування. Тільки приймаючий медіа-вузол, який генерує ключ шифрування, має відповідний ключ дешифрування і можливість дешифрувати зашифрований текст, створений з використанням цього ключа шифрування. Важливо відзначити, що при асиметричному шифруванні доступ до ключа шифрування, що використовується для шифрування, не надає ніякої інформації про те, як розшифрувати пакет даних.

При операції розділення медіа-вузол, в якому пакет був розділений, передає початковий стан медіа-вузлу, в якому результуючі субпакети будуть змішуватися, а сервер DMZ, зв'язаний з вузлом змішування, використовує цей початковий стан для ідентифікації алгоритму розділення і, отже, алгоритму, який буде використовуватися при змішуванні субпакетів.

Як зазначено вище, в дво- або триканальних варіантах здійснення винаходу сигнальна функція виконується сигнальним вузлом, який працює в окремій групі серверів, які називаються сигнальними серверами. У таких варіантах здійснення винаходу початкові стани і ключі можуть передаватися через сигнальні сервери, а не з відправляючого медіа-вузла безпосередньо до приймаючого медіа-вузлу. Таким чином, відправляючий медіа-вузол може відправляти початковий стан або ключ на сигнальний сервер, а сигнальний сервер може перенаправляти початковий стан або ключ на приймаючий медіа-вузол. Як було зазначено вище, сигнальні сервери відповідають за розробку маршрутів пакета, тому сигнальний сервер знає наступний медіа-вузол, якому направляється кожен пакет.

Щоб ускладнити завдання для потенційних хакерів, список або таблиця можливих методів скремблювання, розділення або шифрування в селекторі може періодично (наприклад, щогодини або щодня) "перетасовуватися" таким чином, щоб методи, що відповідають конкретним початковим станам або ключам, були змінені. Таким чином, алгоритм шифрування, застосовуваний даними медіа-вузлом до пакету, створеного в момент часу t_1 в день 1, може відрізнитися від алгоритму шифрування, який він застосовує до пакету, що був створений в той же час t_1 в день 2.

Кожен з серверів DMZ зазвичай фізично пов'язаний з одним або декількома медіа-вузлами в одній і тій же "групі серверів". Як було зазначено вище, медіа-вузол може запитувати вказівки про те, що робити з отриманим пакетом, надаючи пов'язаному з ним серверу DMZ початковий стан або ключ (наприклад, на основі часу або стану при створенні пакета), але цей медіа-вузол не може отримати доступ до спільних секретів або будь-яким іншим даним або коду на сервері DMZ. Сервер DMZ відповідає на такі запити, визначаючи початковий стан або ключ і метод, який повинен використовувати медіа-вузол для дескремблювання, дешифрування або змішування пакета. Наприклад, якщо пакет скремблований, і медіа-вузол хоче знати, як його дескремблювати, сервер DMZ може перевірити список (або селектор) алгоритмів скремблювання, щоб знайти конкретний алгоритм, який відповідає цьому початкового стану. Потім сервер DMZ дає команду медіа-вузлу дескремблювати пакет відповідно до цього алгоритму. Іншими словами, медіа-вузол передає запити, що містяться в початкових станах або ключах, серверу DMZ, а сервер DMZ відповідає на ці запити видачею команд.

Незважаючи на те, що медіа-вузли доступні через Інтернет (хоча у них немає IP-адрес, які розпізнаються DNS), сервери DMZ повністю ізольовані від Інтернету, та з'єднуються тільки в локальній мережі дротами або волоконно-оптичним кабелем з підключеними до цієї мережі медіа-серверами.

В "одноканальних" варіантах здійснення винаходу початкові стани і ключі передаються між відправляючим та приймаючим медіа-вузлами в складі пакету даних або можуть передаватися в окремому пакеті до передачі пакета даних по тому ж маршруту, що і пакет даних. Наприклад, при шифруванні пакета медіа-вузол №1 може включити в пакет ключ шифрування на основі моменту часу, в який було виконано шифрування. Коли цей пакет надходить на медіа-вузол №2, медіа-вузол №2 передає ключ на відповідний сервер DMZ, а сервер DMZ може за

допомогою цього ключа вибрати метод дешифрування в своєму селекторі та виконати дешифрування. Потім медіа-вузол №2 може направити запит свого облікового запису DMZ, і знову зашифрувати пакет, перш ніж передати його в медіа-вузол №3. Сервер DMZ знову звертається до селектору, повідомляє медіа-вузлу №2, який метод він повинен використовувати при шифруванні пакета, та передає медіа-вузлу №2 ключ, який відображає стан, що відповідає методу шифрування. Медіа-вузол №2 виконує шифрування та передає зашифрований пакет і ключ (окремо або в складі пакету) медіа-вузлу №3. Потім цей ключ може бути аналогічним чином використаний медіа-вузлом №3 для дешифрування пакета і т.д. В результаті немає одного статичного методу дешифрування, який хакер міг би використовувати при дешифруванні пакетів.

Використання часу або умови динамічного "стану" в наведеному вище прикладі для визначення методу скремблювання, шифрування або розділення, який необхідно включити в початковий стан або ключ, є тільки ілюстративними. Будь-який змінний параметр, наприклад, кількість вузлів, через які пройшов пакет, також може бути використаний як "початковий стан" або ключ, який буде використаний для вибору конкретного методу скремблювання, шифрування або розділення. В "двоканальних" варіантах початкові стани і ключі можуть передаватися між медіа-вузлами через другий канал "команд і управління", який складається із сигнальних серверів, а не передаватися безпосередньо між медіа-вузлами. Сигнальні вузли можуть також надавати медіа-вузлам інформацію маршрутизації та повідомляти медіа-вузлам на маршруті пакету, як саме цей пакет повинен бути розділений або змішаний з іншими пакетами, при цьому вони вимагають, щоб кожен медіа-вузол застосовував ідентифікаційну "мітку" для кожного переданого пакета, щоб наступний медіа-вузол (вузли) був здатний розпізнавати пакет (и). Сигнальні сервера, по можливості, надають даному медіа-вузлу інформацію тільки про попередній і наступний медіа-вузол для пакета, що проходить через мережу. Жоден окремий медіа-вузол не знає весь маршрут пакета через хмару SDNP. У деяких варіантах здійснення винаходу функція маршрутизації може бути розділена між двома або більше сигнальними серверами, при цьому один сигнальний сервер визначає маршрут до конкретного медіа-вузлу, другий сигнальний сервер визначає маршрут від цього медіа-вузла до іншого медіа-вузлу і так далі до вихідного шлюзового вузла. При цьому жоден сигнальний сервер теж не має повної інформації про маршрут пакета даних.

В "триканальних" варіантах третя група серверів, звана "серверами імен", використовується для ідентифікації елементів в хмарі SDNP і для зберігання інформації, що стосується ідентифікації пристроїв, підключених до хмари SDNP, і їх відповідних IP-адрес або SDNP-адрес. Крім того, сервери імен постійно контролюють медіа-вузли в хмарі SDNP, підтримуючи, наприклад, поточний список активних медіа-вузлів та таблицю затримок поширення для кожної комбінації медіа-вузлів у хмарі. На першому етапі розміщення виклику клієнтський пристрій, наприклад, планшет, може відправити IP-пакет на сервер імен із запитом адреси та іншу інформацію одержувача або абонента. Крім того, окремий виділений сервер імен використовується для роботи при першому контакті щоразу, коли пристрій вперше підключається, тобто реєструється, в хмарі.

В якості додаткового переваги, з точки зору безпеки, в одиночній хмарі SDNP можуть бути передбачені окремі "зони" безпеки, що мають різні селектори, початкові стани і генератори ключів та інші спільні секрети. Суміжні зони з'єднуються мостовими медіа-вузлами, які містять спільні секрети обох зон і мають можливість перетворювати дані, відформатовані відповідно до правил для однієї зони, в дані, відформатовані відповідно до правил для іншої зони, і навпаки.

Аналогічним чином, для зв'язку між різними хмарами SDNP, що обслуговуються, наприклад, різними провайдером послуг, між інтерфейсами мостових серверів в кожній хмарі формується двобічний канал зв'язку. Кожен інтерфейс мостового серверу має доступ до відповідних спільних секретів і інших елементів безпеки для кожної хмари.

Подібні методи безпеки зазвичай можуть застосовуватися на "останній милі" між хмарою SDNP і клієнтським пристроєм, наприклад, стільниковим телефоном або планшетом. Клієнтський пристрій зазвичай поміщається в окрему зону безпеки від хмари та спочатку має стати авторизованим клієнтом SDNP, що передбачає установку на клієнтському пристрої спеціального програмного пакета для зони безпеки пристрою, як правило, шляхом завантаження з сервера адміністрування SDNP. Клієнтський пристрій пов'язаним з хмарою SDNP через шлюзовий медіа-вузол в хмарі. У шлюзового медіа-вузла є доступ до спільних секретів, що належать як до хмарної зони безпеки, так і до зони безпеки клієнтського пристрою, але клієнтський пристрій не має доступу до спільних секретів, що належать до хмари SDNP.

Для підвищення рівня безпеки клієнтські пристрої можуть обмінюватися початковими станами і ключами безпосередньо один з одним через сигнальні сервери. Таким чином,

клієнтський пристрій, що передає, може відправляти початковий стан та/або ключ напряду до приймаючого клієнтському пристрою. У таких варіантах здійснення винаходу пакет, прийнятий приймаючим клієнтським пристроєм, буде знаходитися в тій же скрембльованій або зашифрованій формі, що і пакет, відправлений з клієнтського пристрою. Таким чином, 5 приймаючий клієнтський пристрій може використовувати початковий стан або ключ, який він отримує від відправляючого клієнтського пристрою, щоб дескремблювати або дешифрувати пакет. Обмін початковими станами і ключами безпосередньо між клієнтськими пристроями доповнює власне динамічне скремблювання і шифрування мережі SDNP, і, таким чином, являє собою додатковий рівень безпеки, званий вкладеною безпекою.

Крім того, клієнтський пристрій або шлюзовий вузол, з яким він обмінюється даними, можуть змішувати пакети, які представляють собою дані одного та того ж виду - наприклад, голосові пакети, файли текстових повідомлень, документи, частини програмного забезпечення, або які представляють собою інформацію різного типу, наприклад, один голосовий пакет і один текстовий файл, один текстовий пакет і одне відео або фотозображення - до того, як пакети 15 досягнуть мережі SDNP, вихідний шлюзовий вузол або клієнтський пристрій одержувача можуть розділити змішаний пакет для відновлення вихідних пакетів. Це може бути зроблено із застосуванням скремблювання, шифрування або розділення, які відбуваються в мережі SDNP. У таких випадках відправляючий клієнтський пристрій може відправити приймаючому клієнтському пристрою початковий стан, в якому вказується, як розбити пакет, щоб відтворити вихідні пакети, які були змішані в відправляючому клієнтському пристрої або шлюзовому медіа-вузлі. Виконання послідовного змішування і розділення може включати в себе лінійну послідовність операцій або, як варіант, використовувати вкладену архітектуру, де клієнти приймають заходи безпеки хмари SDNP та свої власні.

Важливою перевагою даного винаходу є те, що в мережі SDNP немає єдиного пункту управління і що жоден вузол і жоден сервер в цій мережі не має повного уявлення про те, як відбувається передача даних або як вона може динамічно змінюватися.

Наприклад, сигнальні вузли, що працюють на сигнальних серверах, знають маршрут (або, в деяких випадках, тільки частину маршруту), за яким відбувається передача даних, але вони не мають доступу до вмісту переданих даних і не знають, хто є істинним абонентом або клієнтом. 30 Крім того, сигнальні вузли не мають доступу до спільних секретів на серверах DMZ медіа-вузла, тому вони не знають, як шифруються, скремблюються, розділяються або змішуються пакети даних під час передачі.

Сервери імен SDNP знають справжні номери телефонів або IP-адреси абонента, але не мають доступу до переданих даних або маршрутизації різних пакетів та субпакетів. Як і вузли сигналізації, сервери імен не мають доступу до спільних секретів на серверах DMZ медіа-вузла, 35 тому вони не знають, як шифруються, скремблюються, розділяються або змішуються пакети даних під час передачі.

Медіа-вузли SDNP, фактично передають медіа-контент, не маючи уявлення про те, хто є відправником інформації, і не знають маршрут, по якому проходять різні фрагментовані субпакети через хмару SDNP. Насправді, кожен медіа-вузол знає тільки, які пакети даних очікувати (ідентифікуються своїми мітками або заголовками) і куди їх відправляти далі, тобто "наступний перехід", але медіа-вузли не знають, як дані шифруються, скремблюються, змішуються або розділяються, а також не знають, як вибрати алгоритм або дешифрувати файл, використовуючи стан, числове значення початкового стану або ключ. Знання, необхідні для 45 правильної обробки сегментів даних вхідних пакетів, відомі тільки серверу DMZ, який використовує свої спільні секрети, алгоритми, недоступні через мережу або самому медіа-вузлу.

Іншим оригінальним аспектом цього винаходу є його здатність зменшувати мережеву затримку і мінімізувати затримку поширення для забезпечення QoS, а також виключати ехо-сигнали або відкидати виклики шляхом управління розміром пакетів даних, тобто одночасно відправляючи невеликі пакети даних через хмару, а не покладаючись на одне високошвидкісне з'єднання. Динамічна маршрутизація мережі SDNP використовує свої знання про затримки поширення між вузлами мережі для динамічного вибору найкращого маршруту для передачі даних в цей момент часу. В іншому варіанті здійснення винаходу для високопріоритетних клієнтів мережа може полегшувати маршрутизацію передачі, відправляючи дубльовані повідомлення у фрагментованому вигляді через хмару SDNP, вибираючи тільки найшвидші дані для відновлення вихідного звуку або вмісту даних.

Серед багатьох переваг системи SDNP відповідно до цього винаходу в варіантах з паралельною і "решітчастою передачею" пакети можуть бути фрагментовані при їх переході в 60 хмару SDNP, таким чином попередивши можливість потенційним хакерам зрозуміти вміст

повідомлення навіть у тому випадку, якщо вони здатні розшифрувати окремих субпакет або групу субпакетів; також в "динамічних" варіантах методи скремблювання, шифрування і розділення, що застосовуються до пакетів, постійно змінюються, що не дає потенційному хакеру будь-якої переваги від успішного дешифрування пакета в даний момент часу. Численні додаткові переваги варіантів здійснення даного винаходу стануть очевидні фахівцям в даній області після ознайомлення з наведеними нижче описом.

Короткий опис зображень

На наведених нижче зображеннях компонентам, які мають загальну схожість, присвоєні подібні числові позначення. Однак слід зазначити, що не кожен компонент, якому присвоєно певне числове позначення, обов'язково ідентичний іншому компоненту з тим же числовим позначенням. Наприклад, операція шифрування, що має конкретне числове позначення, необов'язково ідентична іншій операції шифрування з тим же числовим позначенням. Більш того, групи компонентів, наприклад, сервери в мережі, які мають одне спільне ідентифікаційне числове позначення, необов'язково ідентичні один одному.

На рисунку 1 схематично показана телефонна мережа з комутацією каналів.

На рисунку 2 схематично показана мережа зв'язку з комутацією пакетів.

На рисунку 3 схематично показана маршрутизація пакетів в мережі зв'язку з комутацією пакетів.

На рисунку 4 в графічному вигляді показана конструкція IP-пакета для зв'язку по мережі з комутацією пакетів.

На рисунку 5A схематично показана мережа зв'язку, що ілюструє приклади фізичного рівня 1 з'єднання з високою пропускною здатністю.

На рисунку 5B схематично показана мережа зв'язку, що ілюструє приклади фізичного рівня 1 з'єднання останньої милі.

На рисунку 6A схематично показано з'єднання між двома пристроями на фізичному рівні 1.

На рисунку 6B схематично показано з'єднання між трьома пристроями на загальному фізичному рівні 1.

На рисунку 7A схематично показано з'єднання каналного рівня 2 між трьома пристроями з використанням шинної архітектури.

На рисунку 7B схематично показано з'єднання каналного рівня 2 між трьома пристроями з використанням архітектури з концентратором.

На рисунку 7C схематично показано з'єднання каналного рівня 2 між трьома пристроями з використанням архітектури послідовного ланцюга (daisy chain).

На рисунку 8A схематично показано з'єднання каналного рівня 2 між трьома пристроями з мережевим комутатором.

На рисунку 8B схематично в простому вигляді показаний мережевий комутатор.

На рисунку 8C схематично показана робота мережевого комутатора.

На рисунку 9 в графічному вигляді показана конструкція каналного рівня 2 IP-пакета з використанням протоколу Ethernet.

На рисунку 10 схематично в простому вигляді показаний мережевий міст Ethernet-радіо.

На рисунку 11 в графічному вигляді показана конструкція каналного рівня 2 IP-пакета з використанням протоколу Wi-Fi.

На рисунку 12A схематично показана двонаправлена робота точки доступу до мережі Wi-Fi.

На рисунку 12B схематично показана двонаправлена робота перетворювача Wi-Fi.

На рисунку 13 в графічному вигляді показано еволюцію телефонного зв'язку, передачі тексту і даних по мережах стільникового зв'язку.

На рисунку 14A в графічному вигляді показано розділення по частоті в мережах зв'язку 4G/LTE.

На рисунку 14B в графічному вигляді показано кодування при мультиплексуванні з ортогональним частотним розділенням каналів (OFDM), використовуваному в радіозв'язку 4G/LTE.

На рисунку 15 в графічному вигляді показана конструкція каналного рівня 2 IP-пакета з використанням протоколу 4G/LTE.

На рисунку 16 схематично показана мережа зв'язку кабельного модему.

На рисунку 17 схематично показана конструкція каналного рівня 2 мережі зв'язку кабельного модему.

На рисунку 18 в графічному вигляді показана конструкція решітчастого кодування, що використовується в кабельних модемах на основі DOCSIS.

На рисунку 19 в графічному вигляді показана конструкція каналного рівня 2 пакету зв'язку з використанням протоколу DOCSIS.

На рисунку 20 схематично показано мережеве з'єднання рівня 3 між трьома пристроями.

На рисунку 21 в графічному вигляді показані пакети зв'язку, сформовані відповідно до семирівневої моделі OSI.

5 На рисунку 22 в графічному вигляді показана конструкція мережевого рівня 3 для порівняння пакетів зв'язку IPv4 і IPv6.

На рисунку 23 в графічному вигляді показаний IP-пакет відповідно до протоколу IPv4.

На рисунку 24 в графічному вигляді показаний IP-пакет відповідно до протоколу IPv6.

На рисунку 25 в графічному вигляді показані поля адреси, побудовані відповідно до протоколів IPv4 та IPv6.

10 На рисунку 26 в графічному вигляді показано поле протоколу/наступного заголовка в IP-пакеті і його відповідне корисне навантаження.

На рисунку 27 схематично показаний транспортний рівень 4 з'єднання між трьома пристроями.

15 На рисунку 28A в графічному вигляді показана конструкція транспортного рівня 4 IP-пакета з використанням протоколу TCP.

На рисунку 28B наведена таблиця, що описує поля протоколу TCP.

На рисунку 29 в графічному вигляді показана послідовності передачі TCP-пакета.

На рисунку 30 в графічному вигляді показана конструкція транспортного рівня 4 IP-пакета з використанням протоколу UDP.

20 На рисунку 31A схематично показана передача даних на транспортному рівні 4 від клієнта до хоста.

На рисунку 31B схематично показана передача даних на транспортному рівні 4 від хоста до клієнта.

На рисунку 31C наведена таблиця, що описує звичайне розділення портів UDP і TCP.

25 На рисунку 31D наведена таблиця, що описує виділені блоки для зарезервованих адрес і адрес ад-хост-портів, що використовуються UDP і TCP.

На рисунку 32A схематично показаний перетворювач мережевих адрес (NAT).

На рисунку 32B схематично показана робота перетворювача мережевих адрес.

30 На рисунку 33 схематично показані три пристрої, підключені на прикладному рівні 5, рівні 6 і рівні 7.

На рисунку 34 схематично показана отримання вмісту з використанням додатка рівня 7 для протоколу передачі файлів (FTP).

На рисунку 35A схематично показане завантаження веб-сторінок за допомогою програми рівня 7 для протоколу передачі гіпертексту (HTTP).

35 На рисунку 35B в графічному вигляді показана веб-сторінка HTML, створена шляхом завантаження з різних серверів.

На рисунку 36 схематично показана додатки рівня 7 для електронної пошти на основі протоколу IMAP.

На рисунку 37 показана таблиця, в якій порівнюється QoS для різних станів мережі.

40 На рисунку 38 показаний графік залежності часу прийому-передачі (RTT) від затримки поширення всередині вузла мережі.

На рисунку 39 схематично показані різні приклади шкідливого ПО в мережі зв'язку.

На рисунку 40 в простому вигляді представлено мережеве з'єднання хмари і останньої милі, а також шкідливі програми, які використовуються при кібератаках.

45 На рисунку 41A показані електронні пристрої, здатні контролювати зв'язок по Ethernet і Wi-Fi.

На рисунку 41B показані електронні пристрої, здатні контролювати стільниковий телефонний зв'язок.

На рисунку 41C показано електронний пристрій, здатний відстежувати зв'язок по волоконно-оптичному кабелю.

50 На рисунку 42 показана таблиця, в якій порівнюються десять наявних у продажу шпигунських програм.

На рисунку 43 наведена карта світу, на якій показані інциденти, пов'язані з проведенням кібератак всього за один день.

55 На рисунку 44 показана можливість аналізу IP-пакетів та проведення атак "людина посередині" в мережі з комутацією пакетів.

На рисунку 45 показана кібератака з використанням виявлення на основі опитування портів.

На рисунку 46 показана кібератака, яка використовує захоплення IP-пакетів.

На рисунку 47 схематично показано шифрування з двома ключами.

На рисунку 48A схематично показана віртуальна приватна мережа.

60 На рисунку 48B показаний стек зв'язку віртуальної приватної мережі.

- На рисунку 48С схематично показаний VoIP-виклик за спеціальною VPN.
 На рисунку 49А схематично показаний OTT VoIP-виклик через Інтернет.
 На рисунку 49В схематично показаний VoIP-виклик по тимчасовій мережі.
 На рисунку 50 схематично показана традиційна передача пакетів по мережі.
 5 На рисунку 51А схематично показаний процес скремблювання пакета.
 На рисунку 51В схематично показаний процес дескремблювання пакетів.
 На рисунку 51С схематично показані різні алгоритми скремблювання пакетів.
 На рисунку 51D схематично показано статичне параметричне скремблювання пакетів.
 На рисунку 51Е схематично показано динамічне скремблювання з прихованим числом.
 10 На рисунку 51F схематично показано динамічне скремблювання пакетів з підмішуванням псевдовипадкового сигналу.
 На рисунку 52 схематично показано статичне скремблювання пакетів в лінійній мережі.
 На рисунку 53 схематично показаний процес повторного скремблювання пакетів.
 На рисунку 54 схематично показано динамічне скремблювання пакетів в лінійній мережі.
 15 На рисунку 55А схематично показаний процес шифрування пакетів.
 На рисунку 55В схематично показаний процес дешифрування пакетів.
 На рисунку 56 схематично показаний процес скремблювання з шифруванням і його зворотна функція.
 На рисунку 57 схематично показано статичне скремблювання з шифруванням в лінійній мережі.
 20 На рисунку 58 схематично показаний процес повторного пакетування даних DUSE, що включає повторне скремблювання та повторне шифрування.
 На рисунку 59 схематично показано динамічне скремблювання з шифруванням в лінійній мережі.
 25 На рисунку 60А схематично показаний процес розділення пакетів фіксованої довжини.
 На рисунку 60В схематично показаний процес змішування пакетів фіксованої довжини.
 На рисунку 61А схематично показані різні способи змішування пакетів.
 На рисунку 61В схематично показано змішування пакетів з конкатенацією.
 На рисунку 61С схематично показано змішування пакетів з чергуванням.
 30 На рисунку 62А схематично показаний спосіб змішування з подальшим скремблюванням.
 На рисунку 62В схематично показаний спосіб скремблювання з подальшим змішуванням.
 На рисунку 63 схематично показано статичне скремблювання з подальшим змішуванням в лінійній мережі.
 На рисунку 64 схематично показано динамічне скремблювання з подальшим змішуванням в лінійній мережі.
 35 На рисунку 65 схематично показані різні процеси обробки зашифрованих пакетів.
 На рисунку 66А схематично показано динамічне змішування зашифрованих скремблених даних в лінійній мережі.
 На рисунку 66В схематично показано статичне змішування скремблених даних з динамічним шифруванням в лінійній мережі.
 40 На рисунку 66С схематично показано динамічне змішування, скремблювання і шифрування в лінійній мережі з використанням методу "повернення до нормального стану".
 На рисунку 66D схематично показаний метод DUS-MSE повернення до нормального стану.
 На рисунку 67А схематично показано змішування пакетів з одним виходом.
 45 На рисунку 67В схематично показано змішування пакетів з декількома виходами.
 На рисунку 67С схематично показано розділення пакетів змінної довжини.
 На рисунку 67D схематично показано розділення пакетів фіксованої довжини.
 На рисунку 67Е показана блок-схема, що ілюструє алгоритм змішування.
 На рисунку 67F показана блок-схема, що ілюструє алгоритм розділення.
 50 На рисунку 67G схематично показана блок-схема, що ілюструє двоступеневий алгоритм змішування та скремблювання.
 На рисунку 67H показана блок-схема, що ілюструє гібридний алгоритм змішування/скремблювання.
 На рисунку 67I показана блок-схема, що ілюструє ідентифікацію міток.
 55 На рисунку 68А схематично показана різні типи маршрутизації пакетів.
 На рисунку 68В схематично показана одномаршрутні або лінійна передача.
 На рисунку 68С схематично показана мультимаршрутна або паралельна передача.
 На рисунку 68D схематично показана передача по решітчастому маршруту.
 На рисунку 68Е схематично показаний альтернативний варіант здійснення передачі по
 60 решітчастому маршруту.

На рисунку 69 схематично показана статична мультимаршрутна передача.

На рисунку 70 схематично показано статичне мультимаршрутне скремблювання.

На рисунку 71А схематично показано динамічне мультимаршрутне скремблювання.

На рисунку 71В схематично показані різні комбінації скремблювання та розділення.

5 На рисунку 71С схематично показано вкладене змішування, розділення, скремблювання і шифрування.

На рисунку 72 схематично показаний метод статичного скремблювання з подальшим розділенням і динамічним шифруванням.

10 На рисунку 73 схематично показана мультимаршрутна передача статичних скремблених даних з динамічним шифруванням.

На рисунку 74 схематично показані різні комбінації методів розділення, скремблювання і шифрування.

На рисунку 75 схематично показана статична решітчаста маршрутизація даних змінної довжини.

15 На рисунку 76 схематично показана статична скремблена решітчаста маршрутизація даних змінної довжини.

На рисунку 77А схематично показано змішування та розділення для решітчастої передачі даних змінної довжини.

20 На рисунку 77В схематично показано змішування та розділення для решітчастої передачі даних фіксованої довжини.

На рисунку 77С схематично показані різні комбінації можливостей підключення вузла зв'язку до решітчастої мережі.

На рисунку 77D схематично показані можливості підключення вузла непланарної решітчастої мережі.

25 На рисунку 78А схематично показано змішування та розділення повторно скремблених даних.

На рисунку 78В схематично показано змішування дескремблених даних решітчастих виходів.

30 На рисунку 78С схематично показана операція розділення та скремблювання для решітчастих виходів.

На рисунку 78D схематично показано повторне скремблювання та повторне змішування для решітчастої передачі.

На рисунку 79А схематично показано змішування та розділення скремблених даних фіксованої довжини для решітчастої передачі.

35 На рисунку 79В схематично показаний альтернативний варіант змішування та розділення скремблених даних фіксованої довжини для решітчастої передачі.

На рисунку 80 схематично показана статична скремблена решітчаста маршрутизація даних змінної довжини.

На рисунку 81А схематично показано змішування та розділення зашифрованих даних.

40 На рисунку 81В схематично показано змішування дешифрованих даних решітчастих виходів.

На рисунку 81С схематично показане розділення і шифрування для решітчастих виходів.

На рисунку 82А схематично показаний зашифрований пакет з повторним скремблюванням для решітчастої передачі.

45 На рисунку 82В схематично показана операція дешифрування, дескремблювання та змішування (DUM) для решітчастих виходів.

На рисунку 82С схематично показана операція розділення, скремблювання і шифрування (SSE) для решітчастих виходів.

На рисунку 83А схематично показаний медіа-вузол SDNP для решітчастої передачі.

На рисунку 83В схематично показаний одномаршрутний медіа-вузол SDNP.

50 На рисунку 83С схематично показаний одномаршрутний прохідний медіа-вузол SDNP.

На рисунку 83D схематично показаний медіа-вузол SDNP для надлишкового дублювання маршруту.

На рисунку 83Е схематично показаний медіа-вузол SDNP, що виконує одномаршрутне скремблювання.

55 На рисунку 83F схематично показаний медіа-вузол SDNP, що виконує одномаршрутне дескремблювання.

На рисунку 83G схематично показаний медіа-вузол SDNP, що виконує повторне одномаршрутне скремблювання.

60 На рисунку 83H схематично показаний медіа-вузол SDNP, що виконує одномаршрутне шифрування.

На рисунку 83I схематично показаний медіа-вузол SDNP, що виконує одномаршрутне дешифрування.

На рисунку 83J схематично показаний медіа-вузол SDNP, що виконує повторне одномаршрутне шифрування.

5 На рисунку 83J схематично показаний медіа-вузол SDNP, що виконує одномаршрутне шифрування скрембльованих даних.

На рисунку 83L схематично показаний медіа-вузол SDNP, що виконує одномаршрутне дешифрування дескрембльованих даних.

10 На рисунку 83M схематично показаний медіа-вузол SDNP, що виконує повторне одномаршрутне пакетування.

На рисунку 83N схематично показаний решітчастий вхід шлюзу SDNP.

На рисунку 83O схематично показаний решітчастий вихід шлюзу SDNP.

На рисунку 83P схематично показаний скрембльований вхід шлюзу SDNP і дескрембльований вихід шлюзу SDNP.

15 На рисунку 83Q схематично показаний зашифрований вхід шлюзу SDNP і дешифрований вихід шлюзу SDNP.

На рисунку 83R схематично показаний скрембльований зашифрований вхід шлюзу SDNP і дескрембльований дешифрований вихід шлюзу SDNP.

20 На рисунку 83S схематично показані шлюзи SDNP, що виконують решітчасте повторне скремблювання та решітчасте повторне шифрування.

На рисунку 84A схематично показані з'єднання медіа-вузла SDNP.

На рисунку 84B схематично показана хмара SDNP.

На рисунку 84C схематично показаний шифрований зв'язок між медіа-вузлами SDNP.

На рисунку 84D схематично показаний міжвузловий шифрований зв'язок SDNP.

25 На рисунку 85A схематично показана хмара SDNP з можливістю підключення останньої милі до клієнта стільникового телефону.

На рисунку 85B схематично показаний шлюз SDNP з незахищеним з'єднанням останньої милі.

На рисунку 85C схематично показаний шлюз SDNP з захищеним з'єднанням останньої милі.

30 На рисунку 85D схематично показаний альтернативний варіант здійснення шлюзу SDNP з захищеним з'єднанням останньої милі.

На рисунку 86 схематично показані різні клієнти, підключені до хмари SDNP.

На рисунку 87 схематично показана маршрутизація пакетів в хмарі SDNP.

На рисунку 88A схематично показана маршрутизація пакетів, що починається в хмарі SDNP.

35 На рисунку 88B схематично показана маршрутизація пакетів першого хмарного переходу в хмарі SDNP.

На рисунку 88C схематично показана маршрутизація пакетів другого хмарного переходу в хмарі SDNP.

40 На рисунку 88D схематично показана маршрутизація пакетів третього хмарного переходу в хмарі SDNP.

На рисунку 88E схематично показана маршрутизацію пакетів з шлюзу хмари SDNP.

На рисунку 88F схематично показана зведена інформація про маршрутизації пакетів в хмарі SDNP для конкретного сеансу.

45 На рисунку 89A схематично показана маршрутизація пакетів, що починається в хмарі SDNP, для альтернативного сеансу.

На рисунку 89B схематично показана маршрутизація пакетів першого хмарного переходу в хмарі SDNP для альтернативного сеансу.

На рисунку 89C схематично показана маршрутизація пакетів другого хмарного переходу в хмарі SDNP для альтернативного сеансу.

50 На рисунку 89D схематично показана маршрутизація пакетів третього хмарного переходу в хмарі SDNP для альтернативного сеансу.

На рисунку 89E схематично показана маршрутизація пакетів четвертого хмарного переходу в хмарі SDNP для альтернативного сеансу.

55 На рисунку 89F схематично показана маршрутизацію пакетів з шлюзу хмари SDNP для альтернативного сеансу.

На рисунку 89G схематично показана зведена інформація про маршрутизації пакетів в хмарі SDNP для альтернативного сеансу.

На рисунку 90 схематично показано вміст пакету SDNP, схильного для проведення атак типу "людина посередині" і аналізу пакетів.

60 На рисунку 91A в графічному вигляді схематично показана передача пакетів SDNP в часі.

На рисунку 91B схематично показана передача пакетів SDNP в часі в вигляді таблиці.

На рисунку 91C в графічному вигляді схематично показаний пакет SDNP для альтернативного сеансу передачі пакетів в часі.

5 На рисунку 92A схематично показано управління вхідними пакетами SDNP для медіа-вузла SDNP.

На рисунку 92B схематично показано управління вихідними пакетами SDNP для медіа-вузла SDNP.

На рисунку 93 схематично показаний вибір алгоритму SDNP.

На рисунку 94 схематично показана регулярна перестановка алгоритму SDNP.

10 На рисунку 95A схематично показана багатозонна хмара SDNP.

На рисунку 95B схематично показана багатозонне управління захищеністю SDNP.

На рисунку 95C схематично показаний багатозонний дуплексний міст SDNP.

На рисунку 95D схематично показана багатозонна мережа SDNP, що містить кілька хмар.

На рисунку 95E схематично показаний незахищений зв'язок між хмарами SDNP.

15 На рисунку 95F схематично показана використання багатозонних дуплексних мостів SDNP для захищених каналів зв'язку між хмарами.

На рисунку 96A схематично показаний захищений шлюз SDNP і канал останньої милі до планшета клієнта.

На рисунку 96B схематично показані функції хмарного інтерфейсу.

20 На рисунку 96C схематично показані функції клієнтського інтерфейсу.

На рисунку 96D схематично показані функції клієнта.

На рисунку 97A схематично показані функціональні елементи захищеного шлюзу хмари SDNP.

25 На рисунку 97B схематично показано взаємозв'язок функціональних елементів в захищеному шлюзі хмари SDNP.

На рисунку 98 схематично показаний клієнтський інтерфейс в захищеному шлюзі хмари SDNP.

На рисунку 99A схематично показано управління ключами при багатозонній передачі.

30 На рисунку 99B схематично показано управління ключами при багатозонній передачі з використанням скремблювання при передачі в хмарі SDNP.

На рисунку 99C схематично показано управління ключами при багатозонній передачі з використанням скремблювання при передачі в хмарі SDNP і на одиночному маршруті останньої милі.

35 На рисунку 99D схематично показано управління ключами при багатозонній передачі з наскрізним скремблюванням.

На рисунку 99E схематично показано управління ключами при багатозонній передачі з використанням скремблювання при передачі в хмарі SDNP і на одиночному маршруті останньої милі з повторним скремблюванням.

40 На рисунку 99F схематично показано управління ключами при багатозонній передачі з повторним скремблюванням, специфічним для зони.

На рисунку 100A схематично показана доставка і установка коду SDNP.

На рисунку 100B схематично показана доставка і багатозональна установка коду SDNP.

На рисунку 101A схематично показана доставка секретних даних SDNP на сервер DMZ.

На рисунку 101B схематично показано зв'язок по медіаканалах секретних даних.

45 На рисунку 101C схематично показана передача секретних даних і ключів по медіаканалах SDNP.

На рисунку 102 схематично показано динамічне управління SDNP через сигнальний сервер SDNP.

50 На рисунку 103A схематично показана доставка ключів та початкових станів SDNP через сигнальний сервер SDNP.

На рисунку 103B схематично показаний альтернативний варіант здійснення доставки ключів та початкових станів SDNP через сигнальний сервер SDNP.

На рисунку 104 схематично показана доставка клієнту SDNP.

55 На рисунку 105A схематично показана одноканальна доставка ключів та початкових станів SDNP клієнту.

На рисунку 105B схематично показаний альтернативний варіант здійснення одноканальної доставки ключів та початкових станів SDNP клієнту.

На рисунку 106 схематично показана клієнтська перестановка алгоритму SDNP.

60 На рисунку 105A схематично показана двоканальна доставка ключів та початкових станів SDNP клієнту.

На рисунку 108 схематично показана доставка відкритого ключа клієнта SDNP.

На рисунку 109 схематично показана одноканальна решітчаста передача даних SDNP.

На рисунку 110A показана блок-схема спеціального зв'язку по медіаканалах SDNP, частина

1.

5 На рисунку 110B показана блок-схема спеціального зв'язку по медіаканалах SDNP, частина

2.

На рисунку 110C показана блок-схема спеціального зв'язку по медіаканалах SDNP, частина

3.

На рисунку 110D показана блок-схема спеціального зв'язку по медіаканалах SDNP, частина

10

4.

На рисунку 110E показана блок-схема спеціального зв'язку по медіаканалах SDNP, частина

5.

На рисунку 110F показана блок-схема спеціального зв'язку по медіаканалах SDNP, частина

6.

15

На рисунку 111A приведена блок-схема, яка містить зведені дані про спеціальну послідовності відправки пакета SDNP.

На рисунку 111B показана карта мережі, яка містить зведені дані про маршрутизацію відправки в SDNP.

20

На рисунку 112A приведена блок-схема, яка містить зведені дані про спеціальну послідовність відповіді пакета SDNP.

На рисунку 112B показана карта мережі, яка містить зведені дані про маршрутизацію відповіді в SDNP.

На рисунку 113A схематично показана підготовка пакету SDNP.

25

На рисунку 113B схематично показаний альтернативний варіант здійснення підготовки пакета SDNP.

На рисунку 114 показано таблиця, яка містить зведені дані про один з варіантів реалізації архітектури пакета SDNP.

На рисунку 115 схематично показаний варіант здійснення двоканальної решітчастої передачі даних SDNP, в якому сигнальна функція в середині хмари реалізується тими ж серверами, які діють як медіа-вузли, а сигнальна функція на ділянках першої і останньої милі реалізується окремими сигнальними серверами.

30

На рисунку 116 схематично показаний альтернативний варіант здійснення двоканальної решітчастої передачі даних SDNP, в якому сигнальна функція, як в хмарі, так і на ділянках першої і останньої милі реалізується окремими сигнальними серверами.

35

На рисунку 117 схематично показана триканальна решітчаста передача даних SDNP.

На рисунку 118 схематично показаний вузол SDNP і реєстрація пристроїв.

На рисунку 119 схематично показаний контроль затримки поширення в SDNP в режимі реального часу.

40

На рисунку 120 показано графік, який ілюструє контроль затримки поширення тестового пакета.

На рисунку 121 схематично показана триканальна решітчаста передача даних SDNP.

На рисунку 122 схематично показані резервні сервери імен SDNP.

На рисунку 123 схематично показані резервні сигнальні сервери SDNP.

45

На рисунку 124A показана блок-схема триканального зв'язку SDNP, частина 1.

На рисунку 124B показана блок-схема триканального зв'язку SDNP, частина 2.

На рисунку 124C показана блок-схема триканального зв'язку SDNP, частина 3.

На рисунку 124D схематично показана блок-схема, що показує триканальний зв'язок SDNP, частина 4.

50

На рисунку 124E схематично показана блок-схема, що показує триканальний зв'язок SDNP, частина 5.

На рисунку 125A приведена блок-схема, яка містить зведені дані про триканальний послідовності відправки пакета SDNP.

На рисунку 125B показана карта мережі, яка містить зведені дані про маршрутизації триканальної відправки пакетів SDNP.

55

На рисунку 126A приведена блок-схема, яка містить зведені дані про триканальної відповідної послідовності пакетів SDNP.

На рисунку 126B показана карта мережі, яка містить зведені дані про маршрутизації триканальної відповідної послідовності пакетів SDNP.

60

На рисунку 126C приведена блок-схема, яка містить зведені дані про альтернативний варіант здійснення триканальної відповідної послідовності пакетів SDNP.

На рисунку 127 схематично показана попередня обробка пакету вузла SDNP.

На рисунку 128 схематично показано повторне пакетування SDNP.

На рисунку 129A схематично показано відновлення пакетів останнього вузла в реальному часі.

5 На рисунку 129B схематично показано відновлення пакетів буферизованого останнього вузла.

На рисунку 129C схематично показано відновлення пакетів буферизованого клієнта.

На рисунку 129D показана блок-схема, яка містить зведені дані про конструкції клієнтського пакета.

10 На рисунку 130 схематично показана команда SDNP та пакети сигналів управління.

На рисунку 131 схематично показано визначення динамічного маршруту SDNP.

На рисунку 132A приведена блок-схема, що показує пакети сигналів управління і команд, шлях 1-1.

15 На рисунку 132B приведена блок-схема, що показує пакети сигналів управління і команд, шлях 1-2.

На рисунку 132C схематично показано відновлення пакета SDNP.

На рисунку 133A схематично показано уявлення рівнів OSI фрагментованої передачі SDNP.

На рисунку 133B схематично показано уявлення рівнів моделі OSI для тунельної фрагментованої передачі SDNP.

20 На рисунку 134 схематично показана маршрутизація передачі пакетів SDNP.

На рисунку 135 показана таблиця, що порівнює зв'язок по мережі SDNP зі зв'язком по інших мережах з комутацією пакетів.

Опис винаходу

Після майже півтора століття використання телефонних мереж з комутованими каналами сучасні системи і мережі зв'язку за період часу тривалістю всього лише в якесь десятиліття
25 перейшли до використання методу пакетної комутації даних з використанням протоколу мережі Інтернет, з передачею даних через мережі стандартів Ethernet, Wi-Fi, 4G/LTE і DOCSIS3 по звичайним і оптоволоконним кабелями. Можна назвати багато переваг технології спільної передачі голосових, текстових даних і комп'ютерних даних, а також статичних та відеозображень; серед них - використання резервних шляхів передачі даних для гарантування
30 надійної доставки IP-пакетів (що є причиною, по якій Інтернет спочатку і створювався), а також безпрецедентний рівень взаємодії та можливостей по об'єднанню мереж в масштабах всього світу. Однак, як і у випадку з будь-якими інноваціями, ступінь складності завдань, які доводиться вирішувати в зв'язку з введенням в дію нових технологій, можна порівняти з кількістю наданих
35 ними переваг.

Недоліки існуючих провайдерів послуг зв'язку

Як було описано у вступній частині цієї розповіді, сучасні системи зв'язку характеризуються багатьма недоліками. Найпродуктивніші системи зв'язку сьогодні, що складаються з створеного на замовлення цифрового обладнання, яке знаходиться у власності найбільших світових
40 провайдерів послуг телекомунікації, таких як AT & T, Verizon, NTT, Vodafone і т.д., як правило, забезпечують чудову якість передачі голосу, але за високою вартістю, яка включає в себе високі тарифи щомісячної абонплати, тарифи на підключення послуг, тарифи на використання телекомунікації, складні плани обліку різних швидкостей передачі даних, оплату роумінгових послуг при використанні телекомунікації, а також різні тарифи на використання послуг з
45 технічного обслуговування обладнання. Так як форма власності зазначених мереж є приватною, інформація про фактичну безпеку передачі даних не знаходиться в громадському доступі; і факти порушення безпеки даних, здійснення хакерських атак і несанкціонованого втручання в роботу систем не оприлюднюються публічно. З урахуванням кількості фактів підслуховування телефонних розмов та порушення права на недоторканність приватного життя,
50 про які засоби масової інформації повідомляють щодня, існують серйозні сумніви в безпеці процесу зв'язку, що забезпечується приватними провайдерами відповідних послуг - під загрозою може перебувати безпека передачі даних, що знаходяться якщо не в приватній хмарі зазначених провайдерів, то, у крайньому разі, в процесі їх передачі з використанням з'єднань "останньої милі".

55 "Інтернет-провайдери" є ще однією ланкою в глобальній мережі комунікацій. Як описано в передмові до даного винаходу, процес передачі голосових повідомлень по мережі Інтернет з використанням технології VoIP характеризується безліччю проблем, пов'язаних з QoS, включаючи наступні:

- Інтернет, будучи мережею з пакетною комутацією даних, не призначений для регулярної в часі передачі IP-пакетів, або ж для підтримки додатків в режимі реального часу з низькими значеннями тимчасових затримок та високою якістю обслуговування.

5 - Характер маршрутизації IP-пакета є непередбачуваним, що призводить до виникнення затримок мінливою величини, раптовим "викидам" великої кількості помилок передачі даних, а також несподіваним "провалам" в передачі голосових даних.

- Маршрут передачі IP-пакета вибирається на розсуд Інтернет-провайдера, який визначає конкретну мережу, через яку буде здійснюватися передача пакетних даних, і може регулювати процес маршрутизації з урахуванням балансування завантаження своєї власної мережі або ж 10 кращого обслуговування своїх VIP-клієнтів за рахунок погіршення якості з'єднання основної маси своїх клієнтів - загального трафіку в своїй мережі.

- Провайдери послуг "низької якості" ("OTT providers"), такі як, наприклад, Line, KakaoTalk, Viber і т.д., використовують вільні канали передачі даних в мережі Інтернет, подорожуючи по ньому "автостопом" і, таким чином, не маючи можливості контролювати чинники, що 15 забезпечують якість передачі даних в мережі.

- Використання "великовагових" аудіокодеків, які не здатні забезпечити розбірливість переданих голосових даних навіть при помірних швидкостях передачі даних.

- Використання технології передачі голосових даних, що базується на застосуванні протоколу передачі даних TCP, призводить до високих тимчасових затримок та зниження якості 20 аудіопотока, що пов'язано з наявністю затримок, викликаних використанням протоколів "рукоштовування" та процесами повторної передачі IP-пакетів. При використанні в процесі передачі даних тільки протоколу UDP без допоміжних засобів, відсутні гарантії збереження цілісності корисного навантаження.

Окрім проблем низької якості обслуговування, рівень безпеки сьогоденних пристроїв і 25 мереж є абсолютно неприйнятним і не відповідає потребам майбутніх глобальних комунікацій. Як описано у вступі та було раніше показано на рисунку 40, рівень безпеки мереж перед обличчям великої кількості здійснюваних кібернетичних атак з використанням комунікаційних пристроїв, включаючи ті, на яких встановлено шпигунське програмне забезпечення, троянські програми, програми-віруси та програми для фішингу, є неадекватним; серед небезпек, що 30 чекають користувача на рівні "останньої ланки" - шпигунське програмне забезпечення, перехоплення пакетів, підслуховування телефонних розмов та перехоплення розмов з використанням шахрайських піратських базових станцій; на рівні місцевої мережі або частини мережі, що обслуговується місцевими телефонними компаніями (з'єднання "останньої милі") - шпигунське програмне забезпечення, перехоплення пакетів, вірусні програми та піратські "атаки посередників". До самої хмари може здійснюватися несанкціонований доступ, що 35 супроводжується зломом засобів забезпечення безпеки на будь-якому з шлюзів - тут вступають в дію такі фактори, як використання вірусного програмного забезпечення, атаки кіберпіратів ("атаки посередників"), DoS-атаки, а також несанкціоноване урядове стеження. Підсумувавши, можна сказати, що безпека сьогоденного процесу зв'язку знаходиться під загрозою з 40 урахуванням численних вразливостей, з легкістю використовуваних кіберпіратами, що забезпечують можливість здійснення кіберзлочинів або порушення приватності користувачів в кібернетичному просторі; серед зазначених видів уразливості можна назвати наступні:

- Наявність відкритої інформації про одержувача IP-пакета, включаючи IP-адресу одержувача, номер порту одержувача і MAC-адресу одержувача.

45 - Наявність відкритої інформації про джерело IP-пакета, включаючи IP-адресу джерела, номер порту джерела і MAC-адресу джерела.

- Наявність відкритої інформації про тип використовуваного транспортного рівня 4, а також (на основі інформації про номер порту) - про тип запитуваної послуги і даних програми, вкладених усередині корисного навантаження IP-пакета

50 - При використанні незашифрованих файлів - наявність доступу до всіх даних програми та файловим даним, закритим усередині корисного навантаження IP-пакета, - включаючи приватну та конфіденційну інформацію, дані входу в систему, паролі додатків, фінансові дані, а також відео- та фотодані.

55 - Наявність діалогу зв'язку, що дає можливість кіберзлочинцю здійснювати повторні спроби злому зашифрованих файлів

- Наявність численних можливостей по установці шкідливого ПО (включаючи шпигунське ПЗ, програми для фішингу та троянські програми) на пристрої зв'язку і маршрутизатори з використанням шляхів проникнення вірусів через протокол FTP, повідомлення електронної пошти та веб-сторінки.

Повторюючи ключове положення, можна заявити, що основним недоліком, характерним для мереж з пакетною комутацією, що використовують Інтернет-протокол (показано на рисунку 44) є те, що будь-яка вороже налаштована сторона або кіберпірат, перехопивши 670 IP-пакет, можуть бачити, які пристрої брали участь в процесі створення даних, що входять до складу IP-пакета, звідки прийшов цей IP-пакет, куди він пересилається зараз, як відбувається процес передачі даних (тобто, чи використовується протокол UDP або ж використовується протокол TCP), а також запит на надання якої послуги був відправлений (тобто, дані програми якого типу містяться всередині корисного навантаження). В даному відношенні, кіберпірат здатний визначити "контекст" розмови, що полегшує йому процес злому використовуваного коду шифрування, компрометації безпеки використовуваного пароля і діставання несанкціонованого доступу до файлів, даних та змістом корисного навантаження.

Шифрування - Для захисту від різного роду описаних кібератак, сучасні мережеві адміністратори, фахівці з інформаційних технологій та самі програмні додатки можуть використовувати, в основному, лише один засіб захисту - шифрування. Шифрування є засобом перетворення звичайного, простого тексту (тексту, призначеного для читання, тексту виконуваних програм, призначених до перегляду відеофайлів, файлів зображень або ж призначених до прослуховування аудіофайлів) в альтернативні типи файлів, відомі під назвою "зашифрований текст", які представляються у вигляді потоків беззмістовних текстових символів.

Процес шифрування, перетворення незахищеного файлу в зашифрований файл, включає в себе використання логічних або математичних алгоритмів, які називаються "кодом", що дозволяє перетворити дані в еквівалентні текстові елементи без демонстрації будь-якої очевидної моделі процесу перетворення, що протікає під час шифрування. Після цього зашифрований файл пересилається по мережі зв'язку або ж з використанням будь-якого комп'ютерного носія на пристрій-одержувач. Після отримання файлу отримуючий пристрій декодує зашифроване повідомлення з використанням процесу, відомого під назвою "дешифрування", для відображення оригінального вмісту файлу. Дисципліна, що вивчає процеси шифрування і дешифрування та відома під загальним найменуванням "криптографія", поєднує в собі елементи математики (включаючи теорію чисел, теорію множин та теорію розробки алгоритмів), комп'ютерної науки і електротехніки.

При використанні простих технологій шифрування, із застосуванням "одного ключа" або "симетричного ключа", для розблокування процесу шифрування і дешифрування файлу може використовуватися одне ключове слово або одна ключова фраза, заздалегідь відома обоим сторонам, що обмінюються інформацією. Наприклад, під час Другої світової війни підводні човни і морські судна обмінювалися повідомленнями по відкритих каналах радіозв'язку з використанням технології шифрування повідомлень. На самому початку, в процесі шифрування використовувався лише один ключ. Шляхом аналізу моделей коду, криптологи Альянсу іноді могли вирахувати ключове слово або модель, що використовувалися в процесі шифрування, і, таким чином, таємно прочитати зашифровані повідомлення. В міру ускладнення методів шифрування процес злому коду вручну все більше ускладнішався.

Код шифрування еволюціонував в шифр, заснованому на механіко-машинному обчисленні, яке було ранньою формою комп'ютерних обчислень. У той час єдиним способом злому коду була крадіжка шифрувального апарату та використання для дешифрування повідомлень тих же самих інструментів, які використовувалися і для їх шифрування. Найважчим завданням було викрасти шифрувальний апарат і не бути спійманим при виконанні цього завдання. Якщо ворогові ставало відомим, що таємниця шифрувальної машини була порушена, він просто змінював використовуваний код (шифр) і оновлював з урахуванням цієї зміни вже функціонуючі шифрувальні апарати. Даний принцип все ще практикується й сьогодні - найефективнішою кібератакою є та, яка не була виявлена.

З появою комп'ютерних технологій і настанням часів Холодної війни процес шифрування став більш складним, проте швидкість роботи комп'ютерів, що використовуються в процесі злому кодів шифрування, також збільшилася. На кожній стадії розвитку процесу безпечного зв'язку, технології та практичні навички шифрування інформації і можливості по злому коду шифрування розвивалися практично паралельно один одному, з однією та тією ж швидкістю. Важливий наступний крок у розвитку технологій шифрування був зроблений в 70-х роках, коли була представлена інноваційна технологія шифрування подвійним ключем; цей принцип використовується і по сьогодні. Одним з найбільш відомих методів шифрування подвійним ключем є використання криптосистеми відкритого ключа RSA-кодування - методу, названого за першими літерами імен його розробників: Rivest, Shamir і Adleman. Незважаючи на те, що в розвитку зазначеного методу офіційно визнані заслуги лише зазначених розробників, багато хто з їхніх сучасників незалежно від них також прийшли до використання подібного методу. Метод

RSA використовує два криптографічних ключі, в основі яких лежать два великих простих числа, які не розголошуються публічно. Один з алгоритмів використовується для перетворення зазначених двох простих чисел в ключ шифрування, який в цьому документі позначається як E-key; інший математичний алгоритм використовується для перетворення тих же двох таємних простих чисел в таємний ключ дешифрування, який в даному документі позначається також як D-key. Користувач системи RSA, який вибрав таємні прості числа (в даному документі позначається як "власник ключа"), поширює, або "публікує" даний ключ E-key, згенерований з використанням відповідного алгоритму і має розмір, який, як правило, варіюється від 1024 біт до 4096 біт, всім, хто бажає зашифрувати свій файл. У зв'язку з тим, що цей ключ відправляється багатьом користувачам в незашифрованому вигляді, ключ E-key відомий під назвою "відкритого ключа".

Сторони, які бажають підтримувати зв'язок з власником ключа, потім використовують вказаний відкритий ключ E-key в поєднанні з доступним алгоритмом, що знаходяться у відкритому доступі, який, як правило, пропонується в формі комерційного програмного забезпечення, в процесі здійснення шифрування будь-якого файлу, який передбачається переслати конкретному власнику ключа. При отриманні зашифрованого файлу власник ключа використовує таємний ключ D-key для дешифрування зазначеного файлу; в процесі даної дії файл перетворюється в звичайний незашифрований текст. Унікальністю методу подвійного ключа, в загальному випадку, і, конкретно, алгоритму RSA є той факт, що відкритий ключ E-key, який використовується для шифрування файлу, не може використовуватися для дешифрування. Для дешифрування може бути використаний тільки таємний ключ D-key, яким володіє власник ключа.

Концепція використання подвійного ключа, роздільного ключа або "множинного" ключа для шифрування і дешифрування файлів не обмежується методом RSA або будь-яким іншим алгоритмічним методом, але, з методологічної точки зору, описує метод здійснення зв'язку в якості певної послідовності кроків. Рисунок 47, наприклад, ілюструє процес обміну двома ключами під час процедури обміну інформацією засобами мережі з комутованими пакетами даних. Згідно ілюстрації, ноутбук 35, який "хоче" отримати захищений файл з стільникового телефону 32, спочатку генерує два ключі - ключ E-key 690 для шифрування і ключ D-key 691 для дешифрування, з використанням конкретного алгоритму. Ноутбук 35 потім пересилає ключ E-key 690 на стільниковий телефон 32 з використанням громадської мережі 692, через яку передається 695 IP-пакет. Процес пересилання 695 IP-пакета, очевидно, ілюструє використання незашифрованої форми, MAC-адреси, IP-адреси джерела (NB) і адреси порту №9999 ноутбука 35, а також IP-адреси одержувача (CP), порту №21 стільникового телефону 32 та протоколу передачі даних TCP, а також зашифрованої копії ключа E-key 690 в якості свого корисного навантаження.

Використовуючи заздалегідь узгоджений алгоритм шифрування або програмний пакет, стільниковий телефон 32 потім обробляє незашифрований текстовий файл 697A з використанням алгоритму шифрування 694A і ключа шифрування E-key 690, що дає на виході зашифрований файл, а саме, зашифрований текст 698, який передається в якості корисного навантаження 696 IP-пакета в процесі захищеної передачі даних 693 з мобільного телефону 32 на ноутбук 35. При отриманні 696 IP-пакета, для дешифрування файлу з використанням секретного ключа дешифрування (наприклад, ключа D-key 691) використано алгоритм 694B. Так як забезпечується узгодження ключа D-key 691 з ключем E-key 690, то, по суті, при роботі алгоритму 694B для дешифрування зашифрованого тексту 698 використовується інформація про обидва ключі; в процесі забезпечується створення незашифрованого звичайного тексту 697B. У той час як корисне навантаження 696 IP-пакета захищене в формі зашифрованого файлу (тобто, зашифрованого тексту 698), інша частина IP-пакета і раніше пересилається в незашифрованій формі (що відкриває можливість її перехоплення), відкритої до прочитання будь-яким кіберпіратом, включаючи інформацію про IP-адресу джерела (CP) та номер порту (№20), а також про IP-адресу одержувача (NB) та пов'язаному номера порту (№9999). Отже, навіть якщо корисне завантаження саме по собі і не може бути відкрито, інші складові процесу обміну інформацією як і раніше можуть відслідковуватися.

Віртуальні приватні мережі - Іншим методом забезпечення безпеки, також заснованим на використанні процедури шифрування, є використання так званої віртуальної приватної мережі, або VPN (Virtual Private Network). При використанні VPN в мережі, де відбувається передача зашифрованих IP-пакетів, формується "тунель" або "безпечний канал". При використанні VPN шифрується не тільки корисне навантаження, але також і вміст всього IP-пакета, який потім "вбудовується" в інший незашифрований IP-пакет, виконуючи функцію "переносника" даних, за допомогою якого інкапсульований пакет передається від одного шлюзу VPN до іншого.

Спочатку, віртуальні приватні мережі використовувалися для об'єднання нерівноправних локальних мереж в одне ціле на великій відстані - наприклад, коли компанії-власники приватних мереж в Нью-Йорку, Лос-Анджелесі і Токіо хотіли об'єднати свої різні локальні мережі в одну з використанням такої ж функціональності, як нібито це була одна глобальна приватна мережа.

Основний принцип роботи віртуальних приватних мереж показаний на рисунку 48А, де сервер 700, будучи частиною однієї локальної мережі, яка підтримує певну кількість пристроїв в бездротовому режимі з використанням радіоз'єднань 704 та в дротовому режимі з використанням з'єднань 701, з'єднується за допомогою "віртуальної приватної мережі" (VPN), що складається з контенту 706 і VPN-тунелю 705, з другим сервером 707, який має провідні з'єднання 708 з настільними комп'ютерами 709А через 709С, з ноутбуком 711, а також з базовою станцією Wi-Fi 710. На додаток до вказаного з'єднання, що характеризується відносно низькою пропускну здатністю, сервер 707 з'єднаний також з суперкомп'ютером 713 за допомогою високошвидкісного з'єднання 712. У процесі функціонування даної системи, зовнішній 714 IP-пакет з сервера А (із зазначенням IP-адреси джерела (S8) і номера порту (№ 500)) відсилається на сервер з IP-адресою одержувача S9 і номером порту 500. Зазначений зовнішній 714 IP-пакет описує процес формування серверами 700 і 707 зашифрованого "тунелю" передачі даних між ними в прямому та зворотному напрямку. Корисне навантаження VPN зовнішнього пакета 714 містить в своєму складі 715 IP-пакет "останньої милі", що забезпечує безпосередній обмін інформацією між настільним комп'ютером 702В з IP-адресою джерела DT та відповідним спеціально створеним для цієї мети портом №17001, а також ноутбуком 711 з IP-адресою джерела NB та відповідним спеціально створеним для цієї мети портом №21; зазначений пакет несе в собі запит на передачу файлів.

Для того щоб здійснити зазначену передачу даних в безпечному режимі з використанням віртуальної приватної мережі, був створений тунель VPN 705, та було ініційовано сесію ще до того, як сталася фактична передача даних. При використанні в корпоративному середовищі, тунель VP 705 не надсилається через Інтернет з використанням спеціально створених для цього портів, але, в загальному випадку, пересилається з використанням спеціально виділеного провайдера Інтернет-послуг або провайдера пов'язаних послуг, що має в своєму розпорядженні власний волоконно-оптичний кабель і мережу апаратних засобів. Провайдер послуг часто укладає з компанією щорічні або довгострокові контакти, за умовами яких гарантується певна пропускна здатність VPN-серверів за певну плату. В ідеальному випадку вказаний виділений високошвидкісний канал зв'язку використовується для прямого з'єднання серверів 700 і 707 без проміжних з'єднань (або "з'єднань останньої милі"), застосування яких могло б негативно відбитися на швидкості передачі даних через віртуальну приватну мережу, якість надаваних послуг або безпеки переданих даних.

На практиці, використання традиційних віртуальних приватних мереж вимагає наявності двоступеневого процесу - на першій стадії створюється віртуальна приватна мережа або здійснюється "вхід" в неї, а на другому - відбувається передача даних всередині безпечного каналу, або "тунелю". Концепція тунелювання ілюструється ієрархічно на рисунку 48В, де зовнішні IP-пакети, що переносяться стеками передачі даних 720 і 721, формують VPN-з'єднання 722 на рівнях з 1 по 4, використовують Рівень 5 для створення віртуального VPN-сеансу 723 та використовують Рівень 6 - Рівень представлення - для сприяння процесу шифрування 725, для забезпечення процесу передачі інформації між шлюзами з використанням VPN-каналу 705 між серверами 700 і 707. У той час як VPN-з'єднання 722 використовує Інтернет-протокол для пересилання IP-пакетів, Рівень 1 протоколу фізичного рівня VPN і Рівень 2 канальний рівень передачі даних VPN, в загальному випадку, підтримуються виділеним провайдером послуг і не використовують непередбачуваних маршрутів передачі даних через мережу Інтернет. Дані Прикладного Рівня 6, що передаються в якості потоку інформації "від пристрою до пристрою" 706 між, наприклад, настільними комп'ютерами 702С і 709А, передаються в якості інкапсульованих (тунельованих) даних 726, включаючи всі сім рівнів OSI, необхідних для встановлення зв'язку так, як нібито віртуальної приватної мережі не існувало.

В процесі функціонування зовнішній IP-пакет з стека передачі даних 720 відразу після його пересилання на сервер 707 відкривається для виявлення інкапсульованих даних 726 - істинного вмісту пакета. Таким чином, в процесі наскрізної передачі даних в мережі не виявляються подробиці формування VPN-тунелю, за винятком того факту, що VPN-тунель повинен бути утворений перш, ніж буде здійснена якась спроба передачі даних, та повинен бути закритий після закінчення розмови. Якщо VPN-тунель не буде відкритий до початку зазначеного процесу, це призведе до того, що буде здійснюватися передача незашифрованого 715 IP-пакета, дані якого можуть бути перехоплені, викрадені, заражені вірусом і т.д. Якщо VPN-тунель не буде закритий після закінчення розмови, це може надати кіберзлочинцеві можливість приховати свою

протизаконну діяльність всередині чийогось VPN-тунелю і, в разі її виявлення, може призвести до того, що підозра в скоєнні кримінального злочину буде пред'явлена невинній особі.

У той час як використання віртуальних приватних мереж є звичайним способом забезпечення з'єднання множинних приватних локальних мереж між собою з використанням приватних з'єднань з виділеною пропускнуою спроможністю і шириною пропускання, використання віртуальних приватних мереж в складі публічних мереж і Інтернету є проблематичним при спілкуванні між двома зацікавленими сторонами. Однією з проблем використання віртуальних приватних мереж є те, що VPN-з'єднання повинно бути встановлено заздалегідь, ще до того, як воно зможе бути використано, а не на основі взаємного обміну пакетами даних. Наприклад (згідно з рисунком 48С на прикладі дзвінка з використанням передачі голосу по IP-протоколу (технологія VoIP) з з'єднанням, реалізованим засобами мережі з пакетною комутацією даних), перед тим як буде забезпечено з'єднання стільникового телефону 730 з адресатом (стільниковим телефоном 737), спочатку повинен бути створений VPN-сеанс з використанням інструкції 740 простого алгоритму, згідно ілюстрації. При цьому стільниковий телефон 730 з VPN-додатком відсилає IP-пакети на VPN-хост 733 з використанням будь-якого доступного маршруту "останньої милі" (в даному випадку - радіоканалу зв'язку 741А на базову станцію Wi-Fi 731, після чого інформація відсилається з використанням дротових систем зв'язку 741В на маршрутизатор 732, а потім, з використанням дротових систем зв'язку 741С - на VPN-хост 733). Як тільки сеанс зв'язку між стільниковим телефоном 730 і VPN-хостом 733 буде відкритий, стільниковий телефон 730 видасть інструкцію VPN-хосту 733 по створенню VPN-тунелю 741, що веде до VPN-хосту 734; відбувається узгодження параметрів між Сеансовим Рівнем 5 та тунелем, зашифрованим з використанням Рівня 6.

Як тільки VPN-з'єднання буде встановлено, стільниковий телефон 730, відповідно до інструкцій додатку 745, починає здійснювати дзвінок з використанням будь-якого VoIP-телефонного додатку. При виконанні даної інструкції додаток повинен встановити з'єднання "виклику" з використанням систем "останньої милі" між VPN-хостом 734 та стільниковим телефоном 737. Якщо VoIP-додаток не може цього зробити або не має відповідних прав на проведення подібних дій, дзвінок не відбудеться - він буде негайно перерваний. В іншому випадку внутрішній IP-пакет встановить сеанс зв'язку Рівня 5 додатка між викликаючим стільниковим телефоном 730 та стільниковим телефоном-адресатом 737, а також підтвердить той факт, що тестові IP-пакети належним чином дешифровані і розбірливі.

Для того щоб здійснити дзвінок відповідно до інструкції 745, дзвінок обов'язково повинен бути ініційований Рівнем 7 додатка, що працює на телефоні, а не звичайними телефонними функціями набору номера, тому що SIM-карта провайдера послуг телефонії, встановлена в телефоні, не сумісна з функціями VPN-тунелю. Як тільки виклик буде здійснений, стільниковий телефон 730 передаватиме послідовність IP-пакетів, що представляє собою малі фрагменти (або "уривки") голосової інформації, відібрані відповідно до встановленого на телефоні додатком зв'язку. У наведеному прикладі ці фрагменти відсилаються додатком, встановленим на телефоні абонента 730, через канал Wi-Fi 746А на базову станцію Wi-Fi 731, а потім через дротове з'єднання 746В - на маршрутизатор 732 і, нарешті, через дротове з'єднання 746С - на VPN-хост 733. Після цього дані пересилаються через захищений канал зв'язку за допомогою з'єднання 747 на VPN-хост 735 через VPN-тунель 742. Як тільки дані покинуть VPN-тунель, VPN-хост перешле дані на дротове з'єднання 748А, на маршрутизатор 735, а потім, через дротове з'єднання 748В - на систему стільникового зв'язку та вишку 736, яка, в свою чергу, здійснює виклик 737 в якості звичайного телефонного виклику. Процес здійснення виклику з додатка стільникового телефону на телефон, на якому не встановлений аналогічний додаток, носить назву процесу "зовнішнього" виклику.

З наведеного вище прикладу можна бачити, що існує ще одна проблема при підключенні до віртуальної приватної мережі з використанням публічної мережі - з'єднання "останньої милі", як між абонентом (стільниковим телефоном 730) і VPN-хостом 733, так і між VPN-хостом 734, що здійснює "зовнішній" виклик, та приймаючим абонентом (стільниковим телефоном 737), не є складовою частиною віртуальної приватної мережі, а отже, не гарантують належної безпеки, швидкості з'єднання або ж належної якості послуг. Висловлюючись більш конкретно, "остання миля" абонента, що складається із з'єднань 746А, 746В і 746С, а також сполуки "зовнішнього" виклику 748А, 748В і 748С, вразлива до кібератак і перехоплення даних.

Як тільки дзвінок був завершений, та стільниковий телефон 737 був відключений від зв'язку, віртуальна приватна мережа 742 повинна бути закрита відповідно до інструкцій 749, де Рівень 5 віртуальної приватної мережі здійснює координацію процесу закриття VPN-сеансу, та стільниковий телефон 730 відключається від VPN- хоста 733.

Навіть якщо всі зазначені інструкції були виконані, все одно немає гарантії того, що спроба здійснення телефонного дзвінка або пересилання документів через віртуальну приватну мережу не опиниться безуспішною з кількох причин, включаючи наступні:

- Затримка передачі даних у віртуальній приватній мережі може виявитися занадто великою, що унеможливить роботу додатків в режимі реального часу, передачу голосу по IP-протоколу або передачу потокового відео;

- Затримка передачі даних в з'єднанні "останньої милі" віртуальної приватної мережі абонента, який телефонує абонентом і шлюзом VPN або між шлюзом VPN і адресатом телефонного дзвінка може виявитися занадто великою, що унеможливить роботу додатків в режимі реального часу, передачу голосу по IP-протоколу або передачу потокового відео;

- Найближчий до телефонуючого абонента або до абонента-адресата шлюз VPN (тобто, з'єднання "останньої милі") може виявитися розташованим на занадто великій відстані – можливо, навіть більше за відстань до абонента-адресата, без використання VPN, що призведе до значної затримки передачі даних всередині з'єднання, нестабільної роботи мережі, неконтрольованої маршрутизації з використанням невідомих мереж, варіюванню якості обслуговування та численним можливостям для здійснення атаки "людина посередині" з використанням незахищених сегментів з'єднання;

- З'єднання "останньої милі" віртуальної приватної мережі між шлюзом VPN і абонентом-адресатом виклику може не підтримувати функцію "зовнішнього" виклику та передачі пакетів, або ж не підтримувати з'єднання з місцевими телефонними компаніями;

- Місцеві провайдери послуг або урядові цензори можуть блокувати виклики, які надходять від невідомих шлюзів VPN, або ж процес встановлення з'єднання з такими шлюзами з причин, пов'язаних з необхідністю забезпечення національної безпеки або відповідності нормам законодавства;

- При використанні корпоративних віртуальних приватних мереж можливість здійснення викликів з використанням функціональності технології VoIP може надаватися тільки співробітникам компанії або чітко визначеним уповноваженим категоріям користувачів; процеси проведення фінансових транзакцій або перегляду онлайн-відео можуть бути заблоковані; можуть бути заблоковані сервери передачі приватних поштових повідомлень в публічну мережу, такі як, наприклад, Yahoo, Google і т.д.; а також може бути заблокований доступ до численних веб-сайтів, таким як YouTube, програмам онлайн-чатів, або ж, наприклад, сервісу Twitter - відповідно до політики компанії;

- У разі нестабільної роботи мереж, віртуальна приватна мережа по завершенні дзвінка може залишитися у відкритому стані, що виразиться в підтримці поточної розмови в безперервному режимі з виходом на пристрій абонента до тих пір, поки не буде здійснено скид з'єднання VPN-провайдером в ручному режимі. Це може привести до втрати частини пропускну здатності при здійсненні подальших дзвінків або ж до необхідності виплати великих сум за тарифами за незакрите з'єднання.

Порівняння мереж - порівняння комунікацій, пропонованих "over-the top" або OTT-провайдерами, показане на рисунку 49А, з системами зв'язку, що використовують загальнодоступні мережі для підключення до спеціального VPN, показаним раніше на рисунку 48С, швидко виявляє, що крім самого VPN-зв'язку, більшість обох систем зв'язку мають майже ідентичні компоненти та з'єднання. Зокрема, "остання миля" абонента, що має стільниковий телефон 730, радіозв'язок Wi-Fi 746А, базову станцію Wi-Fi 731, дротяні з'єднання 746В і 746С і маршрутизатор 732, являють собою той самий зв'язок "останньої милі" в обох реалізаціях. Аналогічно, на "останній милі" іншого боку, стільниковий телефон 737, з'єднання стільникового телефону 748С, базова станція стільникового зв'язку та вишка 736, дротяні з'єднання 748А і 748В і маршрутизатор 735 ідентичні для версій Інтернету і VPN. Основна відмінність полягає в тому, що в загальнодоступній мережі VPN-тунель 742 з захищеним зв'язком 747 між VPN-вузлами 733 і 734 замінюється серверами/маршрутизаторами 752 і 754, що забезпечують небезпечне комунікаційне з'єднання 755. Інша відмінність полягає в зв'язку OTT, виклик доступний миттєво, як описано в інструкції 750, де для використання VPN необхідні додаткові інструкції 740 і 749 для настройки VPN та завершення сеансу VPN перед наступним викликом.

В обох прикладах з'єднання "останньої милі" пропонують непередбачувану якість послуг зв'язку, можливість перехоплення пакетів і ризик кібератак. Оскільки сервери/маршрутизатори 752 і 774, ймовірно, управляються різними провайдерами в різних локальних мережах, можна інтерпретувати сервери як існуючі різні хмари, тобто хмари 751 і 753. Наприклад, публічно відкриті мережі, що належать і керовані Google, Yahoo, Amazon і Microsoft, можуть розглядатися як різні хмари, наприклад, "хмара Amazon", хоча всі вони пов'язані Інтернетом.

Конкуруюча мережева топологія, однорангова мережа або PPN, показана на рисунку 49В, включає мережу, що складається з великого числа тимчасових вузлів з маршрутизацією пакетів, керованих PPN, а не маршрутизатором або провайдером. У той час як однорангові мережі існували на апаратних засобах протягом десятиліть, саме Napster популяризував цю концепцію як засіб уникнути контролю, витрат і регулювання Інтернет-провайдерів. Зазнаючи переслідувань в судовому порядку з боку урядових органів США за порушення авторських прав на музику, засновники Napster були змушені оголосити про банкрутство компанії; в цей же час, OTT-оператор Skype тільки починав розвиватися. Пізніше, мережа Skype відійшла від традиційної OTT моделі мережі до пірингової, запропонованої компанією Napster.

У PPN-комутації кожен пристрій, який підключається для входу в PPN, стає ще одним вузлом в PPN. Наприклад, якщо в мережі з топологією 761 стільниковий телефон 730 з встановленим ПЗ PPN реєструється в тимчасовій мережі, він, як і всі інші підключені пристрої в регіоні, стає частиною мережі. Виклики, що видаються будь-якими пристроями, переходять з одного пристрою на інший, щоб дістатися до пункту призначення, іншого пристрою, підключеного до PPN. Наприклад, якщо в мережі з топологією 761 стільниковий телефон використовує своє з'єднання PPN для виклику іншого пристрою з підключеним PPN, наприклад, стільникового телефону 768, виклик слідує обхідним шляхом через будь-який пристрій (пристрої), фізично розташований в PPN між цими двома пристроями. Як показано на рисунку, виклик, що виходить від стільникового телефону 730, з'єднується Wi-Fi 731 через базову станцію Wi-Fi 731 з настільним комп'ютером 765А, потім з ноутбуком 766А, зі стаціонарним комп'ютером 765В, потім зі стаціонарним комп'ютером 765С і, нарешті, з стільниковим телефоном 768 через базову станцію стільникового телефону та вишку 767. Таким чином, вся маршрутизація контролюється PPN, а Інтернет не бере участі в управлінні маршрутизацією. Оскільки задіяні обидві сторони, ПЗ PPN, що використовується для підключення до мережі, також виступає в якості додатку для голосового зв'язку на основі технології VoIP.

У разі, коли стільниковий телефон 730 намагається викликати стільниковий телефон 737 з пристроєм не на основі PPN на протилежному кінці землі, маршрутизація повинна обов'язково включати передачу даних через Інтернет, використовуючи точки доступу, особливо при передачі пакетів через океани або гірські хребти. Перша частина маршрутизації в мережі з топологією 761 протікає аналогічно до попереднього прикладу, починаючи з стільникового телефону 730 та проводить маршрут через базову станцію 731 Wi-Fi, настільний комп'ютер 765А, ноутбук 766А, настільні комп'ютери 765В і 765С. На цей момент, якщо ноутбук 766В підключений до мережі, виклик буде прокладений через нього, в іншому випадку виклик повинен бути направлений через базову станцію стільникового телефону та вишку 767 на стільниковий телефон 768, а потім повернутий назад до базової станції стільникового телефону та вишки 767 перед подальшою відправкою.

Якщо виклик перетинає Тихий океан, тоді комп'ютери та стільникові телефони не можуть переносити трафік через океан, тому дзвінок потім обов'язково маршрутизується в Інтернет на сторонній сервер/маршрутизатор 770 в хмарі 763 і далі за допомогою бездротової технології 747 на сторонній сервер/маршрутизатор 771 в хмарі 764. Потім виклик виходить з Інтернету та входить в PPN в мережі з топологією 762 спочатку через стаціонарний комп'ютер 772, який, в свою чергу, підключається до Wi-Fi 773, до ноутбука 776 і до базової станції 736. Оскільки Wi-Fi 733 не працює з програмним забезпеченням PPN додатка, фактичний пакет, що входить в Wi-Fi 773, повинен відправитися на планшет 775 або стільниковий телефон 774 і назад на Wi-Fi 773, перш ніж вирушати на базову станцію стільникового телефону та вишку 736 через дротове з'єднання. Нарешті, виклик стільникового телефону 748С підключається до стільникового телефону 737, який не є пристроєм з підтримкою PPN. Таким чином, з'єднання є "викликом" для PPN, оскільки воно виходить з мережі з топологією 762 PPN. Використовуючи цей підхід PPN, VPN спочатку реєструє зухвалий пристрій в мережі PPN згідно з інструкцією 760, виконавши авторизацію PPN. Після цього виклик може бути виконаний через додаток PPN згідно з інструкцією 769. Перевага такої пірингової мережі в тому, що для передачі виклику на великі відстані необхідності в додатковому обладнанні немає, і, так як кожен пристрій, підключений до PPN, регулярно оновлює свій статус, завантаження та затримку, провайдер PPN може керувати маршрутизацією пакета для мінімізації затримки.

Недоліки такого підходу в тому, що пакети перетинають мережу, що містить безліч невідомих вузлів, що представляють потенційну загрозу безпеці і мають непередбачуваний вплив на затримку і якість обслуговування викликів. Таким чином, за винятком Skype, однорангові мережі, що працюють на Рівні 3 та вище, як правило, не використовуються в мережах зв'язку з комутацією пакетів.

Загальне порівняння провайдерів спеціальних VPN, Інтернет-провайдерів ОТТ та тимчасових мереж PPN наведене нижче.

Мережа	Віртуальна приватна мережа VPN	ОТТ Мережа	Однорангова мережа PPN
Вузли	Публічні/ Хост-сервера	Публічні маршрутизатори/сервери	Користувачі PPN
Пропускна спроможність вузла	Відома інфраструктура	Відома інфраструктура	Змішана, невідома
Хмарна пропускна здатність	Гарантована	Непередбачувана	Непередбачувана
Пропускна здатність "останньої милі"	Залежить від провайдера	Залежить від провайдера	Залежить від PPN
Затримка	Некерована	Некерована	Оптимальний варіант
Стабільність мережі	Некерована	Некерована, надлишкова	Оптимальний варіант
Налаштування виклику	Комплексний вхід	Не вимагається	Авторизація
Ідентифікація користувача	Ім'я користувача	Номер телефону	Ім'я користувача
Якість обслуговування технології VoIP	Від змінного до хорошого	Змінна	Змінна
Хмарна безпека	Шифрується тільки корисне навантаження	Незашифрована	Незашифрована
Захист "останньої милі"	Незашифрована	Незашифрована	Незашифрована
Схильність до перехоплення	Заголовок пакета (Хмара) Весь пакет (Остання миля)	Весь пакет	Весь пакет

5 Як показано на рисунку, в той час як VPN та Інтернет мають фіксовану інфраструктуру, вузли тимчасової мережі розрізняються залежно від того, хто зареєстрований і які пристрої підключені до PPN. Хмарна пропускна здатність, представлена в контексті цієї таблиці як високошвидкісні міжміські з'єднання мереж, наприклад, мережі, які перетинають океани і гірські ланцюги, гарантується за контрактом тільки в разі VPN, в іншому разі – вона непередбачувана.

10 Пропускна здатність "останньої милі" - це локальний провайдер, що залежить як від Інтернет-провайдерів, так і від VPN, але для PPN повністю залежить від того, хто увійшов в систему.

Затримка поширення послідовних IP-пакетів некерована для ОТТ і VPN, оскільки провайдер не контролює маршрутизацію в "останньої милі", але замість цього залежить від місцевих телефонних операторів або інтернет-провайдерів, тоді як PPN мають обмежені можливості, використовуючи оптимальні варіанти для прямого трафіку між вузлами, які в даний момент знаходяться в мережі в певній географії. Аналогічно, для стійкості мережі, PPN мають можливість перенаправляти трафік, щоб підтримувати мережу, але повністю залежать від того, хто входить в систему. Інтернет, з іншого боку, є внутрішньо надлишковим і майже впевнено гарантує доставку, але не обов'язково своєчасну. Стійкість мережі для спеціального VPN залежить від кількості вузлів, дозволених для підключення до хосту VPN. Якщо ці вузли відключені, VPN буде непридатний.

15

20

З точки зору налаштування виклику Інтернет завжди доступний, PPN вимагають додаткової інструкції для входу в PPN до здійснення дзвінка, а VPN можуть включати складну процедуру входу в систему. Більш того, більшість користувачів вважають використання ОТТ телефонних номерів, а не окремих ідентифікаторів входу, використовуваними VPN і PPN, в якості важливого переваги для простоти користування. Всі перераховані три мережі страждають від нестійкої якості технології VoIP, в цілому відстаючи від комерційних операторів зв'язку. З точки зору безпеки всі три варіанти не можуть виступати в якості з'єднань "останньої милі", так як схильні до перехоплення пакетів з відкритими адресами і корисними навантаженнями. VPN забезпечують шифрування хмарного з'єднання, але все ж надають IP-адреси хостів VPN. Таким чином, ніяка мережева опція не вважається безпечною. Таким чином, шифрування використовується

25

30

різними додатками для запобігання злому і кібератакам, або як протокол Рівня 6, або як впроваджена частина Прикладного рівня 7.

Надмірна довіра до шифрування. Незалежно від того, чи використовується вона для шифрування IP-пакетів або створення VPN, сьогодняшня мережева безпека залежить майже виключно від шифрування і являє собою одну слабку сторону в сучасних мережах зв'язку на основі пакетної комутації. Наприклад, було проведено безліч досліджень по методам здійснення атак на шифрування RSA. Хоча обмеження простих чисел на великі розміри значно знижує ризик "злому" коду D-ключа дешифрування з використанням методів "грубої сили", методи поліноміального коефіцієнта успішно продемонстрували злом ключів на основі менших простих чисел. Існують побоювання, що еволюція "квантових обчислень" в кінцевому підсумку призведе до практичних методів "злому" ключів RSA і інших ключів шифрування при кібератаці. Для боротьби з постійно присутнім ризиком "злому" коду з'явилися нові алгоритми і методи шифрування "більшого ключа", такі як "покращений стандарт шифрування" або шифрування AES, затверджений US NIST в 2001 році. Принцип проектування, заснований на шифрі Rijndael, відомому як підстановочно-перестановочна мережа, поєднує в собі як заміну символів, так і перестановку з використанням різних розмірів ключа і блоку. У своєму нинішньому втіленні алгоритм містить фіксовані розміри блоків 128 біт з ключами, що включають в себе різні довжини 128 біт, 192 біта і 256 біт, з відповідною кількістю повторень, використовуваних при перетворенні вхідного файлу, що варіюються в круглих числах в 10, 12, і 14 ступенях відповідно. У практичному плані шифр AES можна ефективно і швидко виконувати як в програмному, так і в апаратному забезпеченні для будь-якого розміру ключа. У криптографічній мові шифрування на основі AES з використанням ключа 256b називається "шифруванням AES256". Також є шифрування AES512 з використанням ключа 512b.

У той час як кожне нове покоління піднімає планку криптографії, щоб зробити кращими методи шифрування, зацікавлені в прибутку кіберзлочинці, щоб швидше "зламати" ці методи, часто концентруються на своїх цілях, замість того, щоб використовувати обчислення для злому зашифрованого файлу. Як описано раніше, використовуючи перехоплення пакетів і опитування портів, кіберпірат може отримати цінну інформацію про розмову, корпоративний сервер або навіть VPN-шлюз. Замість атаки на мережу, кібераналіз дозволяє атакувати персональні комп'ютери, ноутбуки і мобільні телефони Головного або Фінансового директора компанії. Як тільки співробітники компанії переходять по посиланню, що вкладене в електронні листи, автоматично встановлюється шкідливе і шпигунське ПЗ, яке повністю обходить захист брандмауера, оскільки таким чином шкідливі програми отримують доступ "зсередини".

Імовірність злому шифру також збільшується, якщо дані переміщуються по мережі без зміни, тобто статично. Наприклад, в мережі на рисунку 50 базові дані в пакетах 790, 792, 794 і 799 залишаються незмінними в міру того, як пакети переміщуються по мережі. Кожен показаний пакет даних містить послідовність даних або звуку, розташованих послідовно за часом або сторінок, незмінених від вихідного порядку, коли він був створений. Якщо вміст пакета даних є текстовим, читання незашифрованого текстового файлу в послідовності 1A-1B-1C-1D-1E-1F призведе до "розбірливого" тексту для комюніке номер "1". Якщо вміст пакета даних являє собою аудіо, перетворення, тобто "відтворення", незашифрованого текстового файлу в послідовності 1A-1B-1C-1D-1E-1F через відповідний аудіокодек, по суті, заснований на програмному забезпеченні ЦАП, призведе до звуку для номера звукового файлу номер "1".

У будь-якому випадку, в розглядуваному описі, кожен інтервал даних, представлений блоками фіксованого розміру, містить задану кількість біт, наприклад, два байта (2B). Точна кількість біт на кожен інтервал є гнучким тільки в тому випадку, якщо кожен вузол зв'язку в мережі знає, який розмір кожного інформаційного інтервалу. У кожному інтервалі даних містяться аудіо-, відео- або текстові дані, зазначені на кресленнях як число, за яким слідує буква. Наприклад, як показано на рисунку, перший інтервал пакету 790 даних містить контент 1A, де число "1" вказує конкретний зв'язок №1, а буква "A" представляє першу частину даних в повідомленні №1. Аналогічно, другий інтервал пакету 790 даних містить вміст 1B, де число "1" вказує, що воно є частиною одного та того ж повідомлення №1, а буква "B" представляє другу частину даних в повідомленні №1, послідовно дотримуючись за 1A.

Якщо, наприклад, один та той же пакет даних імовірно включає в себе контент "2A", дані представляють перший пакет "A" в іншому повідомленні, зокрема для зв'язку №2, не пов'язаного з повідомленням №1. Пакети даних, що містять однорідні повідомлення, наприклад, де всі дані для зв'язку №1 легше аналізувати та читати, ніж ті, які змішують різні повідомлення. Дані, впорядковані послідовно в правильному порядку, дозволяють кіберзлочинцю інтерпретувати природу даних, будь то аудіо, текст, графіка, фотографії, відео, виконуваний код і т.д.

Більш того, в показаному прикладі, оскільки IP-адреси джерела та призначення пакета залишаються постійними, тобто коли пакети залишаються незмінними під час транспортування по мережі в тій же формі, що і дані, що входять або виходять з шлюзових серверів 21A і 21F, оскільки базові дані не змінюються, у хакера більше шансів перехопити пакети даних і отримати більше шансів проаналізувати та відкрити файли або прослухати розмову. Проста транспортна і одномірна безпека, тобто така, що покладається тільки на шифрування для захисту, збільшує ризик кібератаки, тому що ймовірність успіху вище при такому надмірно простому використанні Інтернету в якості мережі з комутацією пакетів.

Захист мереж реального часу і мережевих пристроїв

Для того, щоб поліпшити якість обслуговування телефонії, відео та передачі даних, одночасно усуваючи безліч уразливих місць в системі безпеки, що створюють проблеми сучасним мережами з комутацією пакетів, необхідний новий та інноваційний системний підхід до управління маршрутизацією IP-пакетів, який управляє глобальною мережею, яка включає розрізнені технології і одночасно сприяє наскрізній безпеці. Завдання такої інноваційної мережі з комутацією пакетів включають такі критерії:

1. Забезпечити безпеку і якість обслуговування глобальної мережі або провайдера послуг телекомунікації, включаючи динамічне управління маршрутизацією трафіку голосу, відео і даних в режимі реального часу в мережі;

2. Застрахувати безпеку і якість обслуговування "локальної мережі або телефонної компанії" в останній милі мережі зв'язку;

3. Забезпечити безпеку і якість обслуговування "останньої ланки" мережі зв'язку, включаючи забезпечення безпечного зв'язку, що проходить по незахищених лініях;

4. Забезпечити безпеку комунікаційних пристроїв і автентифікувати користувачів для запобігання несанкціонованого або шахрайського доступу або використання;

5. Сприяти безпечному способу зберігання даних на пристрої, в онлайн, в мережі або в хмарному сховищі даних, щоб уникнути несанкціонованого доступу;

6. Забезпечити безпеку і конфіденційність всієї неопублічної особистої інформації, включаючи всі фінансові, особисті, медичні і біометричні дані та записи;

7. Забезпечити безпеку і конфіденційність всіх фінансових транзакцій, пов'язаних з онлайн-банкінгом та покупками, кредитними картами і електронною оплатою; а також забезпечити безпеку, конфіденційність і анонімність в проведенні транзакцій і обміні інформацією при міжмашинній (M2M) і міжавтомобільній (V2V) взаємодіях, а також при обміні інформацією між автомобілем та транспортною інфраструктурою (V2X).

З вищевикладених цілей зміст винаходу, що приводиться в даному описі, відноситься до першої теми, описаної в пункті №1, тобто "забезпечити безпеку і якість обслуговування глобальної мережі або провайдера послуг телекомунікації, включаючи динамічне управління маршрутизацією трафіку голосу, відео і даних в режимі реального часу в мережі". Дане питання можна розглядати як досягнення безпеки мережі або хмари без шкоди для експлуатаційних характеристик зв'язку в реальному часі.

Глосарій

Якщо контекст не вимагає іншого, терміни, використовувані в описі динамічної захищеної комунікаційної мережі та протоколу, мають таке значення:

Анонімні пакети даних: Пакети даних, що не містять інформації про початкового відправника або кінцевого одержувача.

Дешифрування: Математична операція, яка використовується для перетворення пакетів даних з зашифрованого тексту в незашифрований текст.

Сервер DMZ (англ. Demilitarized Zone - демілітаризована зона): Комп'ютерний сервер, недоступний безпосередньо з мережі SDNP (англ. Secure Dynamic Network And Protocol - динамічна захищена комунікаційна мережа та протокол) або Інтернету, який використовується для зберігання селекторів, генераторів початкових станів, генераторів ключів та інших спільних секретів.

Динамічне шифрування/дешифрування: Шифрування і дешифрування, засновані на ключах, які динамічно змінюються в міру проходження пакету даних через мережу SDNP.

Динамічне змішування: Процес змішування, коли алгоритми змішування (зворотні по відношенню до алгоритмів розділення) динамічно змінюються в залежності від початкового стану на основі стану, наприклад, часу, стану та зони, коли створюється змішаний пакет даних.

Динамічне скремблювання/дескремблювання: скремблювання та дескремблювання, засновані на алгоритмах, які динамічно змінюються в залежності від стану, наприклад, коли створюється пакет даних або зона, в якій він створюється.

Динамічне розділення: Процес розділення, коли алгоритми розділення динамічно змінюються в залежності від початкового стану на основі стану, наприклад, часу, стану та зони, коли пакет даних розділяється на кілька субпакетів.

5 Шифрування: Математична операція, яка використовується для перетворення пакетів даних з незашифрованого тексту в зашифрований текст.

Передача фрагментованих даних: Маршрутизація розділених та змішаних даних по мережі SDNP.

10 Видалення "сміттєвих" даних (або "видалення інформаційного сміття"): Видалення "сміттєвих" даних з пакетів даних для відновлення вихідних даних або відновлення початкової довжини пакета даних.

Вставка "сміттєвих" даних (або "вставка інформаційного сміття"): Навмисне введення в пакет даних беззмістовної інформації, або для ускладнення розуміння реального вмісту даних, або для управління довжиною пакета даних.

15 Ключ: Приховане цифрове значення, яке генерується шляхом введення стану, наприклад, часу, в генератор ключів, який генерує ключ по секретному алгоритму. Ключ використовується для вибору алгоритму для шифрування даних в пакеті з селектора. Ключ можна використовувати для безпечної передачі інформації про стан з публічних або незахищених ліній.

20 Сервер обміну ключами: Комп'ютерний сервер, часто належить сторонній організації і не залежить від оператора мережі SDNP, який розподіляє відкриті ключі для клієнтів і, при необхідності, для серверів, використовуючи симетричне шифрування, особливо при роботі з ключами, керованими клієнтом, тобто, на основі наскрізного шифрування клієнтом, щоб запобігти можливості шпигунства з боку оператора мережі.

25 Остання ланка: Мережеве з'єднання між пристроєм клієнта та першим пристроєм в мережі, з яким він обмінюється інформацією, як правило, це вишка радіозв'язку, маршрутизатор Wi-Fi, кабельний модем, телевізійна приставка або з'єднання Ethernet.

Остання миля: Мережеве з'єднання між шлюзом SDNP і клієнтом, включаючи останню ланку.

30 Змішування: Об'єднання даних з різних джерел та типів даних для створення одного довгого пакета даних (або ряду менших субпакетів) з нерозпізнаваним вмістом. У деяких випадках раніше розділені пакети даних змішуються для відновлення справжнього вмісту даних. Операція змішування може також включати в себе вставку та видалення "сміттєвих" даних та синтаксичний аналіз.

35 Синтаксичний аналіз: Числова операція, при якій пакет даних розбивається на більш короткі субпакети для зберігання або для передачі.

Скремблювання: Операція, в якій порядок або послідовність сегментів даних в пакеті даних змінюється по відношенню до його звичайного порядку і набуває нерозпізнаваної форми.

40 Розділення: Операція, в якій пакет даних (або кілька послідовних пакетів даних) розбивається на кілька субпакетів, які направляються декільком адресатам. Операція розділення також може включати в себе вставку та видалення "сміттєвих" даних.

Програмний комутатор: Програмне забезпечення, що містить виконуваний код, який виконує функції комутатора і маршрутизатора.

45 SDNP: Аббревіатура (англ. Secure Dynamic Network And Protocol - Динамічна захищена комунікаційна мережа та протокол), що означає гіперзахищену мережу зв'язку, виконану відповідно до даного винаходу.

Сервер адміністрування SDNP: Комп'ютерний сервер, який розподіляє виконуваний код та спільні секрети по серверам SDNP в глобальному просторі або в певних зонах.

Мостовий вузол SDNP: Вузол SDNP, що з'єднує одну хмара SDNP з іншою, що мають різні зони і облікові дані безпеки.

50 Клієнт або клієнтський пристрій SDNP: Мережевий пристрій, зазвичай - стільниковий телефон, планшет, ноутбук, стаціонарний комп'ютер або пристрій IoT, що працює з додатком SDNP для підключення до хмари SDNP, зазвичай підключається на останній милі цієї мережі.

Хмара SDNP: Мережа взаємозалежних серверів SDNP, що запускають виконуваний код програмного комутатора для виконання операцій вузла зв'язку SDNP.

55 Шлюзовий вузол SDNP: Вузол SDNP, що з'єднує хмару SDNP з останньою милею SDNP і клієнтом. Шлюзові вузли SDNP вимагають доступу, щонайменше, до двох зон - до хмари SDNP і останньої милі.

60 Медіа-вузол SDNP: Виконавчий код програмного комутатора, який обробляє вхідні пакети даних з певними ідентифікаційними мітками відповідно до інструкцій сигнального сервера або іншого комп'ютера, що виконує сигнальну функцію, в тому числі шифрування/дешифрування,

скремблювання/дескремблювання, змішування/розділення, тегів розмітки і генерування заголовків та підзаголовків SDNP. Медіа-вузол SDNP відповідає за ідентифікацію вхідних пакетів даних, що мають певні мітки, та за пересилку новостворених пакетів даних до наступного адресату.

5 Медіа-сервер SDNP: Комп'ютерний сервер, на якому розміщується програмний комутатор, який виконує функції медіа-вузла SDNP при двоканальному та триканальному зв'язках, а також виконує завдання вузла сигналізації SDNP та вузла сервера імен SDNP при одноканальному зв'язку.

Сервер імен SDNP: Комп'ютерний сервер, на якому розміщений програмний комутатор, який виконує функції вузла сервера імен SDNP при триканальному зв'язку.

10 Вузол сервера імен SDNP: Виконавчий код програмного комутатора, який управляє динамічним списком кожного пристрою SDNP, підключеного до хмари SDNP.

Мережа SDNP: Вся гіперзахищена мережа зв'язку, яка поширюється від клієнта до клієнта, в тому числі що охоплює ділянки зв'язку останньої ланки і останньої милі, а також хмара SDNP.

15 Вузол SDNP: вузол зв'язку SDNP, що містить програмний комутатор, який запускається на комп'ютерному сервері або на апаратному пристрої, підключеному до мережі SDNP, що функціонує як вузол SDNP - медіа-вузол, сигнальний вузол або вузол сервера імен.

Сервер SDNP: Комп'ютерний сервер, який представляє собою або медіа-сервер SDNP, або сигнальний сервер SDNP, або сервер імен SDNP і реалізує відповідні функції програмного комутатора для роботи в якості вузла SDNP.

20 Сигнальний вузол SDNP: Виконавчий код програмного комутатора, який ініціює виклик або зв'язок між сторонами, визначає всю або частину з декількох маршрутів для передачі фрагментованих даних на основі критеріїв абонента і динамічну таблицю затримок поширення між вузлами, а також формує команди для медіа-вузла SDNP, як управляти вхідними та вихідними пакетами даних.

25 Сигнальний сервер SDNP: Комп'ютерний сервер, на якому розміщується програмний комутатор, який виконує функції вузла сигналізації SDNP в двоканальній та триканальній системах зв'язку SDNP, та також виконує функції вузла сервера імен SDNP в двоканальних системах зв'язку.

30 Параметри безпеки: Цифрові значення, наприклад, початкові стани і ключі, які генеруються генераторами початкових станів або генераторами ключів по секретним алгоритмам в поєднанні з постійно змінюваним вхідним станом, наприклад, мережевим часом, і які тому можна безпечно передавати по загальнодоступних або незахищених лініях зв'язку.

35 Початковий стан: Приховане цифрове значення, яке генерується шляхом введення стану, наприклад, часу, в генератор початкових станів, який генерує початковий стан за секретним алгоритмом. Початковий стан використовується для вибору алгоритму для скремблювання або розділення даних в пакеті з селектора. Початковий стан можна використовувати для безпечної передачі інформації про стан по загальнодоступним або незахищеним лініям зв'язку.

40 Селектор: Список або таблиця можливих алгоритмів скремблювання, шифрування або розділення, які є частиною спільних секретів і які використовуються разом з початковим станом або ключем для вибору конкретного алгоритму скремблювання, дескремблювання, шифрування, дешифрування, розділення або змішування пакета або пакетів.

45 Спільні секрети: Конфіденційна інформація про роботу вузлів SDNP, що включає таблиці або селектори алгоритмів скремблювання/дескремблювання, шифрування/дешифрування та змішування/розділення, а також алгоритми, що використовуються генераторами початкових станів, генераторами ключів, інформація про зони та процеси перетасовки алгоритмів, локально зберігаються на серверах DMZ, недоступних по мережі SDNP або через Інтернет.

50 Стан: Вхідні дані, наприклад, місце розташування, зона або мережевий час, які використовуються для динамічного генерування параметрів безпеки, наприклад, початкових станів або ключів, або для вибору алгоритмів для конкретних операцій SDNP, наприклад, змішування, розділення, скремблювання і шифрування.

Час: Універсальний мережевий час, що використовується для синхронізації зв'язку в мережі SDNP.

55 Дескремблювання: Процес, який використовується для відновлення сегментів даних скремблюваного пакету даних до їх первісного порядку або послідовності. Дескремблювання - це функція, зворотна по відношенню до функції скремблювання.

Зона: Мережа певних взаємопов'язаних серверів, які спільно використовують загальні облікові дані безпеки і спільні секрети. З'єднання останньої милі представляють собою окремі зони від тих, які знаходяться в хмарі SDNP.

60 Розробка динамічної захищеної комунікаційної мережі та протоколу (SDNP)

Для запобігання кібератак та злому системи зв'язку з комутацією пакетів при мінімальній затримці пакетів в режимі реального часу, забезпечення можливості встановлення стійкого з'єднання і максимальної безперервності голосового зв'язку та онлайн-відео, розроблена розглянута в цьому документі динамічна захищена комунікаційна мережу та протокол (SDNP), заснована на ряді керівних принципів, а саме:

- Передача даних в режимі реального часу повинна завжди відбуватися по шляху найменшої затримки.

- При несанкціонованому стеженні або аналізі пакета даних має бути недоступним вміст, відносно того, звідки надійшов пакет, куди він прямує і що в ньому знаходиться.

- Корисне навантаження пакета даних має бути динамічно перешифроване, тобто дешифровано, а потім знову зашифровано з використанням іншого алгоритму шифрування, виключаючи ризик злому в будь-який прийнятний момент часу.

- Навіть після дешифрування, все корисне навантаження пакета даних як і раніше містить незрозумілу інформацію, що представляє собою динамічно скрембльоване поєднання кількох розмов і незв'язаних даних, змішаних зі "сміттєвими" заповнювачами пакета.

Впровадження вищезазначених керівних принципів охоплює безліч унікальних та інноваційних методів, функцій, властивостей і реалізацій, включаючи всі або деякі із наступних варіантів втілення винаходу:

- SDNP використовує одну або кілька виділених хмар, що містять телефонну систему, функції програмного комутатора, реалізовані з використанням пропрієтарного програмного забезпечення управління і контролю, недоступного через Інтернет.

- Передача даних всередині хмари повністю відбувається з використанням виділеної системи маршрутизації пакетів SDNP в пропрієтарних хмарах на основі адрес SDNP і динамічних портів (тобто пропрієтарних адрес NAT), а не IP-адрес. Адреси SDNP не можуть використовуватися або маршрутизуватися через Інтернет або за межами хмари SDNP.

- Мережа SDNP постійно ідентифікує та здійснює динамічну маршрутизацію всієї передачі даних в режимі реального часу за наявними шляхами з найменшою затримкою.

- Ніяка захищена або виконувана в режимі реального часу передача даних не маршрутизується поза хмарию SDNP або через Інтернет, крім передачі даних між хмарами і на ділянці останньої милі, і крім того зазвичай використовує маршрутизацію для одного переходу з невидимими адресами.

- Дані маршрутизації, що містяться в пакеті даних, ідентифікують маршрут для одного переходу між двома сусідніми пристроями, визначаючи тільки адреси SDNP або IP-адреси попереднього і наступного сервера

- Номер телефону або IP-адреси абонента і одержувача виклику, тобто відповідні адреси відправника і одержувача клієнта не присутні ні в заголовках IP-пакетів, ні в зашифрованому корисному навантаженні.

- Спільні секрети, пов'язані з управлінням і контролем, існують в системному програмному забезпеченні, встановленому на захищених серверах DMZ, недоступних через Інтернет.

- Передача пакетів SDNP може здійснюватися по трьох незалежних каналах, якими є "сервер імен", який використовується для ідентифікації елементів в хмарі SDNP; "Медіа-сервери", що використовуються для маршрутизації вмісту і даних, і "сигнальні сервери", що використовуються для управління і контролю пакетів та викликів.

- Інформація про маршрут разом з ключами та числовими початковими станами (при необхідності) надається усім учасникам в обміні медіа-серверів по окремому сигнальному каналу до виклику або встановлення зв'язку і без вмісту. Сигнальний сервер передає медіа-серверам тільки дані про попередній і наступний адресатів пакета, що проходить через мережу.

- Медіа-пакети містять фрагментовані дані, що представляють тільки частину виклику, документа, тексту або файлу, динамічно змішані та повторно змішані з іншими пакетами, що містять фрагментовані дані з інших джерел і різного типу.

- Для захисту передачі даних на ділянках першої і останньої милі використовуються спеціальні методи захисту, в тому числі розділення повідомлень, пов'язаних з сигнальним сервером, і медіа-пакетів та пов'язаних з ними пакетів вмісту.

- Передача пакетів залежить від типу вмісту: голосові повідомлення та відео реального часу або онлайн-відео передаються на основі поліпшеного протоколу UDP, в той час як сигнальні пакети, пакети команд і управління, файли даних, файли додатків, системні файли і інші файли, які чутливі до втрати пакетів або затримки, використовують протокол TCP.

- Для підтвердження того, що пристрій є реальним клієнтом, а не клоном, і для перевірки автентичності того, що людина, яка бере участь у комунікації - є справжнім власником

пристрою, а не самозванцем, використовуються спеціальні методи захисту та перевірки автентичності.

Щоб забезпечити захищену передачу даних з малою затримкою та високою якістю обслуговування в технології VoIP і додатках реального часу, "динамічна захищена комунікаційна мережа та протокол" (SDNP), що розглядається в цьому документі, використовує мережу "динамічної решітки", що включає:

- динамічну адаптивну багатомаршрутну і решітчасту маршрутизації з мінімальною затримкою;
- динамічне скремблювання пакетів;
- динамічну фрагментацію з використанням розділення, змішування, синтаксичного аналізу пакетів, а також заповнювачі пакетів "сміттєвими" бітами;
- динамічне внутрішньовузлове шифрування корисного навантаження в мережі або хмарі;
- динамічний мережевий протокол з маскуванням адрес і необхідною інформацією маршрутизації;
- багатоканальний зв'язок, що відокремлює носій та вміст від сигнальної, командних і контрольних, та мережевих адрес;
- динамічний адаптивний транспортний протокол реального часу з функціями для конкретних типів даних і контекстною маршрутизацією;
- підтримку шифрування корисних навантажень клієнтом з управлінням ключами під керівництвом користувача;
- полегшений аудіокодек для підвищення якості обслуговування в перевантажених мережах.

Як вже було зазначено, зв'язок в мережі SDNP заснований на мультимаршрутній та решітчастій передачі для динамічної маршрутизації пакетів даних. На відміну від одноканального зв'язку з комутацією пакетів, використовуваних для передачі даних ОТТ в Інтернет та технології VoIP, в системі SDNP, відповідно до даного винаходу, вміст пакетів даних не передається послідовно цілісними пакетами, що містять інформацію з загального джерела або від абонента, а передається у фрагментованій формі з динамічним змішуванням та повторним змішуванням вмісту, що виходить від кількох джерел і абонентів, при якому зазначені дані об'єднують неповні уривки даних, вмісту, голосу, відео і файлів з даними різного типу, об'єднаних з заповнювачами із "сміттєвих" даних. Перевага розглядуваної реалізації фрагментації та передачі даних полягає в тому, що навіть незашифровані і нескрембльовані пакети даних майже неможливо інтерпретувати, оскільки вони являють собою комбінацію незв'язаних даних та типів даних.

Завдяки поєднанню фрагментованого змішування та розділення пакетів зі скремблюванням і динамічним шифруванням пакетів, ці гібридні пакети динамічно зашифрованих, скрембльованих фрагментованих даних являють собою беззмістовні пакети інформації, абсолютно незрозумілі для будь-якої сторони або спостерігача, що не мають спільних секретів, ключів, числових початкових станів, а також змінних стану та часу, які використовуються для створення, пакетування і динамічного повторного пакування цих даних.

Крім того, фрагментований вміст кожного пакета та секрети, які використовуються для його створення, діють всього лише на частку секунди, перш ніж пакет буде переконструйований новими фрагментами і новими заходами захисту, наприклад, зміненими початковими станами, ключами, алгоритмами та секретами. Обмежена тривалість часу, протягом якого у кіберпірата є доступ для злому і розкриття залежного від стану пакета даних SDNP, ще більше підвищує захищеність SDNP, при цьому є тільки одна десята секунди на те, щоб виконати обробку, що вимагає десятків тисяч років обчислень, що на дванадцять порядків величини більше часу, доступного для злому.

Комбінація вищезгаданих методів задіє багатовимірний захист, що значно перевершує захист при статичному шифруванні. З цієї причини розглядувана в цьому документі динамічна захищена комунікаційна мережа та протокол визначаються як "гіперзахищена" мережа.

Скремблювання пакетів даних. Відповідно до цього винаходу, що розглядається, захищений зв'язок в мережі з комутацією пакетів спирається на кілька елементів, що дозволяють запобігти злому та забезпечити безпеку, одним з яких є скремблювання пакетів SDNP. Скремблювання пакетів SDNP передбачає перегрупування сегментів даних з послідовності, що робить інформацію незрозумілою і непотрібною. Згідно рисунку 51А, дескрембльований пакет даних - пакет даних 923 - після виконання операції скремблювання 924, перетворюється в скрембльований пакет даних 925. Операція скремблювання може використовувати будь-який алгоритм, чисельний метод або метод управління послідовністю. Алгоритм може являти собою статичне рівняння або включати динамічні змінні або числові початкові стани на основі таких "станів", як час 920 при скремблюванні, та числовий початковий стан 929, згенерованих

генератором початкових станів 921, який може генерувати початковий стан 929, використовуючи алгоритм, який також залежить від такого стану, як час 920 при скремблюванні. Наприклад, якщо кожна дата перетворюється в унікальне монотонно зростаюче число, то кожне початкове значення 929 є унікальним. Час 920 та початковий стан 929 можуть

використовуватися для вибору конкретного алгоритму та також можуть використовуватися для вибору або обчислення конкретної операції скремблювання 924, обраної зі списку доступних методів скремблювання, тобто з алгоритмів скремблювання 922. На схемах потоків даних зручно ілюструвати цю операцію та послідовність скремблювання пакетів з використанням схематичного або символічного уявлення, позначеного тут символом 926.

Операція дескремблювання, показана на рисунку 51В, ілюструє функцію, зворотну операції скремблювання 924, а саме, операцію дескремблювання 927, де стан або час 920 та відповідний початковий стан 929, що використовуються для створення скремблованого пакета даних 925, знову використовуються для скасування операції скремблювання, щоб отримати дескрембловані дані, а саме, дескремблований пакет даних 923. Якщо при первинному скремблюванні пакетів використовувалося певний стан або час 920, той же метод скремблювання повинен бути знову використаний та в операції 927 дескремблювання шляхом його вибору зі списку алгоритмів скремблювання 922. Незважаючи на те, що список алгоритмів скремблювання 922 відноситься до терміну "скремблювання", та ж таблиця алгоритмів використовується для ідентифікації та вибору оберненої функції, необхідної для виконання "дескремблювання", тобто список алгоритмів скремблювання 922 містить інформацію, необхідну як для скремблювання пакетів даних, так і для дескремблювання пакетів даних. Оскільки ці дві функції включають в себе одні й ті ж інструкції, що виконуються в зворотному порядку, список 922 також може бути перейменований в список алгоритмів "скремблювання/дескремблювання" 922. Проте для більшої наочності ця таблиця відзначена тільки назвою функції без вказівки її зворотної функції.

Якщо алгоритм скремблювання, обраний для реалізації операції дескремблювання 927, не відповідає вихідному алгоритму, використаному при скремблюванні пакетів, або якщо початковий стан 929 або стан або час 920 не збігаються з часом скремблювання, то операція дескремблювання не зможе відновити вихідний дескремблований пакет даних 923, і дані пакета будуть втрачені. На схемах потоків даних зручно ілюструвати цей процес дескремблювання пакетів з використанням схематичного або символічного уявлення, позначеного тут символом 928.

Відповідно до винаходу, що розглядається, для виконання операції скремблювання можуть використовуватися різні алгоритми за умови, що процес є оборотним, а це означає, що повторення дій в зворотному порядку по відношенню до вихідного процесу повертає кожен сегмент даних в вихідне та правильне положення в заданому пакеті даних. З математичної точки зору допустимими алгоритмами скремблювання є ті алгоритми, які оборотні, тобто коли функція $F(A)$ має зворотну функцію $F^{-1}(A)$ або, як варіант, перетворення має відповідну зворотну функцію, для якої

$$F^{-1}[F(A)] = A$$

Це означає, що при застосуванні функції F до файлу даних, послідовності, рядку символів, файлу або вектору A та при подальшому застосуванні зворотної функції F^{-1} , будуть отримані вхідні дані A , що не змінилися ні за величиною, ні за порядком проходження елементів.

Приклади таких оборотних функцій ілюструються статичними алгоритмами скремблювання, показаними на рисунку 51С, в тому числі алгоритмами дзеркального відображення і фазового зсуву. В алгоритмах дзеркального відображення сегменти даних замінюються іншими сегментами даних, які є їх дзеркальним відображенням відносно осі симетрії, яка визначається модулем ("mod") процесу дзеркального відображення. При дзеркальному відображенні по модулю 2, згідно рисунку, кожен два сегмента даних вихідного пакета вхідних даних 930 міняються місцями, тобто міняються місцями 1А і 1В, аналогічно 1С і 1D, 1Е і 1F, і т.д., при цьому утворюється скремблований пакет вихідних даних 935 з осями симетрії між першим і другим сегментами даних, між третім та четвертим сегментами даних і т.д. або, на мові математики, в позиціях 1,5; 3,5; 5,5; ...; $(1,5+2n)$.

При дзеркальному відображенні по модулю 3 перший та третій сегменти даних в кожній трійці сегментів даних міняються місцями, а середній пакет кожної трійки залишається в початковому положенні. Відповідно, сегменти даних 1А та 1С міняються місцями, а 1В залишається в центрі трійки; сегменти даних 1D і 1F міняються місцями, а 1Е залишається в центрі трійки і т.д., при цьому утворюється скремблований пакет вихідних даних 936. При дзеркальному відображенні по mod3 осі симетрії знаходяться в позиціях 2, 5, 8,..., $(2+3n)$.

При дзеркальному відображенні по mod4 перший та четвертий, а також другий та третій сегменти даних з кожної четвірки сегментів даних міняються місцями і т.д., при цьому утворюється скрембльований пакет вихідних даних 937 з пакету вхідних даних 931. Відповідно, сегмент даних 1A міняється місцями з 1D; сегмент даних 1B міняється місцями з 1C; і т.д. При дзеркальному відображенні по mod4 вісь симетрії знаходиться між другим та третім сегментами даних кожної четвірки, тобто між 2-м і 3-м сегментами даних, 6-м і 7-м сегментами даних і т.д., або, на мові математики, в позиціях 2,5; 6,5; ...; $(2,5+4n)$. При дзеркальному відображенні по модулю m m -й сегмент даних вхідного пакета даних 932 замінюється першим, тобто 0-м сегментом даних; 0-й сегмент даних замінюється на m -й елемент; і аналогічним чином n -й елемент замінюється на $(m-n)$ -й сегмент даних, при цьому утворюється скрембльований пакет вихідних даних 938.

Інший спосіб скремблювання, також показаний на рисунку 51C, являє собою зсув кадру, де кожен сегмент даних зсувається вліво або вправо на один, два або більше кадрів. Наприклад, при фазовому зсуві на один кадр кожен сегмент даних зсувається на один кадр, при цьому перший сегмент даних зсувається в другу позицію; другий сегмент даних зсувається в позицію третього кадру і т.д., при цьому утворюється скрембльований пакет вихідних даних 940. Останній кадр вхідного пакета даних 930, кадр 1F в наведеному прикладі, зміщується в позицію першого кадру, раніше зайняту сегментом даних 1A.

При фазовому зсуві на 2 кадри перший сегмент даних 1A вхідного пакета даних 930 зсувається на два кадри в позицію, раніше зайняту сегментом даних 1C, четвертий кадр 1D зсувається в останню позицію скрембльованого пакета вихідних даних 941, передостанній сегмент даних 1E зсувається в першу позицію, а останній 1F зсувається в другу позицію. Аналогічним чином, при фазовому зсуві на 4 кадри сегменти даних вхідного пакета даних 930 зсуваються на чотири позиції, при цьому перший кадр 1A, замінює кадр, який раніше займав 1E, 1B замінює 1F, 1C замінює 1A, і т.д., при цьому утворюється скрембльований пакет вихідних даних 942. У разі максимального фазового зсуву перший кадр замінює останній; другий кадр, спочатку зайнятий сегментом 1B, стає першим кадром пакету вихідних даних 943, при цьому другий елемент зсувається в першу позицію, третій - в другу і т.д. Фазовий зсув одного кадру за межі максимального фазового зсуву призводить до того, що вихідні дані не відрізняються від вхідних. У наведених прикладах фазовий зсув даних проводиться вправо. Алгоритм також працює і при фазовому зсуві вліво, але з іншими результатами.

Вищезазначені алгоритми і аналогічні методи в даному описі називаються алгоритмами статичного скремблювання, оскільки операція скремблювання відбувається в один момент часу, та перетворює набір вхідних даних в унікальні вихідні дані. Крім того, описані раніше алгоритми не використовують значення пакета даних для визначення порядку виконання скремблювання. Згідно рисунку 51D, відповідно до винаходу, що розглядається, параметричне скремблювання означає, що метод скремблювання вибирається з таблиці можливих алгоритмів скремблювання, наприклад, sort #A, sort #B і т.д. на основі значення, отриманого з даних, що містяться в самому пакеті даних. Наприклад, припустимо, що кожен сегмент даних може бути перетворений в числове значення, засноване на обчисленні даних, що містяться в цьому сегменті даних. Одним з можливих підходів до визначення чисельного значення сегмента даних є використання десятикового або шістнадцятиричного еквівалента бітових даних в сегменті даних. Якщо сегмент даних містить кілька членів, числовий еквівалент можна знайти, підсумовуючи числа в сегменті даних. Потім сегмент даних об'єднується в одне число або "параметр", за яким потім вибирається метод скремблювання.

У показаному прикладі на етапі 950 проводиться параметричне перетворення нескрембльованого пакета даних 930 в таблицю даних 951, що містить числове значення для кожного сегмента даних. Згідно рисунку 51D, сегмент даних 1A (0-й кадр) має числове значення 23, сегмент даних 1B (1-й кадр) має числове значення 125 і т.д. Потім на етапі 952 для всього пакета даних 930 витягується одне значення пакета даних. У наведеному прикладі сума 953 являє собою результат лінійного змішування всіх значень сегментів даних з таблиці 951 і дорівнює 1002. На етапі 954 це параметричне значення, тобто сума 953, порівнюється з таблицею умов, наприклад, з наявним в програмному забезпеченні набором визначених операторів типу "if-then-else", щоб порівняти суму 953 з рядом неперекриваючих числових діапазонів в таблиці 955, щоб визначити, яку процедуру сортування слід використовувати. У цьому прикладі параметричне значення 1002 знаходиться в діапазоні від 1000 до 1499, а це означає, що слід використовувати процедуру сортування sort #C. Після вибору процедури сортування параметричне значення більше не потрібно. Після цього на етапі 956 вхідні нескрембльовані дані 930 скремблюються обраним методом, при цьому утворюється скрембльований пакет вихідних даних 959. У наведеному прикладі результат сортування Sort

#C, наведений в таблиці 957, містить набір відносних переміщень для кожного сегмента даних. Перший сегмент даних скрембльованого пакета даних 959 (0-й кадр) визначається переміщенням сегмента даних 1D на три позиції вліво, тобто його зрушенням три рази. Перший кадр містить сегмент даних 1B, він не змінює своє початкове положення, тобто переміщається на 0 місць. Другий кадр містить сегмент даних 1E, зрушений на дві позиції вліво щодо вихідного положення. Точно так же третій кадр містить сегмент даних 1F, зрушений на дві позиції вліво щодо вихідного положення. Четвертий кадр скрембльованого пакету вихідних даних 959 містить сегмент даних 1C, зрушений вправо, тобто на +2 позиції, щодо свого вихідного положення. П'ятий кадр містить сегмент даних 1A, зрушений на п'ять позицій вправо, тобто на +5 позицій, щодо свого вихідного положення.

Таким чином, згідно таблиці 957 для процедури сортування sort #C, кожен сегмент даних переміщається в однозначно певну нову позицію, при цьому утворюється параметрично певний скрембльований пакет даних 959. Щоб дескремблювати цей скрембльований пакет даних, процес виконується в зворотному порядку з використанням того ж методу сортування sort #C. Щоб гарантувати вибір того ж алгоритму для виконання операції дескремблювання, параметричне значення 953 пакета даних не може бути змінено в результаті операції скремблювання. Наприклад, лінійне змішування параметричного значення кожного сегмента даних дає одне й те саме числове значення незалежно від порядку чисел.

Динамічне скремблювання використовує стан системи, наприклад, час, щоб мати можливість ідентифікувати умови, при яких був скрембльований пакет даних, що дозволяє вибрати той же метод для виконання операції дескремблювання. В системі, показаної на рисунку 51B, стан використовується для створення замаскованого числового початкового стану, який передається відправнику або одержувачу пакета, який потім використовує цей початковий стан для вибору алгоритму скремблювання з таблиці. В якості альтернативи, відправнику або одержувачу може бути передано саме стан, і цей стан може використовуватися генератором прихованих чисел, розташованим у відправника або одержувача, для генерування прихованого числа, за яким здійснюється вибір алгоритму скремблювання/дескремблювання. Така компоновка показана на рисунку 51E, де стан, наприклад, час 920 використовується для генерування прихованого числа 961 за допомогою генератора прихованих чисел 960 і для вибору методу скремблювання зі списку алгоритмів скремблювання 962. Використовуючи прихований номер 961 для вибору алгоритму з таблиці 962 алгоритму скремблювання, операція скремблювання 963 перетворює нескрембльований пакет даних 930 в скрембльований пакет даних 964. Відповідно до рисунку 51E, стан 920 може бути передано генератору прихованих чисел 960 безпосередньо, або стан 920 може бути передано генератору прихованих чисел через генератор початкових станів 921.

Вибір алгоритму скремблювання по таємному номеру, а не просто по числовому початковому стану, має перевагу, яка полягає в тому, що він виключає будь-яку можливість злочинного відтворення таблиці скремблювання шляхом аналізу потоку даних, тобто статистичного зіставлення повторюваних наборів скрембльованих даних з відповідними числовими початковими станами. Початковий стан може бути видимим в потоці даних i , отже, вразливим для шпигунства, в той час як генератор прихованих чисел та приховане число HN, яке він створює, ґрунтуються на спільних секретах. Таким чином, приховане число HN не присутнє в потоці даних, невразливо для шпигунства та синтаксичного аналізу, що означає, що воно не передається по мережі, а генерується локально по числовому початковому стану. Таким чином, ця математична операція - генератор прихованих чисел - забезпечує підвищення рівня захищеності в частині запобігання хакерських атак, тому що призначення числового початкового стану замасковано.

Відразу після вибору алгоритму числовий початковий стан також може використовуватися як вхідна змінна в алгоритмі процесу скремблювання 963. Подвійне призначення числового початкового стану додатково заплутує аналіз, тому що початковий стан не безпосередньо обирає алгоритм, а працює разом з ним при визначенні остаточної послідовності сегментів скрембльованих даних. Аналогічним чином, щоб дескремблювати динамічно скрембльований пакет даних, початковий стан 929 (або, як варіант, стан або час 920) має бути передано з вузла зв'язку, пристрою або програмного забезпечення, що виконує скремблювання, до будь-якого вузла або пристрою, який буде його дескремблювати.

Відповідно до винаходу, що розглядається, алгоритм генерації початкових станів 921, генератор прихованих чисел 960 та список алгоритмів скремблювання 962 представляють собою "спільні секрети" - інформацію, що зберігається на сервері DMZ (як описано нижче) і невідому ні відправнику, ні одержувачу пакета даних. Спільний секрет встановлюється заздалегідь і не пов'язаний з переданими пакетами даних, крім, можливо, часу установки коду,

де використовуються різні процедури автентифікації для виключення витоку секрету. Як описано нижче, спільні секрети можуть бути обмежені "зонами", тому знання одного набору вкрадених секретів як і раніше не дозволяє хакеру отримати доступ до всієї мережі зв'язку або перехоплювати повідомлення в реальному часі.

Крім спільних секретів при динамічному скремблюванні, де алгоритм скремблювання змінюється під час передачі пакета даних, для скремблювання або дескремблювання даних потрібен початковий стан на основі "стану". Це стан, на якому засновано початковий стан, може бути будь-яким фізичним параметром, таким як час, номер вузла зв'язку, ідентифікатор мережі або навіть координати GPS, за умови, що не існує невизначеності щодо стану, використовуваного при генеруванні початкового стану, та за умови наявності засобів інформування наступного вузла про те, який стан використовувався при скремблюванні попереднього пакету даних. Алгоритм, який використовується генератором початкових станів для створення початкового стану, є частиною спільних секретів, і, отже, знання початкового стану не дозволяє визначити стан, на якому ґрунтується цей початковий стан. Початковий стан можна передавати від одного вузла зв'язку до іншого, помістивши його в сам пакет даних, відправивши його через інший канал або шлях або застосувавши будь-яку комбінацію цих методів. Наприклад, стан, що використовується при генеруванні початкового стану, може являти собою лічильник, який спочатку містить випадкове число, яке згодом збільшується на постійну величину при кожному проходженні пакета даних через вузол зв'язку, при цьому кожне значення лічильника відповідає конкретному алгоритму скремблювання.

В одному з варіантів здійснення динамічного скремблювання при першому виконанні операції скремблювання генерується випадкове число для вибору використовуваного методу скремблювання. Це випадкове число поміщається в пакет даних в його заголовку або тієї частини пакета даних, яка зарезервована для команд і управління, і не піддається скремблюванню. Коли пакет даних надходить в наступний вузол, це введенне число зчитується вузлом зв'язку та використовується програмним забезпеченням для вибору належного алгоритму для дескремблювання вхідного пакета даних. Потім це число, тобто "Значення лічильника", збільшується на одиницю або яке-небудь інше задане ціле число, пакет скремблюється відповідно до алгоритму, пов'язаним з цим новим числом, а це нове число записується в вихідний пакет даних замість попереднього числа. Наступний вузол зв'язку повторює цей процес.

В альтернативному варіанті здійснення даного методу на основі лічильника для вибору алгоритму скремблювання генерується випадкове число для вибору початкового алгоритму скремблювання, і це число надсилається кожному вузлу зв'язку, використовуваному для передачі конкретного пакета даних як "спільний секрет". Значення лічильника, наприклад, починаючи з 0, поміщається в пакет даних в його заголовку або в тій частині пакета даних, яка зарезервована для команд і управління, і не піддається скремблюванню. Потім цей пакет даних пересилається на наступний вузол зв'язку. Коли пакет надходить на наступний вузол зв'язку, сервер зчитує значення лічильника, додає це значення до початкового випадкового числа, ідентифікує алгоритм скремблювання, застосований до попередньої операції скремблювання пакета даних, і дескремблює пакет відповідно до нього. Потім значення лічильника збільшується на одиницю або будь-яке інше задане ціле число, і це значення лічильника знову зберігається в заголовку пакета даних або тієї частини пакета даних, яка зарезервована для команд і управління, і не піддається скремблюванню, перезаписуючи попереднє значення лічильника. Випадкове число, що служить спільним секретом, не передається в пакеті даних зв'язку. Після того як пакет даних надходить в наступний вузол зв'язку, сервер додає випадкове число, яке є спільним секретом, доданам до зміненого значення лічильника, витягнутої з пакета даних. Це нове число однозначно ідентифікує алгоритм скремблювання, використовуваний попереднім вузлом зв'язку для скремблювання вхідних пакетів. У разі застосування цього методу кіберпіратом може бути перехоплено тільки неінформативне значення лічильника з дескрембленої частини пакета даних, які не несуть ніякої важливої інформації.

В іншому альтернативному методі може використовуватися приховане число для передачі стану пакета та того, який алгоритм використовувався для його скремблювання. Приховане число об'єднує змінні в часі стан або початковий стан з спільним секретом, який, як правило, являє собою числовий алгоритм, при цьому вони разом використовуються для створення конфіденційного числа, тобто "прихованого числа", яке ніколи не передається між вузлами зв'язку і, отже, не є предметом розбору або дослідження при проведенні атаки "людина посередині" або з боку кіберпірата. Потім приховане число використовується для вибору застосовуваного алгоритму скремблювання. Оскільки стан або початковий стан не має сенсу без знання алгоритму, який використовується для обчислення прихованого числа, і оскільки

алгоритм, який є спільним секретом, може зберігатися за міжмереживим екраном, недоступним по мережі або Інтернету, жодний моніторинг мережевого трафіку не зможе виявити схему. Для додаткового ускладнення ситуації місцезнаходження початкових станів також може являти собою спільний секрет. В одному з варіантів здійснення винаходу число, яке передається до нескрембльованої частини пакета даних та доступне для спостереження та синтаксичного аналізу даних, наприклад, 27482567822552213, є довге число, в якому тільки його частина є початковим станом. Якщо, наприклад, початковий стан визначається цифрами з третьої по восьму, то реальний початковий стан - це не все число 27482567822552213, а тільки його частина, виділена жирним шрифтом, тобто початковий стан - це 48256. Потім цей початковий стан разом з спільним секретом використовується алгоритмом для генерування прихованого числа, а це приховане число, що використовується для вибору алгоритму скремблювання, динамічно змінюється по всій мережі.

Також відповідно до винаходу, що розглядається, ще одним можливим динамічним алгоритмом скремблювання є підмішування, навмисне введення передбачуваного шуму в потік даних при передачі. Один із можливих способів підмішування полягає в повторній перестановці двох сусідніх сегментів даних, що відбуваються при проходженні пакета по мережі. Згідно рисунку 51F, в момент часу t_0 , що відповідає динамічному стану 990, нескрембльований пакет даних 990 скремблюється за допомогою операції скремблювання пакетів 926, в результаті чого скрембльований пакет даних 1001 в момент часу t_1 відповідає динамічному стану 991. Пакет даних 1001, що входить у вузол зв'язку $N_{1,1}$, розміщений на сервері 971, являє собою послідовність сегментів даних, які прямують до порядку 1D, 1B, 1E, 1F, 1C, 1A. Пакет даних 1001 перетворюється вузлом зв'язку $N_{1,1}$ в момент часу t_2 , при цьому змінюючи порядок проходження сегментів даних шляхом обміну місць сегментів даних 1E і 1B. Результуючий пакет даних 1002, що містить послідовність сегментів даних 1D, 1E, 1B, 1F, 1C, 1A, потім обробляється вузлом зв'язку $N_{1,2}$, розміщеним на сервері 972, в момент часу t_3 , відновлюючи колишню послідовність 1D, 1B, 1E, 1F, 1C, 1A. У кожному черговому вузлі відносні позиції сегментів даних 1B і 1E міняють місцями, або підмішують, виключаючи утворення двох однакових послідовних пакетів. Таким чином, вихідна послідовність скремблювання містить пакети даних 1001, 1003, 1005 і 1007 на відповідні моменти часу t_1 , t_3 , t_5 і t_7 і змінені пакети даних 1002, 1004 і 1006 на відповідні моменти часу t_2 , t_4 і t_6 . Потім пакет даних 1007, що виходить з вузла зв'язку $N_{1,6}$, розміщеного на сервері 972, дескремблюється за допомогою операції дескремблювання пакета 928 для відновлення початкової послідовності даних 930 в момент часу t_8 .

Приклад статичного скремблювання відповідно до розглядуваної динамічної захищеної комунікаційної мережі та протоколу, застосовуваної до пакету даних 930, що проходить по ланцюжку серверів зв'язку 1010-1015, показаний на рисунку 52, де вузол зв'язку $N_{0,0}$, розміщений на сервері 1010, виконує операцію скремблювання пакета 926, в результаті чого утворюється скрембльований пакет даних 1008. Потім скрембльований пакет 1008 проходить по мережі зв'язку з комутацією пакетів без будь-яких подальших змін в послідовності сегментів даних, після чого вузол зв'язку $N_{0,f}$, розміщений на сервері 1015, нарешті виконує операцію дескремблювання пакета 928, відновлюючи вихідну послідовність сегментів в пакеті даних. Ця форма передачі даних являє собою статичне скремблювання, оскільки пакет даних, спочатку скрембльований, не змінюється при проходженні по мережі до тих пір, поки не досягне останнього сервера.

Представлені дані, що проходять по мережі, хоча вони і скрембльовані, можна вважати "текстовим файлом", оскільки в пакетах даних присутні фактичні дані, тобто пакети не перетворені в зашифрований текст. Навпаки, в зашифрованому тексті рядок символів, який містить вихідні дані, незалежно від того, скрембльовані вони чи ні, перетворюється в серію беззмістовних символів за допомогою ключа шифрування, і не може бути знову перетворений в вихідну форму текстового файлу без ключа дешифрування. Роль шифрування в даній системі зв'язку на основі SDNP обговорюється нижче в розділі "Шифрування".

Щоб змінити послідовність пакетів даних під час передачі по мережі, потрібно "ре-скремблювати" пакети, згідно рисунку 53. Процес ре-скремблювання пакетів перетворює скрембльований пакет даних в його спочатку не скрембльований стан, а потім знову скремблює його відповідно до нового алгоритму скремблювання. Таким чином, використовуваний тут термін "ре-скремблювання" означає дескремблювання пакета даних і його подальше повторне скремблювання, як правило, за допомогою іншого алгоритму або методу скремблювання. Такий підхід дозволяє виключити ризик пошкодження даних, що виникає при скремблюванні раніше скрембльованого пакета та втрати контролю над стеженням за послідовністю, необхідною для відновлення вихідних даних. Згідно рисунку, скрембльований пакет даних 1008, спочатку

скремблюється за допомогою операції скремблювання пакетів 926, "ре-скремблюється", спочатку шляхом його дескремблювання за допомогою операції дескремблювання 928, використовуючи зворотну операцію алгоритму скремблювання, за яким було виконано скремблювання даних, а потім пакет даних скремблюється знову за допомогою операції скремблювання 926, але за іншим алгоритмом скремблювання, а не по тому, який використовувався в попередній операції скремблювання 926. Результуючий ре-скремблований пакет даних 1009 відрізняється від попереднього скремблованого пакета даних 1008. Операція ре-скремблювання 1017 являє собою послідовне виконання дескремблювання та подальшого скремблювання, та в цьому документі називається "US ре-скремблювання", де "US" - аббревіатура від англ. Unscrambling-Scrambling, що означає "дескремблювання-скремблювання". Щоб відновити вихідний пакет даних 930, при операції дескремблювання остаточного пакета 928 необхідно застосувати зворотну функцію того ж алгоритму, який використовувався для попереднього ре-скремблювання пакета даних.

Застосування US ре-скремблювання в мережі зв'язку з комутацією пакетів на основі SDNP у відповідності до даного винаходу ілюструє рисунок 54, де пакет даних 930, спочатку скремблований за допомогою операції скремблювання 926 на сервері 1011 послідовно перетворюється за допомогою операції US ре-скремблювання 1017 по мірі проходження пакету даних по мережі серверів зв'язку з комутацією пакетів 1012-1015. Остання операція дескремблювання 928 відбувається на сервері 1016 та відновлює вихідну послідовність сегментів пакета даних 930. Оскільки ре-скремблювання багаторазово повторюється та проводиться в різні моменти часу від t_0 до t_f , результуюча мережа являє собою динамічно скрембловану мережу зв'язку. В процесі роботи нескремблований пакет даних 930 скремблюється за допомогою операції скремблювання 926, реалізованої в вузлі зв'язку $N_{0,0}$, розміщеному на сервері 1011. Використовуючи операцію US ре-скремблювання 1017, реалізовану в вузлі зв'язку $N_{0,1}$, розміщеному на сервері 1012, пакет перетворюється в скремблований пакет даних 1008 в момент часу t_2 . Той же процес повторюється при кожному проходженні пакета даних через вузли зв'язку, що залишилися. Наприклад, в вузлі зв'язку $N_{0,2}$, розміщеному на сервері 1013, операція US ре-скремблювання 1017 перетворює ре-скремблований пакет даних 1008 в новий ре-скремблований пакет даних 1009.

Кожна операція ре-скремблювання 1017 спочатку скасовує попереднє скремблювання на основі попереднього стану пакета, що надходить у вузол зв'язку, наприклад, якщо пакет даних 1008 був скремблований для стану, відповідного часу t_2 , а потім знову проводиться скремблювання цього пакета для нового стану, відповідного часу t_3 , при цьому утворюється ре-скремблований пакет даних 1009. Як було описано раніше, стан, що використовується при визначенні виконаного скремблювання, може включати початковий стан, час або число, пов'язане з будь-яким фізичним параметром, наприклад, часом, номером вузла зв'язку, ідентифікатором мережі або навіть координатами GPS за умови відсутності невизначеності щодо того, як виконувалося попереднє скремблювання. Відповідно, де-скремблювання пакета даних, що входить у вузол зв'язку $N_{0,1}$, розміщеному на сервері 1012, залежить від стану попереднього сервера, який використовувався для скремблювання пакета даних, тобто стану вузла зв'язку $N_{0,0}$, розміщеного на сервері 1011; де-скремблювання пакета даних, що входить у вузол зв'язку $N_{0,2}$, розміщеного на сервері 1013, залежить від стану вузла зв'язку $N_{0,1}$, розміщеного на сервері 1012, під час скремблювання; де-скремблювання пакета даних, що входить у вузол зв'язку $N_{0,3}$, розміщений на сервері 1014, базується на стані вузла зв'язку $N_{0,2}$, розміщеного на сервері 1013 під час скремблювання і т.д. Останній вузол зв'язку в цій мережі, в даному випадку вузол зв'язку $N_{0,f}$, розміщений на сервері 1016, не виконує US ре-скремблювання, а замість цього виконує операцію де-скремблювання 928, щоб відновити первинну нескрембловану послідовність сегментів пакета даних 930.

Відповідно до винаходу, що розглядається, статичне і динамічне скремблювання даних позбавляє інтерпретації скремблованих даних будь-якого сенсу, перетворюючи звук в невпізнаний шум, перетворюючи текст в незв'язний набір символів, перетворюючи відео в "відео-сніг" і код скремблювання в такий, що є невідновлюваним. Само по собі скремблювання забезпечує високу ступінь безпеки. Однак в методі SDNP, розглянутому в цьому документі, скремблювання - це тільки один з елементів, призначених для того, щоб забезпечити і гарантувати захищений зв'язок, що виключає втручання хакерів, проведення кібератак, кіберпіратства та проведення атак типу "людина посередині".

Шифрування пакетів. Відповідно до винаходу, що розглядається, захищений зв'язок у мережі з комутацією пакетів спирається на кілька елементів, що дозволяють запобігти злому та забезпечити безпеку, одним з яких є шифрування SDNP. Як зазначено вище, слово "шифрування" має грецьке походження, що означає "приховувати, ховати, затемнювати", і

являє собою засіб перетворення нормальної інформації або даних, зазвичай званих "текстовим файлом", в "зашифрований текст", який має незрозумілий формат, який робить дані нечитабельними без секретних знань. У сучасному зв'язку ці секретні знання зазвичай пов'язані зі спільним використанням одного або декількох "ключів", які використовуються для шифрування і дешифрування даних. Зазвичай ці ключі містять псевдовипадкові числа, що генеруються за певним алгоритмом. Сьогодні написані численні статті та тексти, які обговорюють переваги і недоліки різних методів шифрування, наприклад: "Криптономікон" Ніла Стівенсона, © 1999 р.; "Книга шифрів: наука про секретності від Стародавнього Єгипту до квантової криптографії" Саймона Сінгха, © 1999 р.; "Практична криптографія" Нільса Фергюсона, © 2013 року і "Криптоаналіз: дослідження шифрів і їх розшифровка", вперше опублікована в 1939 р

Незважаючи на те, що поняття шифрування або шифрів з'явилося в давні часи і добре відомо фахівцям в даній області, застосування криптографії в розглядуваній динамічній захищеній комунікаційній мережі та протоколі є унікальним, воно полегшує як наскрізне шифрування, так і динамічне шифрування для одного міжвузлового переходу стосовно архітектурі самої мережі незалежно від власного шифрування даних будь-якого клієнта. Базовий принцип архітектурного проектування системи передачі даних SDNP полягає в тому, що при наявності достатнього часу будь-який статично зашифрований файл або повідомлення можна в кінцевому підсумку зламати та викрасти інформацію, незалежно від того, наскільки складний шифр. Незважаючи на те, що це припущення може не відповідати дійсності, немає необхідності доводити або спростовувати його, тому що протилежний підхід, тобто очікування збою конкретного методу шифрування, може привести до неприйнятних і незворотних наслідків.

Замість цього система передачі даних SDNP базується на передумові, що всі зашифровані файли мають обмежений "термін зберігання", це метафоричне висловлювання означає, що зашифровані дані є надійними (захищеними) тільки протягом обмеженого періоду часу, і що конфіденційні дані повинні бути заново динамічно перешифровані через регулярні інтервали часу, в ідеалі набагато менші, ніж найоптимістичніші оцінки часу, необхідного для злому шифру з використанням сучасних комп'ютерів. Наприклад, якщо, за оцінкою криптологів, велика серверна ферма криптодвигунів може зламати цей шифр за один рік, то в системі передачі даних SDNP пакет даних буде перешифровуватися через кожну секунду або навіть через кожні 100 мс - інтервал, величина якого на багато порядків менша інтервалу, протягом якого найкраща технологія здатна зламати його. Тому шифрування в SDNP обов'язково є динамічним, тобто залежить від часу, а також може бути залежним від простору, тобто залежати від місця розташування вузла зв'язку в мережі з комутацією пакетів або географічного положення. Таким чином, використовуваний тут термін "перешифрування" відноситься до розшифрування пакету даних і його подальшого повторного шифрування, як правило, за допомогою іншого алгоритму або методу шифрування.

Отже, шифрування в SDNP передбачає багаторазове та часте перетворення даних з незашифрованого текстового файлу в зашифрований текст, що робить інформацію незрозумілою і непотрібною. Навіть якщо шифрування даних конкретного пакета неймовірним чином вдалося зламати, то при використанні методів динамічного шифрування SDNP у наступного пакета даних буде зовсім інший ключ шифрування або шифр, що потребують починати з самого початку новий процес злому його шифрування. Обмеження загального вмісту кожного унікально зашифрованого пакета даних пом'якшує потенційний збиток від несанкціонованого доступу, оскільки відкритий пакет даних сам по собі містить занадто маленький файл даних і не представляє особливої цінності або користі для кіберпірата. Більш того, комбінуючи динамічне шифрування з вищезгаданими методами скремблювання SDNP, можна значно підвищити захищеність зв'язку. Навіть в незашифрованому вигляді перехоплений файл даних містить лише невеликий фрагмент даних, голосу або відео, який скремблювання перетворює в беззмстовну і незрозумілу послідовність сегментів даних.

У відповідності до даного винаходу шифрування SDNP є динамічним та залежним від стану. Згідно рисунку 55А, незашифрований пакет даних являє собою текстовий файл 930, оброблений за допомогою операції шифрування 1020 при цьому утворюється зашифрований пакет даних, що містить шифротекст 1024 або 1025. У разі шифротекста 1024 весь пакет даних текстового файлу 930 шифрується повністю, при цьому сегменти даних від 1А до 1F обробляються як один файл даних. У разі шифротекста 1025 кожен сегмент даних текстового файлу 930 від 1А до 1F шифрується окремо і незалежно, і не об'єднується з іншими сегментами даних. Перший сегмент даних 1А шифрується та поміщається у відповідний перший сегмент даних шифротекста, показаного для наочності рядком символів, що починається з 7\$ та представляє собою довгий рядок символів або цифр, які не показані. Аналогічно, другий

сегмент даних 1В текстового файлу шифрується та поміщається в другий сегмент даних шифротекста, що представляє собою довгий рядок символів, показаний для наочності, починаючи з *^*. Символи 7\$ і *^ показані, щоб продемонструвати початкові елементи беззмстовних рядків символів, цифр, і літер, що не обмежують і не містять ніяких конкретних даних про джерело текстового файлу або довжину рядків символів, які були зашифровані.

Операція шифрування 1020 може використовувати будь-який доступний алгоритм, криптографічний метод або метод шифрування. Незважаючи на те, що алгоритм може являти собою статичне рівняння, в одному з варіантів здійснення винаходу операція шифрування використовує динамічні змінні або "стан", наприклад, час шифрування 920, а також генератор шифрування 1021 для створення "Е-key"1022, який також може залежати від стану, наприклад, часу шифрування 920. Наприклад, дата та час шифрування можуть використовуватися як числове значення для генерації ключа шифрування, який не можна відтворити, навіть якщо був виявлений алгоритм шифрування. Час 920 або інші "стани" також можуть використовуватися для вибору конкретного алгоритму зі списку алгоритмів шифрування 1023, який є списком доступних алгоритмів шифрування. На діаграмах потоків даних зручно ілюструвати цю операцію та послідовність шифрування пакету з використанням схематичного або символічного уявлення, позначеного тут символом, показаним для операції шифрування 1026. У всіх матеріалах опису винаходу символом, що представляє захищені та зашифровані дані, може також служити замок. Замок з розташованим над ним циферблатом вказує на захищений механізм доставки, наприклад, на зашифровані файли, які, якщо вони не будуть прийняті протягом певного інтервалу часу або до певного моменту часу, саморуйнуються і губляться назавжди.

Операція дешифрування, показана на рисунку 55В, ілюструє обернену функцію операції шифрування 1020, а саме, операцію дешифрування 1031, де стан або час 920 і інші стани, які використовуються для створення шифротекста 1024, разом з ключем дешифрування ("D-key") 1030, генеруються генератором ключів D-key 1029, повторно використовуються для скасування шифрування, тобто розшифрування файлу, для отримання незашифрованих даних, що містять вихідний пакет даних 990 текстового файлу. Якщо при первинному шифруванні пакетів використовувався певний стан або час 920, та ж операція шифрування повинна бути знову використана та в операції дешифрування 1031 шляхом її вибору зі списку алгоритмів шифрування 1023. Незважаючи на те, що список алгоритмів шифрування 1023 відноситься до терміну "шифрування", ця ж таблиця алгоритмів використовується для ідентифікації та вибору оберненої функції, необхідної для виконання "дешифрування", тобто список алгоритмів шифрування 1023 містить інформацію, необхідну як для шифрування, так і для дешифрування пакетів даних. Оскільки ці дві функції включають в себе одні й ті ж інструкції, що виконуються в зворотному порядку, таблиця 1023 також може бути перейменована в таблицю алгоритмів "шифрування/дешифрування" 1023. Однак для більшої наочності ця таблиця відзначена тільки назвою функції без вказівки її зворотної функції.

Якщо алгоритм шифрування, обраний для реалізації операції дешифрування 1031, не відповідає зворотному алгоритму по відношенню до вихідного алгоритму, використаному в операції шифрування пакетів, або якщо стан або час 920 не збігаються з часом шифрування, або якщо D-key 1030 не пов'язаний відомим числовим співвідношенням з E-key 1022, використовуваним під час шифрування, то операція дешифрування 1031 не зможе відновити вихідні незашифровані дані 990, і дані пакета будуть втрачені. На схемах потоків даних цей процес дешифрування пакетів зручно ілюструвати за використанням схематичного або символічного уявлення, позначеного тут символом операції дешифрування 1032.

Як вже було зазначено в цьому описі, використання ключів шифрування і дешифрування в криптографії та загальні алгоритми шифрування, такі як симетричне шифрування, з відкритим ключем, шифрування за методом RSA (абревіатура від прізвищ Rivest, Shamir і Adleman) і шифрування за методом AES256 (англ. Advanced Encryption Standard - покращений стандарт шифрування) і ін., є питаннями, добре відомими фахівцям в даній області. Однак застосування таких відомих криптографічних методів в даній системі передачі даних SDNP не так вразливе до злому або дешифрування через приховану інформацію, спільні секрети, залежних від часу динамічних змінних та станів, які є унікальними для даної системи передачі даних SDNP.

Тому навіть в малоймовірному випадку, коли кіберпірат володіє достатньою обчислювальною потужністю, щоб в кінцевому підсумку зламати надійну систему шифрування, йому не вистачає певної інформації, включеної в мережу SDNP як неpubлічна або як спільні секрети, які необхідні для виконання операції дешифрування; крім того він повинен зламати систему шифрування за частку секунди, поки не зміниться шифр. Крім того, кожен пакет даних, що проходить по даній мережі SDNP, використовує власний метод шифрування з унікальними

ключами і динамічними станами. Необхідність одночасного отримання недостатньої інформації, динамічних станів і обмеженого інформаційного вмісту в будь-якому конкретному пакеті робить викрадення скільки-небудь важливих даних з будь-якого конкретного пакета даних як надто складною, так і невдячним завданням для кіберпірата.

Щоб перехопити весь документ, онлайн-відео або голосову бесіду для відновлення зв'язаної послідовності даних, кібератака повинна послідовно зламувати і розшифровувати не один, а тисячі послідовних пакетів SDNP. Вкрай серйозне завдання безперервного злому послідовності пакетів SDNP ще більш ускладнюється поєднанням динамічного шифрування з раніше описаними методами скремблювання пакетів даних. Згідно рисунку 56, створення зашифрованого скремблюваного пакета даних 1024 включає в себе послідовну комбінацію операції скремблювання 926 і операції шифрування 1026 для перетворення нескремблюваного пакета даних 990 текстового файлу спочатку в скремблюваний пакет даних 1008 текстового файлу, а потім в шифротекст 1024 скремблюваного пакета даних. Щоб відновити зашифрований скремблюваний пакет, зворотні функції повинні бути застосовані в зворотній послідовності - спочатку операція дешифрування 1032 для відновлення скремблюваного пакета даних 1035 текстового файлу, а потім операція дескремблювання 928 для відновлення нескремблюваного пакета даних 990 текстового файлу.

Згідно рисунку, скремблювання і шифрування - це додаткові технології для забезпечення захищеного зв'язку. Незашифровані скремблювані дані, що проходять по мережі, можна вважати "текстовим файлом", оскільки в пакетах даних присутні фактичні дані, тобто пакети не перетворені в шифротекст. Зашифровані пакети даних, або шифротекст, містять скремблювані або нескремблювані рядки символів, перетворені в беззмістовну серію безглузвих символів за допомогою ключа шифрування, і не можуть бути відновлені в вихідну форму текстового файлу без відповідного ключа дешифрування. Залежно від використовуваного алгоритму, ключі шифрування і дешифрування можуть являти собою один та той же ключ або різні ключі, пов'язані відомою математичною залежністю. Таким чином, скремблювання і шифрування представляють додаткові засоби для забезпечення захищеного зв'язку відповідно до винаходу, що розглядається, для системи передачі даних SDNP.

Ці два методи, скремблювання і шифрування, можуть розглядатися незалежно, навіть коли вони використовуються в поєднанні один з одним, за винятком того, що послідовність, яка використовується для відновлення вихідного пакета даних з зашифрованого скремблюваного пакета даних, повинна бути зворотною по відношенню до послідовності, що використовується для його створення. Наприклад, якщо пакет даних 990 спочатку був скремблюваний з використанням операції скремблювання 926, а потім зашифрований з використанням операції шифрування 1026, то для відновлення вихідного пакета даних зашифрований скремблюваний пакет даних 1024 спочатку повинен бути дешифрований з використанням операції дешифрування 1032, а потім дескремблюваний з використанням операції дескремблювання 928. З точки зору математики, якщо операція скремблювання F перетворює рядок бітів або символів в еквівалентну скремблювану версію, а операція дескремблювання F-1 скасовує це скремблювання, внаслідок чого

$$F-1 [F (A)] = A,$$

і, аналогічно, якщо операція шифрування G перетворює рядок текстового файлу в еквівалентний шифротекст, а операція дешифрування G-1 скасовує це шифрування, внаслідок чого

$$G-1 [G (A)] = A,$$

то в комбінованому варіанті при послідовному виконанні операцій скремблювання і шифрування з подальшим дешифруванням і дескремблюванням виходить вихідний аргумент A - нескремблюваний пакет даних текстового файлу. Відповідно,

$$F-1 \{G-1 [G (F (A))]\} = A$$

тому що дії виконуються у зворотній послідовності, зокрема, дешифрування [G-1] зашифрованого скремблюваного пакета [G (F (A))] відновлює скремблюваний пакет даних текстового файлу F (A). Подальша операція дескремблювання F-1 скремблюваного пакета текстового файлу F (A) відновлює вихідний пакет даних A.

При використанні лінійних методів ця послідовність оборотна. Наприклад, якщо пакет даних спочатку зашифрований, а потім скремблюваний, то для відновлення вихідного пакета даних скремблюваний шифротекст повинен бути спочатку дескремблюваний, а потім дешифрований. Відповідно,

$$G-1 \{F-1 [F (G (A))]\} = A$$

Зміна послідовності не допускається. Дешифрування пакета даних, який раніше був зашифрований, а потім скремблюваний, без попереднього дескремблювання не відновить вихідний пакет даних, тобто

$$F-1 \{G-1 [F (G (A))]\} \neq A$$

Аналогічно, дескремблювання пакета, який був скремблюваний, а потім зашифрований, також не зможе відновити вихідний пакет даних, тому що

$$G-1 \{F-1 [G (F (A))]\} \neq A$$

У підсумку, якщо пакет текстового файлу скремблюється перед шифруванням, він повинен бути дешифрований до дескремблювання; якщо пакет текстового файлу зашифрований перед скремблюванням, він повинен бути дескремблюваний до його дешифрування.

Хоча й зрозуміло, що скремблювання і шифрування можуть виконуватися в будь-якій послідовності, в одному з варіантів здійснення методів SDNP, у відповідності до даного винаходу, шифрування і дешифрування під час передачі по мережі відбуваються частіше, ніж скремблювання, і тому шифрування має відбуватися після скремблювання, а дешифрування - до дескремблювання, згідно з рисунком 56, а не навпаки. Для зручності комбінація операції скремблювання пакета 926 з подальшою операцією шифрування 1026 визначається як операція шифрування скремблюваного пакета 1041, а її зворотна операція - комбінація операції дешифрування 1032 з подальшою операцією дескремблювання пакета 928 - як операція дескремблювання дешифрованого пакета 1042. Ці гібридні операції можуть використовуватися в статичній та динамічній системі передачі даних SDNP відповідно до даного винаходу.

На рисунку 57, на якому представлена система передачі даних SDNP, пакет текстового файлу 990 проходить через ряд вузлів зв'язку 1011-1016 в мережі зв'язку з комутацією пакетів в статично зашифрований та скремблюваний формі і являє собою пакет даних шифротекста 1040, який не змінюється від вузла до вузла та з часом. Згідно рисунку, на першому сервері - вузлі зв'язку $N_{0,0}$ 1101 - виконується операція шифрування-скремблювання 1041 для перетворення вихідного пакета даних 990 текстового файлу в пакет даних 1040 шифротекста - зашифрованих скремблюваних даних. Після перетворення в момент часу t_1 та відповідного стану 991 зашифрований скремблюваний пакет даних залишається статичним і незмінним, оскільки пакет даних проходить по мережі до тих пір, поки він не досягне вузла зв'язку $N_{0,f}$ 1016, де виконується зворотне перетворення пакета даних в його первісну форму - пакет даних текстового файлу 990 за допомогою операції дешифрування-дескремблювання 1042 в момент часу t_2 . Незважаючи на те, що комбінація скремблювання і шифрування значно підвищує захищеність, вона не є динамічно захищеною, оскільки пакети даних залишаються незмінними з плином часу та в міру проходження по мережі.

Один із засобів підвищення безпеки в будь-якій реалізації з використанням статичного скремблювання шифрування полягає в тому, що для кожного пакета даних, що відправляється, застосовуються різні методи скремблювання та/або шифрування, що включають зміну стану, початкових станів та/або ключів в момент часу t_1 , коли кожен пакет даних входить в мережу зв'язку.

Однак більш надійним є альтернативне рішення з динамічною зміною методу шифрування та/або скремблювання пакета даних у міру проходження пакету по мережі в часі. Щоб полегшити необхідну обробку даних для реалізації чисто динамічної версії системи зв'язку SDNP, необхідно об'єднати раніше певні процеси, щоб "ре-скремблювати" (тобто дескремблювати, а потім скремблювати) і "перешифрувати" (тобто дешифрувати, а потім зашифрувати) кожен пакет у міру його проходження через кожен вузол зв'язку в мережі зв'язку з комутацією пакетів. У цьому документі іноді використовується термін "перепакетувати" або "перепакетування" для комбінації "ре-скремблювання" і "перешифрування", незалежно від того, чи дешифрується пакет спочатку перед дескремблюванням або дескремблюється перед дешифруванням. У будь-якому випадку, операції дескремблювання і дешифрування в даному вузлі повинні виконуватися в порядку, зворотному виконанню операцій скремблювання і шифрування при виході пакета з попереднього вузла, тобто, якщо пакет був скремблюваний, а потім зашифрований в попередньому вузлі, то в поточному вузлі він повинен бути спочатку дешифрований, а потім дескремблюваний. Як правило, пакет після цього скремблюється, а потім зашифровується, коли залишає поточний вузол.

Операція "перепакетування" в вузлі зв'язку показана на рисунку 58, де пакет даних вхідного шифротекста 1040 спочатку дешифрується з допомогою операції дешифрування 1032, а потім дескремблюється за допомогою операції дескремблювання 928 для відновлення пакета нескремблюваного текстового файлу 990, що є вмістом вихідного пакету. Якщо будь-яку інформацію в пакеті необхідно перевірити, зробити розбивку, розділити або перенаправити, незашифрований файл текстового файлу є найкращим форматом для виконання таких

операцій. Потім пакет даних текстового файлу 990 знову скремблюється за допомогою операції скремблювання 926, після чого виконується нове шифрування за допомогою операції шифрування 1026 для створення нового скрембльованого пакета даних шифротекста 1043. Оскільки операція перепакетування вхідного скрембльованого пакета даних шифротекста 1040 здійснюється шляхом послідовного виконання дешифрування, дескремблювання, скремблювання і шифрування, для позначення цієї методики використовується аббревіатура DUSE (англ. Decryption - дешифрування, Unscrambling - дескремблювання, Scrambling - скремблювання, Encryption - шифрування). У динамічній захищеній комунікаційній мережі та протоколі бажано, щоб стан або час, ключ дешифрування і будь-які початкові стани, використовувані для виконання операції дешифрування 1032 і операції дескремблювання 928, відрізнялися від стану або часу, початкових станів або ключів шифрування, які використовуються для виконання операції скремблювання 926 і операції шифрування 1026.

Раніше розглянута операція перепакетування за методом DUSE 1045 може бути реалізована як програмне або апаратне забезпечення, або як вбудована програма в будь-якому вузлі зв'язку. Як правило, для виконання таких операцій вважають за краще використовувати програмне забезпечення, тому що з часом текст програми можна оновлювати або покращувати. Застосування операції перепакетування за методом DUSE 1045 в динамічній мережі показано на рисунку 59, де вузол зв'язку $N_{0,0}$, розміщений на сервері 1011 виконує операцію шифрування скрембльованого пакету 1041, вузол зв'язку $N_{0,f}$, розміщений на сервері 1016, виконує операцію дешифрування - дескремблювання 1042, а проміжні вузли зв'язку від $N_{0,1}$ до $N_{0,4}$, розміщені на серверах з 1012 по 1015, відповідно, виконують операції перепакетування за методом DUSE 1045. В процесі роботи пакет даних текстового файлу 990 спочатку обробляється за допомогою операції скремблювання шифрування 1041 в вузлі зв'язку $N_{0,0}$, а потім обробляється за допомогою операції перепакетування за методом DUSE 1045 в вузлі зв'язку $N_{0,1}$, при цьому утворюється перепакетований скрембльований текстовий файл 1008, що представляє собою пакет після дешифрування, дескремблювання пакета та скремблювання пакету, але ще до шифрування. Потім скрембльований текстовий файл 1008 шифрується, в результаті чого формується шифротекст 1040 в момент часу t_2 та відповідний стан 992. Процес знову повторюється в вузлі зв'язку $N_{0,2}$, а потім в вузлі зв'язку $N_{0,3}$, при цьому утворюється перепакетований скрембльований текстовий файл 1009, який зашифровується і формує шифротекст 1048 в момент часу t_4 та відповідний стан 994. Нарешті, вузол зв'язку $N_{0,f}$ виконує операцію дескремблювання-дешифрування 1042 для відновлення нескрембльованого текстового файлу 990 в момент часу t_f .

Змішування та розділення пакетів. Ще одним ключовим елементом динамічної захищеної комунікаційної мережі та протоколу, що розглядаються в цьому документі, є її здатність розділяти пакети даних на кілька субпакетів, направляти ці субпакети по декількох маршрутах, змішувати та возз'єднувати субпакети для відновлення повних пакетів даних. Процес розділення пакетів показаний на рисунку 60А, де пакет даних 1054 розділяється за допомогою операції розділення 1051 в поєднанні з операцією алгоритмічної розбивки 1052 та з операцією додавання "сміттєвих" даних 1053, яка має можливість вставляти або видаляти сегменти, що не є даними "сміттєвих" даних. Аналогічно надлишковій ДНК, яка присутня в геномі людини, "сміттєві" сегменти даних вставляються за допомогою операції додавання "сміттєвих" даних 1053, для розширення або управління довжиною пакета даних або, при необхідності, для їх видалення. Операція додавання "сміттєвих" даних 1053 особливо важлива, коли для заповнення пакета є недостатня кількість даних. Наявність сегментів "сміттєвих" даних, вставлених в пакет даних, також ускладнює для кіберпіратів можливість відділення реальних даних від шуму. У цьому документі "сміттєвий" сегмент даних - це пакет або сегмент даних, який цілком складається з неінформативних даних (бітів). Ці "сміттєві" біти можуть бути введені в потік пакетів даних, оточуючи реальні дані морем неінформативних бітів.

Мета операції розбивки 1052 полягає в тому, щоб розбити пакет даних 1054 на більш дрібні пакети даних, наприклад, субпакети даних 1055 і 1056 для обробки кожного зі складових компонентів. Розбиття пакета даних 1054 на більш дрібні частини дає такі унікальні переваги, як підтримка багатомаршрутної передачі, тобто передачі пакетів даних декількома різними шляхами, та полегшення унікального шифрування складових частин субпакетів з використанням різних методів шифрування.

Операція розділення може використовувати будь-який алгоритм, чисельний метод або метод розбивки. Алгоритм може являти собою статичне рівняння або включати динамічні змінні або числові початкові стани або такі "стани", як час 920 при першому формуванні вхідного пакета даних 1054 поруч субпакетів та числовий початковий стан 929, згенерований генератором початкових станів 921, який також може залежати від такого стану, як час 920 в

момент створення пакета даних. Наприклад, якщо кожна дата перетворюється в унікальне монотонно зростаюче число, то кожний початковий стан 929 є унікальним. Час 920 та початковий стан 929 можуть використовуватися для ідентифікації конкретного алгоритму, обраного зі списку доступних методів, тобто з алгоритму 1050. Розділення (не змішування) пакета є зворотна операція по відношенню до змішування з використанням того ж алгоритму, що виконується в прямо протилежній послідовності по відношенню до раніше використовуваної, яка була застосована для створення конкретного пакета. В кінцевому рахунку, все, що було зроблено, скасовується, але не обов'язково за один крок. Наприклад, скрембльований зашифрований пакет даних може бути розшифрований, але залишається скрембльованим. Оброблений за допомогою операції розділення 1051 нерозділений вхідний пакет даних 1054 перетвориться в кілька пакетів даних, наприклад, розділиться на пакети 1055 та 1056 постійної довжини за допомогою операції розбивки 1052, щоб алгоритмічно виконати операцію. На діаграмі потоків даних операцію розділення пакета 1051, що включає в себе розбивку 1052 і операцію додавання "сміттевих" даних 1053, зручно ілюструвати за використанням схематичного або символічного уявлення, позначеного тут символом операції розділення 1057.

Термін "розділення", що використовується в цьому документі, може включати в себе розбивку, яка полягає в розділенні пакета на два або більше пакета або субпакета, і крім того він може включати в себе вставку "сміттевих" пакетів або субпакетів, в пакети або субпакети, що утворилися при "розбивці", або видалення "сміттевих" пакетів або субпакетів з утворених при "розбивці" пакетів або субпакетів.

Зворотна функція - операція змішування пакетів 1060, показана на рисунку 60B, об'єднує кілька пакетів 1055 та 1056 в один і формує змішаний пакет 1054. Подібно розділенню пакетів, операція змішування пакетів може використовувати будь-який алгоритм, чисельний метод або метод змішування. Алгоритм може являти собою статичне рівняння або включати динамічні змінні або числові початкові стани або такі "стани", як час 920, які використовуються для задання умов, при яких змішуються пакети вхідних даних 1055 і 1056. Операція змішування, яка використовується для створення пакета даних, може використовувати числові початкові стани 929, які генеруються генератором початкових станів 921, які також можуть залежати від стану, наприклад, часу 920. Час 920 та початковий стан 929 можуть використовуватися для ідентифікації конкретного алгоритму змішування, обраного зі списку доступних методів змішування, тобто з алгоритмів змішування 1050. На схемах потоків даних цей процес змішування пакетів зручно ілюструвати за використанням схематичного або символічного уявлення, позначеного тут символом операції змішування 1061.

У відповідності до даного винаходу, змішування та розділення пакетів можуть використовувати будь-який з можливих алгоритмів. На рисунку 61A показані три з можливих способів змішування, а саме: конкатенація, з чергуванням і алгоритмічні методи. В процесі конкатенації послідовність сегментів пакета даних 1056 додається в кінець пакета даних 1055, при цьому утворюється змішаний пакет 1054. У процесі чергування сегменти пакетів даних 1055 і 1056 по черзі перемішуються, наприклад, як 1A, 2A, 1B, 2B і т.д., при цьому утворюється змішаний пакет даних 1065. Ще одним з методів, використовуваних для змішування пакетів, є алгоритм. У показаному прикладі алгоритм, який представляє собою віддзеркалену симетрію з чергуванням, змінює порядок проходження сегментів даних на 1A, 2A, 1B, 2B, 1C, 2C в першій половині змішаного пакета 1066 і на протилежний порядок в другій половині, тобто 2D, 1D, 2E, 1E, 2F, 1F.

Приклад застосування змішування пакетів з використанням конкатенації в відповідності до даного винаходу показаний на рисунку 61B. Згідно рисунку, в момент часу t_0 незмішані пакети даних 1055 і 1056 змішуються в вузлі зв'язку $N_{0,0}$, розміщеному на сервері 1011 з використанням операції змішування 1061. Потім об'єднаний пакет даних 1066, що утворився, містить послідовність сегментів 1A-1F, за якою слідує 2A-2F, передається по мережі серверів 1011-1016, при цьому він є незмінюваним текстовим файлом, статичний за своїм складом в усі моменти часу 998 до тих пір, поки в вузлі зв'язку $N_{0,t}$, розміщеному на сервері 1016, операція розділення пакетів 1057 не перетворить компоненти змішаного пакета даних 1066 в вихідні пакети даних 1055 і 1056.

Аналогічно, приклад застосування змішування з чергуванням в відповідності до даного винаходу показаний на рисунку 61C. Аналогічно послідовності з попереднього прикладу, утворюється змішаний пакет 1066 з послідовністю сегментів 1A, 1B, 2A, 2B, 3A, 3B Не дивлячись на те, що змішаний пакет відрізняється від показаного в прикладі з конкатенацією, операція розділення даних пакета 1057 здатна відновити вихідні незмішані пакети даних 1055 і 1056, оскільки знання алгоритму змішування та часу, стану або початкових станів, що

використовують в операції змішування, передається вузлу зв'язку $N_{0,f}$, розміщеному на сервері 1016, або в складі пакету даних 1066, або до передачі пакета в момент часу t_0 .

Змішування із скремблюванням - Розглянуті методи зв'язку з комутацією пакетів, що використовують розділення та змішування пакетів даних з різними комбінаціями сегментів даних, можуть, у відповідності з даним винаходом, об'єднуватися із скремблюванням пакетів різними способами. На рисунку 62A пакети нескремблованого текстового файлу 2155 і 1056 змішуються з використанням операції змішування 1061, при цьому утворюється змішаний пакет даних 1067, який в наведеному прикладі формується чергуванням сегментів текстового файлу. Після змішування пакет даних 1067 скремблюється за допомогою операції скремблювання 926, при цьому утворюється скремблований пакет даних текстового файлу 1068. Об'єднана послідовність результатів операції змішування пакетів 1061 та скремблювання пакетів 926 містить операцію змішування та скремблювання 1070, що представляє собою змішування з подальшим скремблюванням.

В альтернативній реалізації в відповідності до даного винаходу окремі пакети даних спочатку скремблюються, а потім змішуються, відповідно до рисунка 62B. У цій реалізації нескрембловані пакети даних текстового файлу 1055 та 1056 спочатку скремблюються окремо і незалежно операцією скремблювання 926, при цьому утворюються відповідні скрембловані пакети даних текстового файлу 1008 і 1009. Ці скрембловані пакети потім змішуються один з одним за допомогою операції змішування 1061, при цьому утворюється змішаний скремблований пакет даних 1069.

Спільне використання змішування та скремблювання, відповідно до цього винаходу, може бути інтегровано в статичні або динамічні мережі зв'язку SDNP. На рисунку 63 пакети даних текстового файлу 1055 і 1056 вводяться в вузол зв'язку $N_{0,0}$, що розміщений на сервері 1011 який виконує операцію змішування та скремблювання 1070, що представляє собою операцію змішування 1061, за якою слідує операція скремблювання 926, для формування змішаного скремблованого пакета 1068. Вміст пакету залишається постійним в усі моменти часу t_n , поки змішаний скремблований пакет 1068 проходить сервери 1011-1016. Потім кінцевий вузол зв'язку $N_{0,f}$, розміщений на сервері 1016, виконує операцію дескремблювання 928, за якою слідує операція розділення 1057, представлена як операція дескремблювання та розділення 1044.

На рисунку 64 показаний приклад динамічного змішування з скремблюванням в мережі зв'язку SDNP. Як і в попередньому прикладі статичної SDNP, пакети даних текстового файлу 1055 і 1056 вводяться в вузол зв'язку $N_{0,0}$, що розміщений на сервері 1011, який виконує операцію змішування та скремблювання 1070, що представляє собою змішування з подальшим скремблюванням. Над змішаним скремблованим пакетом виконується операція US ре-скремблювання 1010 на сервері 1012, при цьому формується змішаний скремблований пакет 1072 в момент часу t_2 , що відповідає стану 992. Потім сервери 1013 і 1014 виконують операцію US ре-скремблювання 1017, щоб повторно дескремблювати, а потім ре-скремблювати пакет даних. Операція US ре-скремблювання повторюється в вузлі зв'язку $N_{0,4}$, розміщеному на сервері 1015, в результаті чого утворюється знову ре-скремблований пакет даних 1073 в момент часу t_5 , що відповідає стану 995. Потім кінцевий вузол зв'язку $N_{0,f}$, розміщений на сервері 1016, виконує операцію дескремблювання-розділення 1044 для відновлення пакетів 1055 і 1056. У показаній динамічній реалізації мережі операція дескремблювання, яка використовується в кожній операції US ре-скремблювання 1017, використовує час або стан пакета даних, створеного на попередньому сервері, а потім ре-скремблюється пакет даних в поточний момент часу. Наприклад, пакет даних 1072, створений в момент часу t_2 на сервері 1012, ре-скремблюється на сервері 1013, тобто дескремблюється, використовуючи стан, пов'язаний з часом t_2 , а потім знову скремблюється з використанням стану, пов'язаного з поточним часом (не показано). Так, на наведеному на рисунку 64 прикладі показано, що в операції змішування та розділення можна вбудовувати повторювані та послідовні операції скремблювання і дескремблювання.

Змішування з скремблюванням і шифруванням. Розглянуті методи зв'язку з комутацією пакетів, що використовують розділення та змішування пакетів даних з різними комбінаціями субпакетів в поєднанні зі скремблюванням пакетів, можуть, у відповідності до винаходу, що розглядається, об'єднуватися з шифруванням. На рисунку 65 показані кілька прикладів функцій, які об'єднують змішування, скремблювання і шифрування, та відповідних їм зворотних функцій. Одним із прикладів є змішування-шифрування-скремблювання, або операція MSE (англ. Mixing-Scrambling-Encryption – змішування-шифрування-скремблювання) 1075, що представляє собою послідовність, що містить операцію змішування 1061, за якою слідує операція скремблювання 926 і, нарешті, операція шифрування 1026. Зворотна функція дешифрування-

дескремблювання-розділення, або операція DUS (англ. Decryption-Unscrambling-Splitting – дешифрування-дескремблювання-розділення) 1076 являє собою зворотну послідовність операцій, а саме: операцію дешифрування 1032, операцію дескремблювання 928 і операцію розділення 1057. Операції MSE 1075 та операції DUS 1076 передбачають наявність шифротексту. Щоб здійснювати передачу та відновити початковий вміст, хоча б по частинах, для його розшифрування повинні використовуватися ті ж спільні секрети, числові початкові стани і ключі шифрування/дешифрування, що використовувалися для створення пакета шифротекста.

Проміжні вузли можуть включати в себе тільки операцію перешифрування 1077, що представляє собою комбінацію операції дешифрування 1032 і операції шифрування 1026, або можуть включати в себе операцію DUSE 1045, що представляє собою послідовно виконувани операції дешифрування 1032, дескремблювання 928, скремблювання 926 і шифрування 1026. В операції перешифрування 1077 і операції DUSE 1045 функціям операції дешифрування 1032 і операції дескремблювання 928 можуть знадобитись початкові стани та/або ключі вузла зв'язку, які були відправлені в пакеті з попереднім часом та станом. Функції операції шифрування 1026 і операції ре-скремблювання 926 також можуть використовувати інформацію, початкові стани і ключі, згенеровані для поточного часу або стану, тобто для того часу, коли вузол зв'язку "оновлює" пакет даних. Оновлення пакетів даних ускладнює доступ до інформації в пакеті даних при проведенні кібератак, тому що дані пакетів знову заплутуються, а час, доступний для злому коду, скорочується.

Один із прикладів використання динамічного комбінаційного змішування, скремблювання і шифрування і їх зворотних функцій показаний на рисунку 66А, де два пакети даних 1055 і 1056 входять в вузол зв'язку $N_{0,0}$, розміщений на сервері 1011 в момент часу t_0 . Ці два пакети можуть представляти однакові типи даних, наприклад, два голосових пакета, два файли текстових повідомлень, два документа, дві частини програмного забезпечення і т.д. , або можуть представляти два різні типи інформації, наприклад, один мовленнєвий пакет і один текстовий файл, один текстовий пакет і одне відео- або фотозображення і т.д. Потім в момент часу t_1 , використовуючи інформацію стану 991 для генерування ключів, числових початкових станів або інших секретів, вузол зв'язку $N_{0,0}$, розміщений на сервері 1011, виконує операцію змішування-скремблювання-шифрування (MSE) 1075. В результаті утворюється скремблований пакет даних в форматі шифротекста, нерозбірливий і не інтерпретований для спостерігача, який не володіє інформацією про стан, що використовувався для його створення. Крім того, в момент часу t_1 генерується числове значення, відповідне часу або стану при змішуванні пакетів, та передається в кінцевий вузол $N_{0,f}$ або шляхом відправки цієї інформації перед відправкою змішаного пакета даних, або, навпаки, вкладенням цього початкового стану в заголовок цього ж пакета даних (розглядається нижче в цьому описі).

Потім дані передаються на вузол зв'язку $N_{0,1}$, розміщений на сервері 1012, що виконує операцію DUSE 1045, дешифруючи та дескремблюючи вхідні дані на основі інформації про стан 991, що відповідає часу t_1 , потім оновлює статус безпеки шляхом скремблювання та шифрування даних знову на основі інформації про стан 992, що відповідає часу t_2 . Якщо інформація про стан 991 передається в кінцевий вузол $N_{0,f}$ шляхом включення її до пакету даних або заголовку, тоді потрібні дві копії інформації про стан – одна для кінцевого вузла $N_{0,f}$, що містить стан 991, у разі змішування, і друга – про стан, використовуваний операцією DUSE, що змінюється щоразу, коли пакет даних переходить від одного вузла до іншого, тобто від станів 991 до 992, 993 і т.д. Використовуючи стан останньої операції, виконаної у вхідному пакеті даних, операція DUSE 1045 виконує повторне скремблювання незашифрованих даних, спочатку дешифруючи їх, потім виконуючи повторне скремблювання, а потім знову зашифровуючи їх, тобто операція повторного скремблювання вложена в операцію повторного шифрування. Підсумковий вихідний пакет даних містить зашифровані дані 1080В з незашифрованим базовим контентом представленим незашифрованим текстом 1080А. Операція DUSE 1045 послідовно повторюється на серверах 1013, 1014 і 1015, що приводить до створення зашифрованого тексту 1081В з незашифрованим базовим контентом представленим незашифрованим текстом 1081А у момент часу t_5 . Зв'язок завершується вузлом зв'язку $N_{0,f}$, розміщеним на сервері 1016, який виконує операцію 1076 дешифрування, дескремблювання та розділення (DUS), що дешифрує та дескремблює вхідний пакет даних на основі інформації про стан 995, що відповідає часу t_5 , використовуваному для останнього оновлення, потім розділяє пакет відповідно до стану 991 при першому змішуванні. Оскільки проміжні вузли не знають умов змішування, то навіть мережевий оператор з доступом до проміжних вузлів не знає умов, що використовуються при змішуванні. Підсумкові виведення 1055 і 1056 незашифрованого тексту в момент часу t_f відновлюють дані, відправлені по мережі, починаючи із часу t_0 . Оскільки

вміст пакета було повторно скрембльовано та повторно зашифровано в міру проходження пакетом через кожний вузол $N_{0,x}$, де $x=0, 1, 2, \dots, f$, можливість перехоплення та інтерпретації переданих пакетів даних надзвичайно ускладнена і дає мало часу для злому.

Простіший спосіб встановлення безпечного зв'язку передбачає змішування та скремблювання пакета на початку зв'язку, але використовує повторювані кроки повторного шифрування. На відміну від повністю динамічного зашифрованого прикладу скремблювання та змішування, представленого на попередній ілюстрації, на рис. 66В включено статичне змішування та скремблювання на сервері 1011 з динамічним шифруванням на серверах 1011-1015, означаючи при цьому, що тільки шифрування змінюється з часом. Зв'язок встановлюється в момент часу t_0 , починаючи з пакетів даних 1055 і 1056, що доставляються на вузол зв'язку $N_{0,0}$, розташований на сервері 1011. Як і в попередньому прикладі, два пакети можуть представляти будь-яке сполучення типів даних, включаючи голосові пакети, текстові повідомлення, документи, програмне забезпечення, відео або фотозображення тощо.

Потім у момент часу t_1 , використовуючи інформацію про стан 991 для генерації ключів, числових початкових значень або інших секретів, вузол зв'язку $N_{0,0}$ виконує операцію змішування, скремблювання та шифрування, або операцію MSE (Mixing-Scrambling-Encrypting) 1075. Отриманий зашифрований текст 1082В являє собою скрембльований пакет даних у форматі зашифрованого тексту, нерозбірливий і інтерпретований для будь-якого спостерігача, який не має інформації про стан, що використовується для його створення. Базовий пакет даних, що містить незашифрований текст 1082А, скремблюється і навіть без шифрування залишається незрозумілим для кіберпіратів, які намагаються відновити вихідні дані, текст, зображення або звук без інформації про стан, ключі, початкові значення і секрети.

Потім дані передаються на вузол зв'язку $N_{0,1}$, розміщений на сервері 1012, який замість виконання операції DUSE, як у попередньому прикладі, тільки повторно шифрує вхідні дані, тобто дешифрує дані на основі інформації про стан 991, що відповідає часу t_1 , потім зашифровує їх знову на підставі інформації про стан 992, що відповідає поточному часу t_2 . Процес, показаний як операція повторного шифрування 1077, призводить до створення пакета вихідних даних, що містить зашифрований текст 1083В, з базовим скрембльованим незашифрованим текстом 1083А, ідентичним попередньому незашифрованому тексту 1082А. Операція повторного шифрування 1077 послідовно повторюється на серверах 1013, 1014 і 1015, що призводить до створення нового зашифрованого тексту. Наприклад, зашифрований текст 1084В і базовий незмінний незашифрований текст 1084А представляють дані, що переміщуються між серверами 1013 і 1014. Основний незашифрований текст 1084А не змінюється з того моменту, як він був спочатку скрембльований операцією MSE 1075 у вузлі зв'язку $N_{0,0}$ у момент часу t_1 . Однак повторні шифрування у вузлах зв'язку $N_{0,1}$ і N_0 уже змінили зашифрований текст двічі, коли він вийшов з вузла зв'язку $N_{0,0}$.

Спільні секрети, які передаються та використовуються для виконання статичного змішування, скремблювання та динамічного шифрування, а також для зміни напрямку процесу, потребують два часи або стани – час t_1 і відповідний стан 991, який використовується для статичного змішування та скремблювання на сервері 1011 і є необхідним для дешифрування та розділення в кінцевій DUS операції 1076 на сервері 1016, та динамічний час і відповідний стан, використовуваний останнім вузлом зв'язку для виконання кожної операції 1077 повторного шифрування на серверах 1012-1015, а також стан, що змінюється динамічно і постійно, коли пакет даних проходить по мережі зв'язку з комутацією пакетів. На останньому етапі зв'язок завершується вузлом зв'язку $N_{0,f}$, розміщеним на сервері 1016, що виконує операцію DUS 1045, дешифруючи та виконуючи дескремблювання, і розділяючи (не змішуючи) вхідний пакет даних для відтворення виведень 1055 і 1056 незашифрованого тексту, ті самі дані, які були відправлені по мережі, починаючи з моменту часу t_0 .

Оскільки пакет зашифрований у вузлі $N_{0,0}$, повторно зашифрований при проходженні вузлів $N_{0,1} \dots N_{0,f-1}$ і дешифрований у вузлі $N_{0,f}$, хоча дані були змішані і скрембльовані тільки один раз, можливість перехоплення та інтерпретації переданих пакетів даних є надзвичайно важкою і дає мало часу для злому. До того ж, змішування декількох джерел даних, описаних раніше в цьому додатку, ще більше ускладнює спроби злому та кіберпіратства, тому що порушник не має поняття про те, які дані в ньому представлені, звідки вони прийшли або куди вони направляються – по суті, йому не вистачає деталей і контексту в характері пакета даних.

Іншим способом керування вмістом пакета даних під час його транспортування є "повернення в робочий стан" на кожному окремому кроці. У цьому способі, показаному на рис. 66С, за винятком вузлів шлюзу, кожен вузол виконує послідовну операцію – операцію DUS 1076, за якою слідує операція MSE 1075, фактично повністю перебудовуючи пакет даних для транспортування на кожному кроці. Відповідно до зображення, пакети вхідних даних 1055 і 1056

спочатку змішуються вузлом $N_{0,0}$ у момент часу t_1 , використовуючи стан 991, що призводить до створення зашифрованого тексту 1080Z, який відповідає незашифрованому тексту 1080Y. Потім зашифрований текст 1080Z відправляється на вузол $N_{0,1}$, де операція DUS 1076 ідентифікує вхідний пакет, який був створений з використанням стану 991, що відповідає моменту часу t_1 , і, як докладно показано на рис. 66D, послідовно дешифрує його, перетворюючи вхідний зашифрований текст 1080Z у незашифрований текст 1080Y. Потім незашифрований текст 1080Y дескремблюється та розділяється (не змішується), тим самим відновлюючи вихідні пакети даних 1055 і 1056.

При підготовці до наступного мережевого стрибка, два вихідних пакети даних знову змішуються та скремблюються, цього разу використовуючи алгоритми, обрані в момент часу t_2 , які відповідають стану 992, що призводить до створення незашифрованого тексту 1080A, який згодом зашифровується для одержання зашифрованого тексту 1080B, готового до відправлення на вузол $N_{0,1}$. Завдяки використанню цього методу, пакети вхідних даних повертаються у вихідний нормальний стан щоразу, коли вони потрапляють у вузол і виходять у зовсім новому "оновленому" стані, який відповідає поточному стану. У цьому методі кожен вузол повинен знати тільки стан вхідного пакета і не повинен знати про будь-які попередні стани, використовувані при транспортуванні даних.

Операції змішування та розділення. Процес змішування та розділення пакетів для об'єднання і розділення даних різних типів, показаних раніше на рис. 60A та рис. 60B, ілюструє пакети фіксованої довжини, підпорядковані принципу "збереження сегментів даних", де загальна довжина пакета довгих даних 1054 має таку саму кількість сегментів даних, що й сума пакетів коротших даних 1055 і 1056, створених з нього. Збереження сегментів даних означає, що при послідовних операціях змішування та розділення сегменти даних не створюються і не руйнуються. Цей простий принцип є проблематичним при комунікації, оскільки кількість даних у реальному часі може бути розрідженою і нездатною заповнити навіть один повний пакет.

У іншому крайньому випадку, коли мережа може бути надто перевантажена, сервер може бути нездатний прийняти довгий пакет без тривалих затримок поширення, що призведе до значної затримки. З цієї чи інших причин динамічне змішування та поділ пакетів даних, відповідно до даного винаходу, забезпечує засіб для керування, об'єднання та розділення пакетів даних різної довжини, контролюючи при цьому як довжину, так і кількість введень пакета даних, а також число та довжину виведень пакетів даних. Використання пакетів даних різної довжини, що містять контент, призначений для різних пунктів призначення, додатково збиває з пантелику хакерів, забезпечуючи при цьому ще один ступінь безпеки мережі. Як показано на рис. 67A, операція розбору (аналізу) 1087 і "сміттєва" операція 1088, призначена для вставки і видалення "сміттєвих" даних, використовуються спільно для керування та контролю довжиною пакета даних у змішаних пакетах даних, які застосовуються як для операцій змішування з одним виведенням, так і з множинними виведеннями.

На рис. 67A показано приклад змішування пакета з одним виведенням, в якому кілька введень різної довжини, у прикладі, показаному як пакети 4 сегментів даних 1090A та 1090C, і пакет 3 сегментів даних 1090B, змішуються з використанням операції змішування 1086 для одержання одного довгого пакета даних 1091. Операція змішування 1086 обирається з переліку алгоритмів змішування 1085 відповідно до поточного часу або стану 920, коли відбувається змішування, включаючи використання числового початкового значення 929, створеного генератором початкових значень 921. Під час операції змішування 1086 "сміттєва" операція 1088 вставляє сегменти "сміттєвих" даних у виведення пакета даних 1091 відповідно до обраного алгоритму.

Після змішування довгий пакет даних 1091 або альтернативні субпакети, що виникають у результаті операції розбору (аналізу) 1092, можуть або зберігатися локально, наприклад, очікуючи надходження інших пакетів даних, або можуть відправлятися на інші вузли мережі зв'язку. Перед відправленням на зберігання або маршрутизацією кожен пакет або субпакет "позначається" заголовком або підзаголовком, що ідентифікує цей пакет. Тег має важливе значення для розпізнавання вхідного пакета, тому він може оброблятися відповідно до інструкцій, отриманих раніше щодо того, як чинити з його даними, у тому числі, як змішувати, скремблювати, шифрувати або розділяти, дескремблювати і дешифрувати вміст пакета даних. Використання заголовків пакета даних і підзаголовків для ідентифікації та маркування пакетів даних докладніше описано нижче в цьому додатку.

Таким чином, крім введення в оману кіберзлочинців, роль аналізу, випадкової інформації та видалення випадкової інформації полягає в керуванні довжиною пакета даних. Наприклад, якщо остаточний довгий пакет даних 1091 є занадто довгим, то, відповідно до обраного алгоритму, операція аналізу (розбору) 1087 розбиває виведення довгого пакета даних 1091 на

коротші фрагменти. Довжина коротких фрагментів може задаватися обраним алгоритмом, наприклад, вирізати об'єднаний довгий пакет із регулярними інтервалами 1092 з "n" субпакетів. Бажана довжина пакета може визначатися апіорі або ґрунтуватися на мережевих умовах, наприклад, максимальна припустима довжина може розраховуватися на основі мережевих затримок. Наприклад, якщо затримка поширення Δt_{prop} між двома вузлами перевищує певне значення, тоді пакет даних буде аналізуватися з точки зору його зменшення, наприклад, довгий пакет даних 1091 розбивається з рівномірними проміжками за допомогою операції аналізу 1092 на "n" субпакети.

Незалежно від того, як аналізується довгий пакет, операція змішування з декількома виведеннями створює кілька виведень пакета даних, наприклад, пакети даних 1093A, 1093B і 1093C, як показано на рис. 67B. У показаному вище процесі "сміттєві" дані можуть вставлятися в субпакети для створення субпакетів контрольованої або фіксованої довжини. Кожен сегмент пакета даних або субпакета, наприклад, 1A, 1B, 1C і т.д., ідентифікується не за його значенням або змістом, а за його "слотовою" позицією в пакеті. Наприклад, довгий пакет даних 1091 містить 18 слотів даних із даними, присутніми в слотах 1, 4, 7, 8, 9, 11, 12, 13, 15 і 17, тоді як субпакет 1093A має довжину всього 6 слотів, що містять фактичний контент даних або аудіо в 1 і 4 слотах.

Для зручності операція змішування з декількома введеннями та одним виведенням (MISO) символічно представлена символом 1089, тоді як операція змішування з множинним введенням і множинним виведенням (MIMO) символічно представлена символом 1094, аналогічним більш ранньому, більш ідеалізованому прикладу, показаному на рис. 60A. Відповідно до винаходу, розглядуваному у цьому документі, змішування з множинним введенням та з одним виведенням 1089 корисне для безпечних з'єднань "останньої милі", тоді як змішування з множинним введенням і множинним виведенням 1094 корисне для реалізації багатоканальних і сітчастих мереж маршрутизації, описаних нижче в цьому додатку. У таксономії розглядуваних елементів і операцій мережі SDNP, операція змішування MISO 1089 може розглядатися як окремий випадок операції змішування MIMO 1094.

Зворотна функція для змішування множинного введення і одного виведення або MISO змішування являє собою поділ одного введення та множинного виведення або SIMO поділ. В одній модифікації, зображеній на рис. 67C, один довгий пакет даних 1091 ділиться за допомогою операції розділення 1100 на безліч субпакетів даних 1103A, 1103B і 1103C, які можуть містити субпакети фіксованої або змінної довжини. У наведеному прикладі субпакет 1103A містить 4 слоти даних, тоді як субпакети 1103B і 1103C містять тільки 3 слоти.

У другій модифікації, зображеній на рис. 67D, один довгий пакет даних 1091 ділиться за допомогою операції розділення 1105 на безліч субпакетів 1108A, 1108B і 1108C ідентичної фіксованої довжини з сегментами "сміттєвих" даних у якості наповнювача, коли непотрібні дані заповнюють весь пакет даних. В обох прикладах час або стан 920 і числове початкове значення 929, використовувані при створенні вхідних пакетів даних, необхідні для вибору алгоритму змішування з таблиці 1085 і для встановлення параметрів, необхідних для виконання операцій розділення 1100 і 1105. Хоча таблиця змішування алгоритму 1085 посилається на термін "змішування", ця ж таблиця алгоритмів використовується для ідентифікації та вибору зворотної функції, необхідної для виконання "розділення", тобто таблиця алгоритмів змішування 1085 містить інформацію, необхідну як для змішування пакетів даних, так і для розділення пакетів даних. Оскільки дві функції містять у собі ті самі кроки, які виконуються у зворотному порядку, таблиця 1085 також може бути перейменована в таблицю алгоритмів "змішування/розділення" 1085. Однак для ясності таблиця позначається тільки цією функцією, а не її зворотною функцією. Методи, використовувані для виконання змішування та розділення пакетів даних, є алгоритмічними і багато в чому схожими на алгоритми скремблювання, описані вище, за винятком того, що вони зазвичай містять у собі кілька пакетів вхідних або вихідних даних. Як виняток, операції змішування або розділення можуть виконуватися в одному пакеті даних - при вставленні або видаленні "сміттєвих" даних.

На рис. 67E показано один конкретний алгоритм змішування, що змішує три вхідні пакети даних 1090A, позначені як субпакет A, 1090B як субпакет B і 1090C як субпакет C, в один довгий пакет даних 1091, потім розбирає довгий пакет даних 1091 на три різні вихідні субпакети 1090D, позначені як субпакет D, 1090E як субпакет E і 1090F як субпакет F. Як представлено графічно, операція змішування 1094 перерозподіляє вміст даних зі слотів вхідних пакетів даних у довгий пакет, а також вставляє "сміттєві" дані в деякі проміжні слоти. Наприклад, третій слот субпакета 1090A, що містить сегмент даних 1C, переміщується в 11 слот довгого пакета даних 1091, 3-й слот субпакета 1090B, що містить сегмент даних 2F, переміщується в 17 слот довгого пакета даних 1091 і 2-й слот субпакета 1090C, що містить сегмент даних 3D, переміщується в 12 слот

довгого пакета даних 1091. Таким чином, повний алгоритм змішування містить підстановчу таблицю, як показано на прикладі нижче:

Слот довгого пакета №	Вхідний субпакет №	Слот вхідного субпакета №	Дані, включені у слот
Слот 1	Субпакет А	Слот 1	1А
Слот 2	Вставлені "сміттєві" дані		
Слот 3	Вставлені "сміттєві" дані		
Слот 4	Субпакет А	Слот 2	1В
Слот 5	Вставлені "сміттєві" дані		
Слот 6	Вставлені "сміттєві" дані		
Слот 7	Субпакет А	Слот 3	1С
Слот 8	Субпакет В	Слот 1	2С
Слот 9	Субпакет С	Слот 1	3С
Слот 10	Вставлені "сміттєві" дані		
Слот 11	Субпакет В	Слот 2	2D
Слот 12	Субпакет С	Слот 2	3D
Слот 13	Субпакет А	Слот 4	1Е
Слот 14	Вставлені "сміттєві" дані		
Слот 15	Субпакет С	Слот 3	3Е
Слот 16	Вставлені "сміттєві" дані		
Слот 17	Субпакет В	Слот 3	2F
Слот 18	Субпакет С	Слот 4	"Сміттєві" дані

5 Таким чином, у цілому, функція операції змішування полягає в тому, щоб визначити, в який слот у змішаному пакеті або довгому пакеті вставлені вхідні дані, а також визначити, які слоти змішаного пакета містять "сміттєві" дані.

10 Табличне подання алгоритму є ілюстративним для демонстрації того, що будь-який перерозподіл субпакетів вхідних даних у довгий пакет даних можливий. Як частина операції змішування 1094 наступною виконується операція розбору 1087, яка розбиває 1092 довгий пакет даних 1091 на три фрагменти рівної довжини для створення вихідних субпакетів 1093D, 1093E і 1093F, позначених відповідно як субпакет D, субпакет E та субпакет F.

15 На рис. 67F показано алгоритм, що виконує операцію розділення або "не змішування" 1101, що починається з трьох субпакетів рівної довжини 1093D, 1093E і 1093F, отриманих у результаті попередньої операції розбору 1087 і перерозподілу даних для створення нових субпакетів 1103A, 1103B і 1103C різної довжини, як описано в таблиці нижче. Мета операції розбору полягає в тому, щоб розбити довгий пакет на різні фрагменти меншого розміру або коротшої тривалості для локального зберігання або серіалізації даних для передачі.

Вхідний субпакет	Вхідний слот №	Вихідний розділений субпакет	Вихідний слот №	Дані, включені у слот
Субпакет D	Слот 1	Субпакет G	Слот 1	1A
	Слот 2	Вилучені "сміттєві" дані		
	Слот 3	Вилучені "сміттєві" дані		
	Слот 4	Субпакет G	Слот 2	1B
	Слот 5	Вилучені "сміттєві" дані		
	Слот 6	Вилучені "сміттєві" дані		
Субпакет E	Слот 1	Субпакет G	Слот 3	1C
	Слот 2	Субпакет H	Слот 1	2C
	Слот 3	Субпакет J	Слот 1	3C
	Слот 4	Вилучені "сміттєві" дані		
	Слот 5	Субпакет H	Слот 2	2D
	Слот 6	Субпакет J	Слот 2	3D
Субпакет F	Слот 1	Субпакет G	Слот 4	1E
	Слот 2	Вилучені "сміттєві" дані		
	Слот 3	Субпакет J	Слот 3	3E
	Слот 4	Вилучені "сміттєві" дані		
	Слот 5	Субпакет H	Слот 3	2F
	Слот 6	Вилучені "сміттєві" дані		

Відповідно до зображення, субпакет 1103A, позначений як субпакет G, містить 4 слоти, де слот 1 заповнюється сегментом даних 1A зі слоту 1 субпакета D, що відповідає слоту 1 довгого пакета 1091, слот 2 заповнюється сегментом даних 1B зі слоту 4 субпакета D, що відповідає слоту 4 довгого пакета 1091, слот 3 заповнюється сегментом даних 1C зі слоту 1 субпакета E, що відповідає слоту 7 довгого пакета 1091, а слот 4 заповнюється сегментом даних 1E зі слоту 1 субпакета E, що відповідає слоту 13 довгого пакета 1091. Аналогічно, субпакет 1103B, позначений як субпакет H, містить 3 слоти, в яких 1 містить сегмент даних 2C з 2 слоту субпакета E, другий містить сегмент даних 2D з 5 слоту субпакета E, а третій містить сегмент даних 2F з 5 слоту субпакета F. Субпакет 1103C також містить три слоти. У слоті 1 сегмент даних 3C надходить зі слоту 6 субпакета E. У слоті 2 сегмент даних 3D надходить зі слоту 6 субпакета E. У слоті 3 субпакета J сегмент даних 3E надходить зі слоту 3 субпакета F.

Таким чином такий алгоритм поділу визначає: (а) скільки буде розділених субпакетів; (б) скільки слотів буде в кожному розділеному субпакеті; (в) в який слот розділених субпакетів направлятимуться дані довгого пакета; (г) які слоти зі "сміттєвими" даними будуть видалятися; та (ґ) якщо вводяться нові слоти, що містять "сміттєві" дані, то можливо полегшити створення субпакета певної довжини. У разі операції розділення, яка слідує за операцією змішування, кількість субпакетів у розділених пакетах повинна дорівнювати кількості субпакетів у пакетах до їхнього змішування, якщо "сміттєві" дані не видаляються або не вставляються.

Ролі описаних операцій змішування та розділення, виконувани відповідно до цього винаходу, можуть адаптуватися для реалізації фрагментованого транспортування даних через будь-яку мережу за тієї умови, що усі вузли в мережі знають, яка послідовність операцій повинна виконуватися. У транспортуванні з одним маршрутом, наприклад, показаним раніше на рис. 61B, пакети даних 1055 і 1056 представляють різні розмови або комюніке від різних абонентів чи джерел. Після об'єднання довгий пакет даних або його розбиті версії готові для транспортування по мережі. Таку функцію можна вважати зв'язком із множинним введенням і одним виведенням або вузлом MISO.

Вихідні пакети даних відновлюються за допомогою зворотної функції, вузла зв'язку з одним введенням і множинним виведенням або вузлом зв'язку SIMO, що виконує поділ. Якщо пакети даних у зв'язку з одним маршрутом досягли кінцевого адресата, їх дані довгого пакета розділяються востаннє, а "сміттєві" дані видаляються, щоб відновити вихідний пакет даних. Змішані дані необов'язково повинні бути однаковими типами даних. Наприклад, один абонент, який викликає, може розмовляти по телефону та одночасно відправляти текстові повідомлення, тим самим одночасно генеруючи або одержуючи два різних потоки даних. Якщо, однак, розділені пакети даних призначені для продовження маршрутизації далі в мережі у незмішаному стані, то до пакетів даних включаються "сміттєві" дані, щоб уникнути перехоплення даних.

При транспортуванні однорідних даних безпека досягається, насамперед, за допомогою скремблювання, як зображено на рис. 64, або за допомогою комбінації скремблювання та шифрування, як зображено на рис. 66A. Комбінацію змішування з наступним скремблюванням, використовуваним в обох прикладах, додатково розглянуто на рис. 67G, де операція змішування 1094 змішує вхідні дані субпакетів 1090A, 1090B і 1090C для формування довгого не скремблюваного пакета даних 1091. Потім операція скремблювання 926 у цьому прикладі виконує лінійний фазовий зсув одним слотом даних праворуч, наприклад, коли дані 1A в слоті 1 не скремблюваного пакета переміщуються в слот 2 у скремблюваний пакет, а дані 1C у слоті 7 переміщуються в слот 8 у не скремблюваному пакеті і т. д., щоб створити не скремблюваний довгий пакет даних 1107.

Операція розбивки 1087 розбиває не скремблюваний довгий пакет даних 1107 уздовж ліній поділу 1092 після 6-го та 12-го слотів для створення виданих субпакетів 1093G, 1093H і 1093J. Результат фазового зсуву не тільки впливає на положення даних у виданих субпакетах, але фактично змінює вміст пакетів. Наприклад, якщо сегмент даних 3D у положенні слоту 12 у не скремблюваному довгому пакеті даних 1107 переміщується в положення 13 після скремблювання, операція розбору 1087, розташована в лінії поділу 1092 після 12 слоту, природно витісняє дані з субпакета 1093H в 1093J, що підтверджується порівнянням субпакета 1093H з його новою послідовністю сегментів даних J-1C-2C-3 C-J-2D (де J вказує на "сміттєві" дані) відносно субпакета 1093E на рис. 67E, що має послідовність сегментів даних 1C-2 C-3C-J-2D-3D.

На рис. 67H показано об'єднання алгоритмічного змішування, тобто перерозподіл вхідних даних з субпакетів для формування довгого пакета даних, причому наступний алгоритм скремблювання може відтворюватися ідентично, шляхом об'єднання операцій змішування та

скремблювання в один крок, просто шляхом зміни алгоритму перерозподілу. Операція гібридного змішування та скремблювання 1094A ідентична алгоритму попереднього змішування, за винятком того, що вона зміщує дані в одну позицію праворуч у довгому пакеті даних 1107 під час перерозподілу. Наприклад, сегмент даних 1A в субпакеті 1090A перерозподіляється в слот 2 довгого пакета даних 1107, а не в слот 1, сегмент даних 3D у субпакеті 10903 перерозподіляється в слот 13 довгого пакета даних 1107, а не в слот 12. Остаточні видані субпакети 1093G, 1093H і 1093J ідентичні виведення субпакетів з використанням послідовності змішування з наступним скремблюванням, як показано на рис. 67G. По суті, алгоритм скремблювання після змішування являє собою інший алгоритм змішування. Оскільки різниці в кінцевому виведенні немає, по всьому тексту розглядуваного опису операції змішування та скремблювання будуть визначатись як окремі, з розумінням того, що два числових процеси можуть бути об'єднані в один. Так само зрозуміло, що зворотний процес, який дескремблює і потім розділяє пакет даних, може замінятися однією комбінованою операцією, яка виконує як дескремблювання, так і розділення в один крок.

При передачі даних одним маршрутом пакети даних не можуть проходити паралельними шляхами, але повинні переміщатися послідовно одним шляхом між медіа-серверами або між пристроєм клієнта та хмарним шлюзом, тобто передача даних за "останню милю". Перш ніж субпакети даних зможуть бути відправлені в мережу, вони повинні позначитися одним або декількома заголовками для ідентифікації пакета, щоб цільовий вузол зв'язку розумів, що робити із вхідним пакетом. Хоча форматування та інформація, які містяться в цих заголовках, докладніше описуються нижче, для ясності і спрощення реалізацію ідентифікації пакетів показано на рис. 67I. Як показано нижче, серія пакетів даних 1099A, 1099B, 1099C і 1099Z надходить послідовно у вузол зв'язку. Кожен пакет даних містить у собі заголовок, наприклад, 1102A, і його відповідні дані, наприклад, 1090A.

Коли пакети даних потрапляють у вузол, операція 1600 відокремлює заголовок від даних для обробки. Як показано для першого вхідного пакета 1099A, заголовок 1102A, позначений Hdr A, відділяється від пакета даних 1099A, а потім передається на операцію зчитування тегів 1602, що визначає, чи одержав вузол зв'язку які-небудь інструкції стосовно пакета 1099A. Якщо вузол не одержав жодних інструкцій стосовно пакета 1099A, відповідні дані видалятимуться. Це показано на прикладі субпакета 1092, позначеного як субпакет Z, що містить дані з ланцюжка повідомлень 6, 7, 8, 9, не пов'язаних із жодною інструкцією, отриманою вузлом зв'язку. Однак, якщо пакет даних "очікується", а саме, його тег відповідає інструкції, раніше прийнятій вузлом зв'язку з іншого сервера, тоді розпізнані пакети даних, у цьому випадку субпакети 1090A, 1090B і 1090C, відправляються на змішування 1089. Відповідний алгоритм, раніше обраний для вхідних пакетів даних, завантажується з таблиці 1050 алгоритму змішування в операцію змішування 1089. Інакше кажучи, раніше вузол зв'язку був проінструктований щодо того, якщо він приймає три пакети, ідентифіковані як Hdr A, Hdr B і Hdr C відповідно, він повинен змішувати ці три пакети згідно з конкретним алгоритмом змішування, зазначеним у Таблиці 1050. Як було зазначено вище, цей алгоритм змішування може включати операцію скремблювання.

Відповідно до цього опису, операція змішування 1059 потім послідовно видає субпакети даних 1093D, 1093E і 1093F, кожен з яких позначений новим ідентифікаційним заголовком, тобто Hdr D, Hdr E і Hdr F, у пакети даних 1099D, 1099E і 1099F, які готові для транспортування до наступного вузла зв'язку в мережі. У зв'язку з одним маршрутом ці пакети даних відправляються по черзі тим самим маршрутом до цільового адресата. Хоча на схемі показано, як теги використовуються для ідентифікації пакетів для змішування, метод ідентифікації тегу ідентичний для виконання певних операцій скремблювання та шифрування, а також їх зворотних функцій дешифрування, дескремблювання та розділення.

Операції змішування та розділення можуть застосовуватися до решітчастої маршрутизації з декількома маршрутами, описаній далі, використовуючи операції змішування та розділення з декількома виведеннями. Різні виведення, представлені стрілками, спрямованими назовні, у символі розділення SMO 1101 на рис. 67F можуть використовуватися для направлення пакетів даних по мережі в різних напрямках, різними шляхами і маршрутами. Інструкції, отримані від вузла зв'язку, вказують тег, який застосовуватиметься як заголовок для кожного з розділених пакетів, а також ідентифікатор вузла, якому повинен відправлятися кожен з розділених пакетів. Вузли-одержувачі також отримують вказівку очікувати пакети. Аналогічно, операція змішування з множинними введеннями та множинними виведеннями 1094, показана на рис. 67B, може застосовуватися до повідомлень із декількома маршрутами. Як показано нижче в цьому додатку, змішування пакетів даних MISO та MIMO і розділення пакетів даних SMO є ключовими елементами при реалізації багатопляхової і решітчастої маршрутизації. Навіть за відсутності скремблювання та шифрування пакетів даних, багатопляхова і решітчаста маршрутизації

пакетів даних значно зменшує ризик цілеспрямованого перехоплення даних кіберпіратами, аналіз пакетів і атаки "людина посередині" в мережі, оскільки жоден вузол зв'язку не містить усієї розмови, не приймає або не передає всі дані в повному обсязі. Кількість субпакетів, зображених на показаних малюнках, використовується винятково з ілюстративною метою.

5 Фактична кількість переданих пакетів може містити десятки, сотні або навіть тисячі субпакетів.

Маршрутизація пакетів. Як показано на прикладі додатка, один шлях містить послідовний потік пакетів даних, використовуваних у мережній комунікації на основі пакетної комутації, такий як Інтернет. Хоча цей шлях може змінюватися згодом, перехоплення потоку даних за допомогою аналізу пакета може, як мінімум на деякий період часу, дати кіберпіратові повні пакети даних взаємозалежної та послідовної інформації. Без скремблювання та шифрування, використовуваних у системі комунікації SDNP, описуваній відповідно до цього винаходу, будь-яка послідовність пакетів даних після перехоплення може бути легко інтерпретована при атаці "людина посередині", що забезпечить ефективні й повторювані кібератаки.

15 Такий одношляховий зв'язок є основою інтернет-, VoIP- і OTT-зв'язку, і однією з причин того, що інтернет-зв'язок сьогодні досить небезпечний. Тоді як послідовні відправлені пакети можуть проходити різні маршрути, поруч із вузлами зв'язку джерела і призначення, імовірність того, що послідовні пакети будуть дотримуватися того самого маршруту і проходитимуть через ті самі сервери, стає все більшою, оскільки маршрутизація пакетів в Інтернеті визначається постачальниками послуг, які монополізують будь-яку географічну зону. Просто відслідковуючи маршрутизацію пакета назад до його джерела, потім аналізуючи пакет поруч із джерелом, імовірність перехоплення декількох пакетів того самого сеансу обміну повідомленнями і потоку даних різко зростає, тому що зв'язок здійснюється тільки одним географічним провайдером інтернет-послуг, або ISP.

25 Як графічно зображено на рис. 68A, одношляховий зв'язок 1110 являє собою послідовний потік даних 1111 від вузла зв'язку $N_{u,v}$ до іншого вузла зв'язку, у цьому випадку до вузла зв'язку $N_{w,z}$. Хоча шлях може змінюватися згодом, у кожному конкретному випадку кожен взаємозалежний пакет послідовно передається в мережу, що проходить до свого адресата уздовж одного єдиного шляху. Відповідно позначення, вузол зв'язку $N_{u,v}$ означає вузол зв'язку, розміщений на сервері "v", розташованому в мережі "u", тоді як вузол зв'язку $N_{w,z}$ позначає вузол зв'язку, розміщений на сервері "z", розташованому в мережі "w". Мережі "u" і "w" являють собою хмари, що належать різним ISP. Хоча маршрутизація пакетів даних всередині інтернет-маршрутизації може переноситися будь-якою кількістю ISP, як пакети даних біля їхнього адресата, вони незмінно переносяться загальним ISP і мережею, що спрощує трасування та аналіз послідовних пакетів даних, що включають одну й ту саму розмову. Цей момент ілюструється графічно на рис. 68B, де одношляховий зв'язок 1111 здійснюється через ряд серверів 1118, що є єдиною мережею зв'язку послідовних шляхів 1110. Відповідно до зображення, зв'язок починається з вузла зв'язку $N_{0,0}$, що проходить послідовно через вузли зв'язку $N_{0,1}$ і $N_{0,2}$, все в тій же мережі з номером "0", до досягнення вузла зв'язку $N_{2,3}$, який переноситься іншим ISP через мережу 2. Після цього дані відправляються на кінцеві вузли, на мережу 1, тобто, вузли зв'язку $N_{1,4}$ і $N_{1,f}$. Таким чином, під час переходу пакетні дані, передані з самого початку в Інтернет, залишаються на сервері 0, перш ніж він зможе поширити їх на іншу мережу ISP. Аналогічно, у міру наближення пакета даних до свого місця призначення, імовірність того, що послідовні пакети пройдуть через ті самі вузли, збільшується, оскільки всі вони розташовані в мережі 1 ISP.

45 На відміну від одношляхового пакетного зв'язку, що використовується для інтернет-зв'язку, OTT-зв'язку та VoIP-зв'язку, у варіанті виконання зв'язку SDNP, відповідно до цього винаходу, вміст пакетів даних не переноситься послідовно взаємозалежними пакетами, що містять інформацію з загального джерела або абонента, який викликає, - це відбувається у фрагментарній формі, динамічно змішуючи та повторно змішуючи контент, що виходить із безлічі джерел і абонентів, які викликають, причому згадані дані збирають неповні фрагменти даних, контенту, голосу, відео та файлів із різними типами даних з наповнювачами "сміттєвих" даних. Перевагою реалізації, що розглядається, фрагментації даних і транспортування є те, що навіть незашифровані і не заскрембльовані пакети даних майже неможливо інтерпретувати, оскільки вони являють собою комбінацію незв'язаних даних і типів даних.

55 Як показано на рис. 68A, SDNP зв'язок фрагментованих пакетів даних не є послідовним, як у транспорті з одним маршрутом 1110, а є паралельним, з використанням транспорту з множинними маршрутами 1114 або транспорту з "решітчастим маршрутом" 1114. У транспорті з множинними маршрутами 1112 масив із двох або більше вузлів зв'язку з комутацією пакетів $N_{u,v}$ і $N_{w,z}$ одночасно створює і транспортує дані декількома маршрутами 1113A, 1113B, 1113C, 60 1113D і 1113E. Оскільки показано п'ять маршрутів, транспортування може здійснюватися, як

мінімум, двома маршрутами і, як максимум, до десятка або більше, якщо це необхідно. Важливо підкреслити, що ця реалізація мережі зв'язку не є простою повторюваною маршрутизацією, яка зазвичай використовується мережею Інтернет і мережею з комутацією пакетів, тобто там, де ті самі дані можуть відправлятися одним шляхом або навіть декількома шляхами одночасно.

5 Передача або обмін даними повними взаємозалежними пакетами даних надлишково декількома каналами фактично збільшує ризик злому, оскільки дає кіберпіратіві множинні джерела ідентичних даних для аналізу та злому.

Замість цього, у комунікаційній системі SDNP, інформація фрагментується, наприклад, із деякою частиною даних, що відправляються маршрутами 1113A, 1113B і 1113D, без даних, що відправляються спочатку маршрутами 1113C і 1113E, а потім, пізніше, фрагментовані дані розділяються та комбінуються по-різному і відправляються маршрутами 1113A, 1113C і 1113E без передачі даних маршрутами 1113B і 1113D. Приклад транспорту з множинними маршрутами 1112 показано на рис. 68C за допомогою мережі, що містить масив серверів зв'язку 1118, виконаних із можливістю встановлення декількох шляхів передачі даних між вузлами зв'язку $N_{0,0}$ і $N_{f,f}$. Відповідно до зображення, транспортування з множинними маршрутами відбувається на чотирьох комплектах взаємозалежних серверів, що представляють мережі з 1 по 4. Один шлях даних, маршрут 1113A, містить вузли зв'язку $N_{1,1}$, $N_{1,2}$, $N_{1,3}$ і $N_{1,4}$. Паралельний шлях даних, маршрут 1113B, містить вузли зв'язку $N_{2,1}$, $N_{2,2}$, $N_{2,3}$, і $N_{2,4}$. Аналогічно, маршрут паралельних даних 1113C містить взаємозалежні вузли зв'язку $N_{3,1}$, $N_{3,2}$, $N_{3,3}$ і $N_{3,4}$, тоді як маршрут 1113D містить взаємозалежні вузли зв'язку $N_{4,1}$, $N_{4,2}$, $N_{4,3}$ і $N_{4,4}$.

При транспортуванні "решітчастим маршрутом" 1114, зображеним також на рис. 68D, повідомлення відправляється декількома взаємодіючими маршрутами, включаючи вищезгадані маршрути 1113A, 1113B, 1113C, 1113D і 1113E, а також перехресні з'єднання 1115 A-1115E між маршрутами 1113 A-113D. Разом з'єднання утворюють "решітку", за допомогою якої пакети даних перемиюються через будь-які комбінації маршрутів і навіть змішуються або повторно комбінуються динамічно, коли пакети даних відправляються іншими маршрутами. При решітчастій маршрутизації 1114 мережа містить масив серверів зв'язку 1118, призначених для встановлення шляхів передачі решітчастих даних між вузлами зв'язку $N_{0,0}$ і $N_{f,f}$. Відповідно до зображення, транспортування з множинними маршрутами відбувається на взаємозалежних серверах з горизонтально і вертикально орієнтованими шляхами передачі даних. Горизонтально орієнтований маршрут 1113A містить вузли зв'язку $N_{1,1}$, $N_{1,2}$, $N_{1,3}$ і $N_{1,4}$, маршрут 1113B містить вузли зв'язку $N_{2,1}$, $N_{2,2}$, $N_{2,3}$ і $N_{2,4}$, маршрут 1113C містить взаємозалежні вузли зв'язку $N_{3,1}$, $N_{3,2}$, $N_{3,3}$ і $N_{3,4}$, і нарешті маршрут 1113D містить взаємозалежні вузли зв'язку $N_{4,1}$, $N_{4,2}$, $N_{4,3}$ і $N_{4,4}$. Вертикально орієнтований маршрут 1115A містить вузли зв'язку $N_{1,1}$, $N_{2,1}$, $N_{3,1}$ і $N_{4,1}$, маршрут 1115B містить вузли зв'язку $N_{1,2}$, $N_{2,2}$, $N_{3,2}$ і $N_{4,2}$, маршрут 1115C містить зв'язані між собою вузли зв'язку $N_{1,3}$, $N_{2,3}$, $N_{3,3}$ і $N_{4,3}$ і маршрут 1115D містить зв'язані між собою вузли зв'язку $N_{1,4}$, $N_{2,4}$, $N_{3,4}$ і $N_{4,4}$. Мережа додатково може бути доповнена діагональними з'єднаннями 1119, як показано на рис. 68E.

Транспортування з множинними маршрутами може комбінуватися різними способами за допомогою скремблювання і шифрування. Приклад транспортування з множинними маршрутами без скремблювання показано на рис. 69, де мережа серверів зв'язку 1118 передає пакет даних 1055 з вузла зв'язку $N_{0,0}$ у момент часу t_0 у вузол зв'язку $N_{f,f}$, у момент часу t_f . На транспорті 1112 вузол зв'язку $N_{0,0}$ виконує операцію розділення 1106, що відсилає сегменти даних 1C і 1E у пакет даних 1125A по маршруту даних 1113A, відправляючи сегмент даних 1B у пакеті даних 1125B по маршруту даних 1113B, відправляючи сегмент даних 1D у пакеті даних 1125C по маршруту даних 1113C і відправляючи сегменти даних 1A і 1F у пакет даних 1125D по маршруту даних 1113D. Субпакети можуть включати сполучення даних і незв'язані субпакети або "сміттєві" дані. Оскільки субпакети не скремблюються, послідовність сегментів даних 1C та 1E у пакеті даних 1125A залишається в послідовному порядку, навіть якщо інші сегменти даних можуть вставлятися між або до, або після них. Нарешті, у вузлі зв'язку $N_{f,f}$ операція змішування 1089 відновлює вихідний пакет даних у момент часу t_f . У будь-який час t_n між часом t_0 і часом t_f вміст пакетів даних з 1125A по 1125D залишається постійним.

Простий варіант вищезгаданого транспортування з множинними маршрутами без скремблювання показано на рис. 70, він містить транспорт із множинними маршрутами зі статичним скремблюванням, що означає, що вхідний пакет даних 1055 скремблюється перед розділенням і передається по декількох маршрутах у мережі. Зокрема, вузол зв'язку $N_{0,0}$ виконує операцію скремблювання та розділення 1071, а не просто операцію розділення 1106, як показано на рис. 69. Отримані скрембльовані змішані пакети даних 1126 A-1126D, як і в попередньому прикладі, є статичними і тимчасовими інваріантами, які залишаються незмінними в будь-який час t_n , тоді як вони незалежно проходять по мережі шляхами 1113 A-1113D

відповідно, поки не досягнуть кінцевого вузла зв'язку $N_{i, f}$, де вони знову поєднуються та дескремблюються за допомогою операції дескремблювання та змішування 1070 для відновлення вихідного пакета даних 1055. Порівняно з попереднім прикладом на рис. 69, єдине істотне розходження в пакетах даних 1126 A-1126D на рис. 70 полягає в тому, що пакети скремблюються, тобто сегменти даних, які вони містять, не представлені у вихідному послідовному порядку. Наприклад, у пакеті даних 1126A сегмент даних 1E зустрічається до сегмента 1B, а в пакеті даних 1126D сегмент даних 1D зустрічається до сегмента 1A. Недоліком статичного пакетного зв'язку є те, що, хоча він не піддається простому аналізу пакетів, він дозволяє кіберпіратів аналізувати постійні, незмінювані дані. Проте, оскільки дані, присутні в якому-небудь одному пакеті даних, що переміщується будь-яким маршрутом, є неповними, фрагментованими, скрембльованими та змішаними з іншими незв'язаними джерелами даних і розмовами, він все-таки значно перевершує ОТТ-зв'язок через Інтернет.

Кращим за статичне є динамічне скремблювання, яке зображене на рис. 71A, де повторне скремблювання пакетів, тобто операція повторного скремблювання US 1017, змінює порядок сегментів даних у пакеті даних, коли пакет даних проходить мережу, що означає, що порівняння будь-якого пакета даних, який проходить заданим маршрутом, змінюється згодом. Наприклад, відносно маршруту проходження пакетів даних 1113A у пакеті даних 1126A у момент часу t_3 одразу після проходження операції повторного скремблювання US 1017 у вузлі зв'язку $N_{1,3}$, сегмент даних 1E знаходиться в 2 тимчасовому слоті і передує сегменту даних 1B, розташованому в 4 тимчасовому слоті. У момент часу t_4 після того, як вузол зв'язку $N_{1,4}$ виконає операцію повторного скремблювання US 1017, пакет даних 1127A зміниться з сегментом даних 1B, розташованим до сегмента 1E, послідовно розташованим у тимчасових слотах 3 і 4. Порівнюючи пакети даних 1126D-1127D, положення сегментів даних 1D і 1A змінюється, але порядок залишається незмінним. Цей метод використовує метод динамічного скремблювання кожного сегмента даних у пакеті даних, а не тільки дані з певного джерела або розмови. Можна змінювати довжину пакета одразу після його дескремблювання та перед повторним скремблюванням, наприклад, шляхом вставки або видалення "сміттєвих" даних. Проте у показаному прикладі довжина пакета залишається фіксованою, тільки зі зміною їхньої послідовності.

Відповідно до зображення, перший вузол зв'язку $N_{0,0}$ виконує операцію скремблювання та розділення 1071, останній вузол зв'язку $N_{i, f}$ виконує операцію змішування та дескремблювання 1070, а всі проміжні вузли зв'язку виконують операцію повторного скремблювання US 1017. У кожному випадку операція дескремблювання залежить від часу або стану вхідного пакета, а операція скремблювання використовує час або стан вихідного пакета даних. При паралельному транспортуванні з множинними маршрутами розділення відбувається тільки один раз у вузлі зв'язку $N_{0,0}$, і змішування відбувається тільки один раз, наприкінці транспортування у вузлі зв'язку $N_{i, f}$. Методологічно ця послідовність може бути віднесена до категорії "скремблювання, а потім розділення". У варіанті виконання динамічного скремблювання, як показано на рис. 71A, відомого як послідовне або лінійне скремблювання, незважаючи на послідовність, попередні операції повинні відтворюватися у порядку, зворотному тому, в якому вони проводилися; у такому разі, зміна порядку розташування сегментів даних у пакеті даних відбувається алгоритмічно, незалежно від характеру і походження контенту. Таким чином, перші вузли зв'язку після розділення, а саме вузли зв'язку $N_{1,1}$, $N_{2,1}$, $N_{3,1}$ і $N_{4,1}$, виконують ту саму операцію дескремблювання, щоб скасувати вплив первинного скремблювання при виконанні операції скремблювання і потім розділення 1071, повертаючи кожен сегмент даних, що містить дані, у його вихідне місце розташування, в якому він перебував до повторного скремблювання. У процесі розділення місце розташування пакета залишається незмінним, там, де він спочатку розташовувався з невикористовуваними слотами, заповненими "сміттєвими" даними. Наприклад, якщо сегмент даних 1B переміщується у п'яту позицію в пакеті за допомогою операції скремблювання та розділення 1118, пакет, що містить сегмент даних 1B, зберігатиме його у п'ятій позиції після розділення. Шляхом дескремблювання пакет перемістить сегмент даних 1B назад у другий слот, якому він належить, навіть якщо усі інші слоти заповнені "сміттєвими" даними. Дислокація "сміттєвих" даних не має значення, оскільки пакети "сміттєвих" даних видалятимуться, тобто "повторно видалятимуться" пізніше в процесі відновлення даних. Як тільки позиція певного сегмента даних відновлюється до вихідного слота за допомогою операції дескремблювання, вона може знову скремблюватися, переміщуючи її в нове положення. Комбінація відновлення сегмента даних у вихідне положення і наступне скремблювання в нову позицію означає, що процес "скремблювання" містить у собі дескремблювання, а потім скремблювання, звідси і назва "повторне скремблювання" US 1017.

Спрощений опис раніше деталізованого методу "лінійного скремблювання, потім розділення", показаного на рис. 71В, протиставляється двом іншим альтернативним виконанням описаного винаходу, що називається тут "складовим скремблюванням, потім розділенням" і "лінійним розділенням, потім скремблюванням". При лінійному методі скремблювання, потім розділенні, послідовно і багаторазово скремблюючи та дескремблюючи кожен пакет даних, оновлюється безпека пакета даних. Таким чином скремблювання, вперше виконане в операції скремблювання та розділення 1071, повинно відмінитися операцією повторного скремблювання US 1017 окремо в кожному зі шляхів проходження даних, де дужки символічно являють собою кілька паралельних шляхів або маршрутів, означаючи час, стан або числове початкове значення, використовуване для вибору та виконання операції скремблювання до розділення в операції скремблювання та розділення 1071, потім передається першому вузлу зв'язку на кожному маршруті зв'язку, отже це дає можливість виконати операцію повторного скремблювання US 1017. Після цього кожен маршрут окремо скремблює і дескремблює пакети даних, що проходять цим маршрутом, при цьому операція повторного скремблювання US 1017 завжди використовує час, стан або числове початкове значення, використовуване для виконання останнього скремблювання, а потім використовує свій поточний час або стан для виконання нового скремблювання. На останньому етапі операції змішування та розділення 1070 скремблювані компоненти повторно збираються в скремблюваній формі, а потім остаточно дескремблюються, використовуючи стан або час, коли вони останній раз скремблювалися для відновлення вихідних даних.

У прикладі "вкладеного скремблювання та розділення", також показаного на рис. 71В, операція 1071 скремблювання потім розділення 1071 спочатку скремблює пакет даних у початковий час або стан, а потім, після розділення даних на кілька маршрутів, кожен шлях даних незалежно виконує другу операцію скремблювання 926, не пов'язану з першою, без скасування першої операції скремблювання. Оскільки операція скремблювання виконується у вже скремблюваному пакеті даних, скремблювання можна розглядати як "вкладене", тобто одне скремблювання всередині іншого. Говорячи мовою програмування, для вкладених об'єктів або програмного коду перше скремблювання, виконуване за допомогою операції скремблювання та розділення 1071, містить "зовнішній" цикл скремблювання, тоді як друге та всі наступні скремблювання операції повторного скремблювання US 1017 являють собою внутрішній цикл скремблювання. Це означає, що дані, які проходять мережу, були двічі скремблювані і повинні дескремблюватися двічі для відновлення вихідних даних. Заклучний етап внутрішнього циклу скремблювання містить операцію дескремблювання 928, відновлюючи пакети даних кожного маршруту в ті самі умови, тобто, ту саму послідовність сегментів даних, як одразу ж після першого розділення пакетів. Потім пакети даних знову збираються в єдиний пакет даних і дескремблюються з використанням операції змішування та дескремблювання 1070.

Ця ж концепція вкладених операцій може використовуватися при виконанні вкладених операцій розділення та змішування, як показано на рис. 71С. У рамках додатку SDNP клієнта 1335, різні джерела даних, включаючи відео, текстові, голосові і файли даних, можуть змішуватися, серіалізуватися, вставлятися з "сміттєвими" даними, скремблюватися, а потім шифруватися операцією MSE 1075. Облікові дані безпеки, включаючи ключ 1030W і початкові значення 929W, можуть передаватися зі стільникового телефону клієнта-відправника 32 безпосередньо на планшет клієнта-одержувача 33 без використання медіа-вузлів, що несуть контент. Наприклад, ця інформація може відправлятися одержувачу, використовуючи окрему мережу "сигнального сервера" (описану нижче) або, альтернативо, оскільки початкові значення і ключі не містять корисної інформації для сторонніх осіб, така інформація може відправлятися навіть клієнту-одержувачу через Інтернет. Ця перша операція, виконувана у пристрої або додатку клієнта, являє собою початок зовнішнього циклу, використовуваного для реалізації безпеки клієнта незалежно від мережі SDNP.

Після того як нечитабельний зашифрований текст 1080W клієнта змішується, скремблюється, засекречується та шифрується, він відправляється на сервер шлюзу SDNP N_{0,0}, де він знову обробляється з використанням різних спільних секретів із різними алгоритмами, станами і повноваженнями безпеки, характерними для конкретної мережі, такими як початкове число 929U і ключ 1030U при підготовці до передачі через хмару SDNP. Цей внутрішній цикл забезпечує безпеку хмарного сервера і повністю не залежить від циклу безпеки клієнта. У рамках SSE операції 1140 шлюзу для вхідних пакетів даних, пакет даних може бути повторно розбитий на різні субпакети і зашифрований у зашифрований текст 1080U і 1080V для багатопотокової або об'єднаної передачі.

У підсумку численні субпакети надходять до призначеного шлюзу $N_{f, f}$, де вони обробляються операцією 1141 DMU для скасування операції розділення початкового шлюзу, тобто операція 1141 DMU скасовує дію операції 1140 SSE, що завершує функцію внутрішнього циклу безпеки. Таким чином, шлюз $N_{f, f}$ скасовує всі пов'язані з мережею заходи безпеки, реалізовані вхідним шлюзом $N_{0,0}$, і відновлює вихідний файл, у цьому випадку зашифрований текст клієнта 1080W, до того самого стану, що й при вході в хмару SDNP.

Але, оскільки цей пакет даних уже змішувався, скремблювався та зашифровувався, пакет даних, що містить шифрований текст 1080 W, що виходить зі шлюзу SDNP і відправляється клієнту-одержувачу, як і раніше зашифрований і не може інтерпретуватися нічим іншим, крім додатка 1335 клієнта-одержувача. Після того як відновлений зашифрований текст доставляється клієнту, він дешифрується та дескремблюється за допомогою операції DUS 1076 відповідно до стану 990 клієнта-відправника, коли він був створений, у момент часу t_0 і в остаточному підсумку розбивається для відновлення різних джерел компонентів даних, включаючи відео-, текстові, голосові файли і файли даних, завершуючи зовнішній цикл безпеки.

Таким чином, щоб запобігти підриву мережі, наприклад, коли кіберзлочинець, діючи в ролі оператора мережі SDNP, намагається підрвати безпеку SDNP "зсередини", облікові дані безпеки зовнішнього циклу, наприклад, розподілені (спільні) секрети, початкові стани, ключі, зони безпеки і т.д., навмисне робляться відмінними від даних внутрішнього циклу безпеки.

В іншому варіанті здійснення цієї розробки, зображеному на рис. 71B, у процесі "лінійного розділення і наступного скремблювання" дані спочатку розділяються, а потім окремо скремблюються по кожному маршруту даних. Слідом за операцією розділення даних 1057 реалізується і виконується незалежна операція скремблювання 926 маршрут за маршрутом. Після скремблювання пакети даних, що проходять кожним маршрутом, послідовно повторно скремблюються за допомогою операцій повторного US скремблювання 1017, що припускає рескремблювання вхідного пакета з використанням того самого часу, стану або числових початкових значень, використовуваних операцією скремблювання 926 для їхнього створення. Після цього кожен маршрут окремо скремблює і рескремблює пакети даних, що проходять цим маршрутом, при цьому операція рескремблювання US 1017 завжди використовує час, стан або числове значення, використовуване для виконання останнього скремблювання, а потім використовує поточний час або стан для виконання нового скремблювання. Останній етап включає операцію рескремблювання 928, відновлення пакетів даних кожного маршруту в колишній стан, тобто тієї самої послідовності сегментів даних, що була після першого поділу пакетів. Потім пакети даних повторно збираються в один пакет рескремблених даних з використанням операції змішування 1061.

Незалежно від використовуваної послідовності змішування та скремблювання оброблені пакети даних також можуть бути піддані статичному або динамічному шифруванню для поліпшення та підвищення ступеня безпеки. Приклад цієї комбінації, який показано на рис. 72, містить метод, описаний як "статичне скремблювання з наступним розділенням і динамічним шифруванням", що включає такі етапи:

1. Початком служить введення незашифрованого тексту в момент часу t_0 .
2. Скремблювання незашифрованого тексту 1055 з використанням статичного скремблювання пакетів 926 у момент часу t_1 .
3. Поділ скремблених незашифрованого тексту 1130 на кілька розділених пакетів даних 1131A, 1133A та інші з використанням операції розділення 1106 у момент часу t_2 .
4. Направлення розділених пакетів даних 1131A, 1133A та інших на кілька різних паралельних маршрутів, які не пересікаються, у момент часу t_3 (зверніть увагу на те, що тільки два з цих паралельних маршрутів докладно показано на рис. 72).
5. Незалежне шифрування кожного пакета даних 1131A, 1133A та інших у момент часу t_4 з використанням шифрування 1026, включаючи ключі шифрування та числові початкові значення, що відповідають стану 994, у результаті одержують зашифрований текст 1132A, 1134A та ін.
6. Незалежне дешифрування кожного пакета даних 1132A, 1134A та інших з інформацією про стан 994, включаючи розподілені (спільні) секрети, ключі, числові початкові значення і т.д., використовуючи дешифрування 1032, у результаті одержують незашифрований текст 1131B, 1133B та ін.
7. Незалежне повторне шифрування незашифрованого тексту 1131B, 1133B та інших з використанням шифрування 1026 у момент часу t_6 з використанням ключів шифрування і числових початкових значень, що відповідають стану 996, в результаті одержують зашифрований текст 1132B, 1134B та ін.
8. Незалежне дешифрування кожного пакета даних 1132B, 1134B та інших з інформацією про стан 996, включаючи розподілені (спільні) секрети, ключі, числові початкові значення і т.д.,

використовуючи дешифрування 1032, в результаті одержують незашифрований текст 1131C, 1133C та ін.

9. Змішування незашифрованого тексту 1131C, 1133C та інших у момент часу t_7 з використанням операції змішування 1089 для одержання скрембльованого незашифрованого тексту 1130.

10. Рескремблювання скрембльованого незашифрованого тексту 1130 у момент часу t_8 з використанням стану 991, що відповідає моменту часу t_1 , у момент початкового скремблювання для відновлення вихідного незашифрованого простого тексту 1055.

У представленому прикладі обробка вихідного пакета даних включає послідовне застосування скремблювання, розділення та шифрування, представлених як операція 1140. Кінцева операція містить у собі дешифрування, змішування та рескремблювання, представлені операцією 1141. Усі проміжні етапи включають повторне шифрування, що містить у собі як дешифрування, так і шифрування.

Приклад використання цього методу при мультимаршрутній передачі проілюстровано на рис. 73, де вузол зв'язку $N_{0,0}$ виконує операцію скремблювання, розділення, шифрування 1140A, а вузол зв'язку $N_{i,1}$ виконує операцію дешифрування, змішування та рескремблювання 1141A, тоді як усі проміжні вузли виконують операцію повторного шифрування 1077. При мультимаршрутній передачі відповідно до цього винаходу можливі різні комбінації статичного і динамічного скремблювання та статичного і динамічного шифрування.

Як опція для скремблювання, розділення та шифрування, як альтернативний спосіб здійснення цієї розробки пакети даних можуть бути розділені, потім скрембльовані і зашифровані з використанням операції розділення, скремблювання, шифрування 1140B, показаної на рис. 74. При використанні такого методу вхідний пакет даних спочатку розділяється в ході операції 1106. Далі пакети даних на кожному маршруті незалежно скремблюються в ході операції 926 і зашифровуються операцією 1026. Потім отримані пакети даних можуть бути повторно дешифровані незалежно один від одного, а потім повторно зашифровані з використанням операції повторного шифрування 1077 або можуть бути дешифровані, рескрембльовані, повторно скрембльовані і повторно зашифровані з використанням операції 1045 повторної обробки пакетів DUSE.

На відміну від решітчастої маршрутизації, описаної нижче, при мультимаршрутній передачі, як показано на рис. 69-73, кожен пакет даних, що проходить через мережу, обробляється тільки один раз даним вузлом зв'язку, і жоден вузол зв'язку не обробляє більше одного пакета даних, що містять зв'язані дані або загальний ланцюжок повідомлень, тобто маршрути даних 1113A, 1113B, 1113C і 1113D є окремими, чіткими і непересічними.

Решітчаста маршрутизація. Повертаючись до рис. 68A слід зазначити, що решітчаста маршрутизація та передача пакетів, описані тут, подібні до паралельної мультимаршрутної передачі, за винятком того, що пакети даних, переміщувані в мережі різними маршрутами, можуть перетинатися маршрутами на одних і тих самих серверах. При статичній решітчастій маршрутизації, як викладено тут, ці пакети даних проходять через загальний сервер без взаємодії, так, ніби жодних інших даних ланцюжків повідомлень або комунікаційних даних навіть не існувало. При динамічній решітчастій маршрутизації при введенні вузла зв'язку пакети даних можуть взаємодіяти з іншими пакетами даних, присутніми одночасно на тому самому сервері.

Використовуючи раніше описаний метод розділення та змішування, групи сегментів даних можуть відділятися або видалятися з одного пакета даних, поєднуватися або приєднуватися до іншого пакета даних і відправлятися в напрямку до місця призначення, відмінному від того, за яким він був отриманий. Решітчаста маршрутизація відповідно до цієї розробки може використовувати пакети даних змінної або фіксованої довжини. У пакетах змінної довжини кількість сегментів даних, що містять пакет даних, може змінюватися залежно від кількості трафіку, що проходить через даний вузол зв'язку. При решітчастій передачі з використанням пакетів даних фіксованої довжини кількість сегментів даних, використовуваних для формування повного пакета даних, фіксована на деякому постійному числі або, як варіант, на деякій кількості сегментів даних, скоректованих з обліком квантованого цілочисельного приросту.

Основна розбіжність між використанням пакетів даних змінної та фіксованої довжини полягає у використанні "сміттєвих" даних як вмісту пакетів. У пакетах зі змінною довжиною даних використання "сміттєвих" даних є необов'язковим, ґрунтується головним чином на міркуваннях безпеки або для використання невикористовуваних маршрутів з метою відстеження затримок проходження мережі. Використання "сміттєвих" даних у пакетах даних фіксованої довжини є обов'язковим, оскільки неможливо забезпечити наявність достатньої кількості сегментів даних для заповнення пакетів, що виходять з вузла зв'язку. Таким чином, "сміттєві" дані обов'язково використовуються постійно та безупинно як вміст пакетів для забезпечення

того, щоб кожен пакет даних, що виходить із сервера, був заповнений до зазначеної довжини до того, як буде відправлений по мережі.

Приклад передачі статичних решітчастих даних по комунікаційній мережі 1112 зв'язку проілюстровано на рис. 75, де пакет 1055 даних розбитий вузлом зв'язку $N_{0,0}$ у момент часу t_0 на чотири пакети різної довжини, зокрема пакет даних 1128A, що включає сегмент даних 1F, пакет даних 1128B, що включає сегмент даних 1C, пакет даних 1128C, що включає сегменти даних 1A і 1D, і пакет даних 1128D, що включає сегменти даних 1B і 1E. Зазначені сегменти даних можуть сполучатися з іншими сегментами даних з інших пакетів даних і ланцюжків повідомлень також різної довжини. Для ясності фрагменти даних з інших ланцюжків повідомлень навмисно не були наведені в ілюстрації.

Під час статичної передачі вміст пакета даних, тобто сегменти даних, що містяться в ньому, залишаються незмінними при проходженні мережі. Наприклад, пакет даних 1128A, що включає сегмент даних 1F, проходить по вузлах зв'язку послідовно від вузла $N_{0,0}$ до вузла зв'язку $N_{1,1}$, а потім до вузлів зв'язку $N_{2,1}$, $N_{3,2}$, $N_{3,3}$, $N_{4,3}$, і $N_{4,4}$, після чого збирається заново пакетами 1128B, 1128C і 1128D у кінцевому вузлі зв'язку $N_{f,f}$ для відтворення пакета даних 1055 у момент часу t_f . Аналогічним чином, пакет даних 1128C, що включає сегменти даних 1A і 1D, послідовно проходить через вузли зв'язку від вузла $N_{0,0}$ до вузла $N_{3,1}$, потім до вузла зв'язку $N_{2,3}$ і вузла зв'язку $N_{1,4}$, після чого збирається заново пакетами 1128A, 1128B і 1128D у кінцевому вузлі зв'язку $N_{f,f}$ у момент часу t_f . Під час статичної решітчастої передачі множинні пакети даних проходять через загальні сервери без змішування або взаємодії. Наприклад, пакети даних 1128A і 1128B проходять через вузол зв'язку $N_{2,1}$, пакети даних 1128B і 1128C проходять через вузол зв'язку $N_{2,3}$, а пакети даних 1128A і 1128D проходять через вузол зв'язку $N_{3,3}$, безперешкодно один для одного, без обміну вмістом або обміну сегментами даних.

Оскільки канали передачі даних можуть мати різну довжину та різні затримки проходження, деякі пакети даних можуть надходити на кінцевий вузол зв'язку $N_{f,f}$ раніше за інші. У таких випадках відповідно до цієї розробки пакети даних повинні тимчасово перебувати у вузлі зв'язку $N_{f,f}$ доти, поки не надійдуть інші зв'язані пакети даних. І хоча на рисунку видно, що остаточне складання та відновлення вихідного пакета даних 1055 відбувається у вузлі зв'язку $N_{f,f}$, на практиці остаточне складання пакета, тобто змішування, може відбуватися на такому пристрої, як настільний комп'ютер, ноутбук, мобільний телефон, планшет, ресивер, автомобіль, холодильник або інший апаратний пристрій, під'єднаний до мережі. Інакше кажучи, щодо решітчастої передачі немає ніякого розходження між вузлом зв'язку та пристроєм, підключеним до вузла зв'язку, тобто вузол зв'язку $N_{f,f}$ можна розглядати як настільний комп'ютер, а не істинний високопродуктивний сервер. З'єднання пристрою з розглядуваною хмарою SDNP, тобто з'єднання "останньої милі", докладніше відображено далі в цьому додатку.

Згадана вище статична маршрутизація може бути об'єднана з кожним із вищезгаданих методів SDNP, як описано, включаючи скремблювання, шифрування або їхні комбінації. Наприклад, на рис. 76 статична решітчаста маршрутизація з пакетами даних змінної довжини поєднується зі статичним скремблюванням. Як можна побачити, у момент часу t_1 пакет із дешифрованими даними 1055 конвертується в скремблований пакет незашифрованих даних 1130, який потім розділяється вузлом зв'язку $N_{0,0}$, після чого розділені пакети, змішані з небажаними даними, відправляються по мережі 1112. Маршрутизація аналогічна попередньому прикладу, за винятком того, що сегменти даних навмисно дезорганізовані та змішані з сегментами "сміттєвих" даних до маршрутизації. Наприклад, пакет даних 1132C, що включає сегменти даних 1D і 1A, розділені проміжним пакетом "сміттєвих" даних, обходить вузли зв'язку послідовно від вузла зв'язку $N_{0,0}$ до вузла зв'язку $N_{3,1}$, потім до вузлів $N_{2,3}$, $N_{3,2}$ і $N_{1,4}$ поки, нарешті, повторно не збирається пакетами 1128A, 1128B і 1128D у кінцевому вузлі зв'язку $N_{f,f}$ для відтворення пакета даних 1055 у момент часу t_f . Аналогічним чином пакет даних 1132D, що включає пакети даних 1E і 1B, у зворотному порядку проходить вузли зв'язку послідовно від вузла зв'язку $N_{0,0}$ до вузла зв'язку $N_{4,1}$, потім до вузлів зв'язку $N_{4,2}$, $N_{3,3}$ і $N_{2,4}$, після чого, нарешті, повторно збирається з пакетами 1128A, 1128B і 1128C у кінцевому вузлі зв'язку $N_{f,f}$ у момент часу t_f . У цьому кінцевому вузлі під час змішування виконується операція видалення для усунення "сміттєвих" даних для створення вихідних скремблованих даних 1130. Після дескремблювання вихідні дані 1055 відновлюються.

Для реалізації динамічного решітчастого обміну згідно з розробкою, описаною в цьому документі, пакети повинні оброблятися для зміни їхнього вмісту та напрямку в межах кожного вузла зв'язку, що обробляє пакет. Цей процес містить у собі об'єднання вхідних пакетів даних в один довгий пакет даних або, як можливий варіант, використання буфера даних, що утримує ті самі субпакети так, ніби був створений довгий пакет даних, а потім відбувся розподіл цих пакетів на різні комбінації та відправлення цих пакетів різним адресатам. Цей процес може

здіяти пакети змінної або фіксованої довжини, як описано вище. На рис. 77A показано елементи комунікаційної мережі SDNP, включаючи вузли зв'язку $N_{a,b}$, $N_{a,d}$, $N_{a,f}$ і $N_{a,h}$, усі відправляють у мережі "A" відповідні пакети даних змінної довжини 1128B, 1128D, 1128F і 1128H відповідно до вузла зв'язку $N_{a,j}$, що виконує операцію змішування 1089, збираючи пакети в короткий або довгий пакет даних 1055. Потім пакет 1055 розділяється з використанням операції розділення 1106 у вузлі зв'язку $N_{a,j}$ для створення нових пакетів даних 1135N, 1135Q і 1135S зі змінною довжиною, які відправляються у вузли зв'язку $N_{a,n}$, $N_{a,q}$ і $N_{a,s}$ відповідно. На вузол зв'язку $N_{a,v}$ дані або "сміттєві" дані не відправляються. У кожному разі довжина вхідних пакетів є змінною, і пакети можуть містити "сміттєві" дані або дані з інших повідомлень, переговорів або комюніке, які не представлені. Відповідно до зображення, комбінація операції змішування 1089 і операції розділення 1106 виконується за допомогою вузла зв'язку $N_{a,j}$ для полегшення динамічної решітчастої маршрутизації з використанням операції змішування та розділення даних 1148. Щойно розділені пакети 1135N, 1135Q, 1135S і 1135V (за умови, що останній містить "сміттєві" дані) і їхня маршрутизація визначаються або динамічними інструкціями, відправленими у вузол зв'язку $N_{a,j}$ мережею SDNP, або з використанням визначеного набору алгоритмів або команд при відсутності таких вхідних команд і сигналів керування, як описано нижче.

Для обробки вхідних пакетів, тобто їхнього змішування та подальшого розбивання на нові пакети різних комбінацій, вузол $N_{a,j}$ повинен до того, як надійдуть дані, одержувати інструкції про те, як йому варто ідентифікувати оброблювані пакети даних і що з ними робити. Ці інструкції можуть включати фіксовані алгоритми, що зберігаються локально як поділюваний секрет, тобто набір визначених алгоритмів або команд, або ж послідовність може бути визначена прямо в "динамічній" інструкції щодо командування та керування, відправлена вузлу заздалегідь, даних, з іншого сервера, який контролює маршрутизацію, але не на сервер, що передає дані. Якщо інструкції про те, що робити з вхідними даними, включені в сам потік даних, тобто частину носія або контенту, маршрутизацію називають "одноканальним" зв'язком. Якщо маршрутизація пакетів даних визначається іншим сервером і передається на медіа-сервер, маршрутизація даних називається "двоканальним" (можливо, і триканальним) зв'язком. Оперативні характеристики одно-, дво- і триканального зв'язку описані докладніше в додатку нижче.

Незалежно від того, яким чином надаються інструкції, медіавузол повинен розпізнавати вхідні пакети даних, щоб довідатися про інструкцію, що стосується певного пакета даних. Ця ідентифікуюча інформація або "тег" виконує роль поштового індексу або штрих-коду маршрутизації пакета для ідентифікації пакетів, які цікавлять. Однак вхідні пакети даних 1128B, 1128D, 1128F і 1128H, як показано на рис. 77A, являють собою тільки аудіо- або текстовий вміст пакета, а не розпізнавальні мітки. Процес використання позначених даних, присутніх у заголовку пакета, для ідентифікації кожного конкретного пакета даних і визначення того, як варто змішувати вхідні пакети даних, було описано раніше на рис. 67I. Конкретні приклади міток та інформації про маршрутизацію, що міститься в пакетах даних, обговорюються в додатку до даного документа. Як тільки вузол $N_{a,j}$ одержав інформацію про те, які пакети даних потрібно шукати і який алгоритм використовувати в операції змішування 1089 і операції розділення 1106, дані можуть оброблятися.

Еквівалент пакета даних фіксованої довжини для тієї самої операції показано на рис. 77B, де вузли зв'язку $N_{a,b}$, $N_{a,d}$, $N_{a,f}$ і $N_{a,h}$, усі в мережі "A" відправляють відповідні пакети даних фіксованої довжини 1150B, 1150D, 1150F і 1150H, відповідно, до вузла зв'язку $N_{a,j}$, що, у свою чергу, виконує операцію змішування та розділення 1148 для створення нових пакетів даних фіксованої довжини 1151N, 1151Q і 1151S, відправлених вузлом зв'язку $N_{a,n}$, $N_{a,q}$ і $N_{a,s}$ відповідно. Дані або "сміттєві" дані 1151V не передаються на вузол зв'язку $N_{a,v}$. У кожному випадку довжина вхідних пакетів є фіксованою і обов'язково включає "сміттєві" дані як вміст або дані з іншого ланцюга повідомлень або комюніке, які не показані, для підтримки фіксованої довжини пакетів даних, тобто включати задану кількість сегментів даних.

Взаємозв'язок серверів, як описано в мережевому протоколі Рівня 3, містить безліч з'єднань, причому кожен вихід комунікаційного вузла підключений до входу іншого комунікаційного вузла зв'язку. Наприклад, як показано на рис. 77C, виходи комунікаційного вузла $N_{a,b}$, що виконує операцію змішування та розділення 1149B, з'єднані з входами комунікаційних вузлів $N_{a,j}$, $N_{a,q}$, $N_{a,v}$ і $N_{a,f}$. Виходи вузла $N_{a,q}$, що виконує операцію змішування та розділення 1149Q, з'єднані з входами вузлів $N_{a,b}$, $N_{a,j}$ і $N_{a,f}$ та іншого комунікаційного вузла, не показаного на ілюстрації. Подібним чином виходи комунікаційного вузла $N_{a,f}$, що виконує операцію змішування та розділення 1149F, з'єднані з входами комунікаційних вузлів $N_{a,q}$, $N_{a,j}$ і $N_{a,v}$ та іншого комунікаційного вузла, не показаного на ілюстрації; виходи комунікаційного вузла $N_{a,j}$, що виконує операцію змішування та розділення 1149J, з'єднані з входами комунікаційних вузлів $N_{a,q}$

і $N_{a, v}$ разом з іншими вузлами зв'язку, не показаними на ілюстрації; і виходи вузла $N_{a, v}$, які виконують операції змішування та розділення 1149V, з'єднані з входами комунікаційних вузлів $N_{a, f}$ та інших комунікаційних вузлів, не показаних на ілюстрації.

Оскільки з'єднання вхід-вихід є описами мережі, а не просто з'єднаннями або схемами рівня 1 РНУ, ці мережні з'єднання між пристроями можуть бути встановлені або розірвані на тимчасовій основі для будь-якого пристрою, що підтримує з'єднання рівня 1 РНУ і канал передачі даних рівня 2 для вищезгаданої мережі або хмари. Крім того, оскільки з'єднання являють собою можливі шляхи передачі даних по мережі та не є фіксованими, постійними електричними контурами, то факт, що вихід комунікаційного вузла $N_{a, b}$ з'єднаний із входом комунікаційного вузла $N_{a, q}$, а вихід комунікаційного вузла $N_{a, q}$ з'єднаний з введенням комунікаційного вузла $N_{a, b}$, не створює зворотного зв'язку або перегону фронтів, як це було б в електричних колах.

Насправді будь-який комп'ютер, електрично підключений до мережі, може бути доданий або вилучений як комунікаційний вузол, що динамічно та на тимчасовій основі використовує програмне забезпечення. Підключення комп'ютера до мережі пов'язане з "реєстрацією" комунікаційного вузла з сервером імен або будь-яким сервером, що виконує функцію сервера імен. Як описано в розділі "Основні положення" цього додатка, в Інтернеті сервер імен являє собою мережу комп'ютерів, що визначають свою електронну ідентифікацію як інтернет-адресу, що використовує формати IPv4 або IPv6. Найважливіший сервер імен Інтернету – це глобальні сервери DNS або доменних імен. Деякі комп'ютери не використовують справжні Інтернет-адреси, а замість цього мають адреси, призначені NAT або транслятором мережевої адреси.

Описувана захищена динамічна мережа та протокол аналогічним чином використовують функцію сервера імен для відстеження кожного пристрою в мережі SDNP. Щоразу, коли запускається вузол зв'язку SDNP або, говорячи комп'ютерною мовою, щоразу, коли програмне забезпечення вузла SDNP завантажується, новий пристрій динамічно реєструється на сервері імен мережі, у такий спосіб інші вузли SDNP дізнаються, що він підключений до мережі та доступний для зв'язку. У системі триканального зв'язку сервери імен SDNP відділені від серверів, які використовуються для керування та контролю, тобто сигнальних серверів, і від медіасерверів, що несуть фактичний комунікаційний зміст. У системі одноканального зв'язку один набір серверів повинен виконувати як завдання сервера імен, так і маршрутизацію керування та переносити контент. Таким чином, три типи систем SDNP, описаних тут, одноканальні, двоканальні та триканальні, відрізняються серверами, використовуваними для виконання функцій перенесення, сигналізації та іменування. В одноканальних системах сервери вузлів зв'язку виконують усі три функції; у двоканальних системах функції сигналізації та іменування відділені від функції перенесення та виконуються серверами сигналізації; а в триканальних системах функція іменування відділена від функцій перенесення та сигналізації і виконується серверами імен. На практиці дана мережа SDNP не обов'язково повинна бути однорідною і може бути підрозділена на одноканальні, двоканальні та триканальні ділянки.

Будь-який новий вузол зв'язку SDNP, підключений до мережі, реєструється, повідомляючи серверу імен свою SDNP-адресу. Ця адреса є не Інтернет-адресою, відомою тільки мережі SDNP, і до неї не можна отримати доступ через Інтернет, оскільки, як і адреса NAT, адреса SDNP для мережі Інтернет не має змісту, незважаючи на те, що вона відповідає Інтернет-протоколу. По суті, зв'язок із використанням описуваної захищеної динамічної мережі та протоколу являє собою "анонімний" зв'язок, оскільки IP-адреси не пізнавані в Інтернеті, і тільки остання адреса SDNP і наступна адреса SDNP, тобто наступний пункт призначення пакета, є в даному пакеті.

Важливою реалізацією мережі SDNP є її здатність автоматично модулювати загальну доступну смугу хмари, як тільки трафік збільшується або зменшується в будь-який час доби. До мережі автоматично додаються додаткові вузли зв'язку SDNP, як тільки трафік збільшується, і вилучаються зайві при його зменшенні, знижуючи витрати мережі без шкоди для стабільності або продуктивності.

Ця особливість означає, що смуга пропускання та довжина мережі SDNP, описані тут, також можуть бути динамічно скоректовані для мінімізації експлуатаційних витрат, тобто витрат за невикористовувані цикли обчислень на невикористовуваному вузлі вилучаються, зі збереженням водночас можливостей збільшення продуктивності в міру необхідності. Переваги варіанта мережі SDNP, реалізованого програмним забезпеченням або "програмним комутатором", різко контрастують із фіксованим устаткуванням і високою вартістю комунікаційних мереж, реалізованих апаратним забезпеченням з пакетною комутацією, які усе ще поширені сьогодні.

У мережі, реалізованій програмним комутатором, будь-який комунікаційний вузол, завантажений програмним забезпеченням мережі зв'язку SDNP і підключений до мережі або Інтернет, може бути за необхідності доданий в SDNP, як показано на графі мережі на рис. 77D, де комп'ютерні сервери 1149D, 1149B, 1149F, 1149Q, 1149H, 1149N, 1149J, 1149S і 1149V можуть бути додані як відповідні комунікаційні вузли $N_{a,q}$, $N_{a,d}$, $N_{a,b}$, $N_{a,f}$, $N_{a,q}$, $N_{a,h}$, $N_{a,n}$, $N_{a,j}$, $N_{a,s}$ і $N_{a,v}$ відповідно, як тільки виникає потреба у трафіку у вузлі або у зв'язку по його з'єднаннях. Таким чином, кожне посилання в хмарі SDNP може розглядатися як постійне фізичне з'єднання рівня 1 PHY з відповідною лінією передачі даних рівня 2 у сполученні з мережевим з'єднанням рівня 3, що встановлюється тільки тоді, коли SDNP за необхідності запускає, тобто активує, новий комунікаційний вузол.

Таким чином, хмара SDNP на основі програмного комутатора є адаптивною і динамічною, змінною за необхідності. На відміну від рівноправних мереж, де дані передаються через будь-який пристрій або комп'ютер, навіть із невідомою пропускну здатністю та надійністю, кожен комунікаційний вузол SDNP є попередньо оціненим пристроєм, завантаженим програмним забезпеченням SDNP, реалізованим із програмним комутатором, і повністю придатним для під'єднання до хмари SDNP і перенесення даних з використанням запропонованого протоколу захищеного зв'язку, що включає інформаційний вміст (такий як поділюваний секрет) плюс синтаксис, наприклад, певний формат заголовка. Розподілені (спільні) секрети описують алгоритми, генератори початкових станів, методи скремблювання, методи шифрування та методи змішування, але не визначають формату всього пакета даних SDNP. Параметри безпеки, тобто параметри, використовувані в певний час і для конкретних повідомлень, є типом спільних секретів, але розподілені (спільні) секрети також включають увесь перелік алгоритмів, навіть ті з них, які не використовуються. Оскільки програмне забезпечення зашифроване, а алгоритми та розподілені (спільні) секрети обробляються динамічно, навіть у випадку, якщо код SDNP розміщений у загальнодоступній хмарі, такий як Amazon або Microsoft, оператори сервера не мають можливостей для моніторингу вмісту трафіку даних на комунікаційному вузлі SDNP, окрім загального обсягу даних, що транспортуються.

Нові клієнти SDNP, такі як мобільний телефон, планшет або ноутбук, також реєструються автоматично сервером імен або шлюзом SDNP при кожному включенні як природне розширення динамічної мережі. Таким чином, не тільки хмара SDNP, але й та кількість клієнтів, доступних для підключення, налаштовуються автоматично, чітко відображаючи кількість підключених до мережі та активних користувачів у будь-який момент часу.

Скрембльована або зашифрована решітчаста маршрутизація. Для підтримки динамічних автономних можливостей кожен вузол зв'язку SDNP виконує запроповану комбінацію змішування та розділення даних, скремблювання та дескремблювання, шифрування та дешифрування паралельно для одночасної підтримки багаторазової передачі інформації, комуніке та безпечних сеансів. При включенні програмного комутатора зв'язку SDNP усі реалізовані функції та послідовність цих операцій можуть бути повністю сконфігуровані за допомогою програмних інструкцій, позначених у вигляді спільних секретів, які здійснюються пакетом даних, або позначених паралельним сигнальним каналом для керування та контролю, окремого та відмінного від вузла зв'язку SDNP, використовуваного для перенесення носія. Незважаючи на те, що можливо велика кількість перестановок і комбінацій, приклади, показані тут, призначені для представлення гнучкості зв'язку, заснованої на SDNP, а не для обмеження застосування різних функцій SDNP, описаних для конкретної послідовності етапів обробки даних. Наприклад, скремблювання може передувати або йти за змішуванням або розділенням, шифрування може відбуватися на початку, наприкінці або в середині і т.д.

Одна з таких операцій, а саме рескрембльована операція змішування та розділення 1155, показана на рис. 78A, виконує послідовність конкретних функцій SDNP за допомогою безлічі вхідних пакетів даних з вузлів зв'язку $N_{a,b}$, $N_{a,d}$, $N_{a,f}$ і $N_{a,h}$. Останні включають операцію дескремблювання 928, що виконується по кожному вхідному пакету даних, змішуючи та розділяючи пакети даних за допомогою операції змішування та розділення 1148, з наступним рескремблюванням нових пакетів даних за допомогою операції скремблювання 926 і пересиланням даних пакетів у решітчасту мережу. Як показано на рис. 78B, послідовність здійснення безлічі незалежних операцій дескремблювання 928 на кожному вході, після яких необхідна операція змішування 1089, разом охоплюють операцію 1156A "дескрембльованого змішування ланкових входів". Для зручності послідовність може бути представлена символічно за допомогою операції дескремблювання та змішування 1161.

Зворотною операцією дескремблювання та змішування є операція "розділення та скремблювання" 1156B для ланкових виходів, показана на рис. 78C, що містить послідовність поділу пакета даних з операцією розділення 1106, при якій потрібне виконання множинних

незалежних операцій скремблювання 926 для кожного виходу. Для зручності послідовність може бути представлена символічно за допомогою операції розділення та скремблювання 1162. Як показано на рис. 78D, послідовна комбінація операції 1156A двокомбінованого дескремблюваного змішування ланкових входів, при якій необхідна операція 1156B розділення та скремблювання для ланкових виходів, включає операцію "повторного скремблювання та реміксу" для решітчастої передачі даних, символічно показану як операція 1163.

Застосування вищезгаданої операції дескремблюваного змішування ланкових входів 1161, при якій необхідна операція розділення та скремблювання 1162 для ланкових виходів, показана на рис. 79A. Тут входи пакетів даних фіксованої довжини 1157B, 1157D, 1157F, і 1157H відповідних вузлів зв'язку $N_{a, b}$, $N_{a, d}$, $N_{a, f}$ і $N_{a, h}$ обробляються за допомогою операції дескремблюваного змішування ланкових входів 1156 у вузлі зв'язку $N_{a, j}$ для формування довгого пакета даних 1160. Незважаючи на те, що операція 1156 включає функціональні можливості для незалежного дескремблювання вхідних пакетів даних перед змішуванням, цей крок не потрібен і тому пропускається, оскільки входи пакетів даних фіксованої довжини 1157B, 1157D, 1157F і 1157H не скремблюються. Довгий пакет даних 1160 потім обробляється шляхом операції розділення та скремблювання 1162, що призводить до змішування, скремблюваних пакетів даних 1158N, 1158Q, 1158S і 1158V, відправлених на відповідні вузли зв'язку $N_{a, n}$, $N_{a, q}$, $N_{a, s}$ і $N_{a, v}$ для решітчастої передачі даних.

Аналогічна операція скремблюваного змішування та розділення для решітчастої передачі пакетів даних фіксованої довжини представлена на рис. 79B для вхідних пакетів даних 1165B, 1165D, 1165F і 1165H, які скремблюються. Ці пакети даних містять у собі "сміттєві" сегменти даних, на що вказують сегменти даних без ідентифікаційного номера. Операція дескремблювання та змішування 1161 у вузлі зв'язку $N_{i, j}$ потім створює довгий пакет 1166, коротший, ніж у попередньому прикладі, тому що "сміттєві" пакети даних навмисне видаляються. В альтернативному варіанті здійснення винаходу "сміттєві" пакети можуть бути збережені. Довгий пакет 1166 потім обробляється за допомогою операції розділення та скремблювання 1162 для створення кратних виходів пакета даних 1165N, 1165Q, 1165S і 1165V, відправлених на відповідні вузли зв'язку $N_{a, n}$, $N_{a, q}$, $N_{a, s}$ і $N_{a, v}$ для решітчастої передачі даних. У цих пакетах даних "сміттєві" дані були повторно вставлені для заповнення пакетів даних заданою кількістю сегментів даних. Незважаючи на те, що в цілому переважніше та простіше введення "сміттєві" сегменти даних наприкінці пакета даних, як показано пакетами даних 1165N і 1165S, якщо пропонує так алгоритм, "сміттєві" пакети необов'язково можуть бути введені в інше місце в пакеті даних, наприклад, у перший слот, як показано в пакеті даних 1165V.

Приклад динамічної решітчастої передачі даних зі статичним скремблюванням паралельно комунікаційній мережі відповідно до винаходу показано на рис. 80, що включає мережу взаємозалежних комп'ютерних серверів 1118, які працюють на програмному забезпеченні зв'язку SDNP. Вузол зв'язку $N_{0,0}$ виконує операцію скремблювання та розділення 1162, вузол зв'язку $N_{i, f}$ виконує операцію змішування та дескремблювання 1161, а всі інші вузли виконують операцію рескремблювання та повторного змішування 1163. Незважаючи на те, що в показаному прикладі кожен сервер виконує тільки одну спеціальну операцію, само собою зрозуміло, що програмне забезпечення SDNP, встановлене на всіх комп'ютерних серверах 1118, здатне виконувати будь-яку функцію SDNP, у міру необхідності, включаючи операцію скремблювання та розділення 1162, операцію дескремблювання та змішування 1161, операцію рескремблювання та повторного змішування 1163 та інші, як зазначено вище.

Під час операції вхідний пакет даних 1055 спочатку скремблюється вузлом зв'язку $N_{0,0}$ у момент часу t_1 за допомогою операції скремблювання та розділення 1162. При цьому створюється скремблюваний пакет даних 1130, що розділяється на чотири пакети різної довжини: пакет даних 1170A, що включає сегмент даних 1F і асоційований сегмент "сміттєвих" даних у першому слоті; пакет 1170B, що включає сегмент даних 1C; пакет даних 1170C, що включає сегменти даних 1A і 1D у зворотному порядку, та пакет даних 1170D, що включає сегменти даних 1B і 1E у порядку зростання. Показані сегменти даних можуть бути об'єднані з іншими сегментами даних з інших пакетів даних і каналів зв'язку різної довжини, де сегменти даних з інших каналів зв'язку були навмисно виключені з ілюстрації для ясності. Відомо, що проходить час, поки пакети даних проходять через мережу і їхній вміст розділяється та повторно змішується. Проте, для ясності, час було навмисно вилучено з зображення, за винятком приблизного часу, показаного на початку та наприкінці процесу зв'язку.

Під час активної решітчастої передачі даних вміст пакета даних, його сегменти даних змінюються у міру проходження мережі. Наприклад, пакет даних 1170A, що включає сегмент "сміттєвих" даних і сегмент даних 1F, проходить вузли зв'язку послідовно. Спочатку від вузла

зв'язку $N_{0,0}$ до вузла зв'язку $N_{1,1}$ і потім до вузла зв'язку $N_{2,1}$, де він змішується з пакетом даних 1170B, що включає сегмент даних 1C, для формування пакета даних 1171A. Останній містить послідовність сегментів даних 1C, 1F і сегмент "сміттєвих" даних, що відправляється на вузол зв'язку $N_{1,2}$ і потім на вузол зв'язку $N_{2,3}$. Упродовж того ж періоду часу пакет даних 1170C, що
 5 включає послідовність сегментів даних 1D, 1A, передається з вузла зв'язку $N_{0,0}$ у вузол зв'язку $N_{3,1}$. Тут він пересилається без змін, так само як пакет даних 1171C у вузол зв'язку $N_{3,2}$. У рамках операції змішування та розділення, виконуваної вузлом зв'язку $N_{3,1}$, другий пакет даних 1171B, що включає повністю "сміттєві" дані без вмісту, генерується та відправляється на вузол зв'язку $N_{2,1}$. Причина маршрутизації повністю "сміттєвого" пакета, позбавленого контенту, є
 10 двосторонньою причиною – по-перше, заплутати кіберпіратів шляхом виведення декількох пакетів даних з вузла зв'язку $N_{3,1}$, а по-друге, одержати оновлені внутрішньомережеві дані затримки від невикористовуваних посилань або маршрутів.

При вході у вузол зв'язку $N_{3,2}$ пакет даних 1171C розбивається на два пакети даних: пакет даних 1172C, що включає сегмент даних 1D, і який направляється на вузол зв'язку $N_{3,3}$; пакет
 15 даних 1172B, що включає сегмент даних 1A; і провідний сегмент даних, що включає "сміттєві" дані, які направляються на вузол зв'язку $N_{2,3}$. Після досягнення сервера $N_{2,3}$ пакет даних 1172B змішується з вхідним пакетом 1171A, а потім знову розбивається на пакет 1173A, що включає сегменти даних 1F і 1A і посилає на вузол зв'язку $N_{1,4}$ де сегменти кінцевих "сміттєвих" даних додаються для формування пакета даних 1174A, що направляється на вузол зв'язку $N_{f,f}$ у
 20 момент t_{14} . Одночасно в результаті операції розділення, виконуваної у вузлі зв'язку $N_{2,3}$, пакет даних 1173B відправляється далі на вузол зв'язку $N_{3,4}$, де сегмент кінцевих "сміттєвих" даних додається до сегмента даних 1C перед відправленням його в останній вузол зв'язку $N_{f,f}$ у момент t_{16} (час не показаний).

При цьому пакет даних 1170D, що включає сегменти даних 1E і 1D, переноситься з вузла зв'язку $N_{0,0}$ на вузол зв'язку $N_{4,1}$ і на вузол зв'язку $N_{4,2}$, де він рескремблюється, формуючи пакет
 25 даних 1172D, що включає сегменти даних 1B і 1E у зворотному порядку. При вході у вузол зв'язку $N_{3,3}$ пакет даних 1172D змішується з пакетом даних 1172C, а потім знову розбиваються, формуючи пакети даних 1173C і 1173D. Пакет даних 1173C, що включає сегмент даних 1B, направляється на вузол зв'язку $N_{2,4}$, звідки він пересилається на кінцевий сервер $N_{f,f}$ у момент
 30 t_{15} , як пакет даних 1174B. Незважаючи на те, що пакети даних 1173C і 1174B ідентичні, кожен містить тільки сегмент даних 1B, тобто пакет 1173C фактично не змінюється вузлом зв'язку $N_{2,4}$. Це узгоджується з часом t_{15} і його відповідним станом, включаючи початкові числа, ключі, розподілені (спільні) секрети, алгоритми і т.д. у вузлі зв'язку $N_{2,4}$. Інший пакет даних, тобто пакет даних 1173D з вихідним вузлом зв'язку $N_{3,3}$, потім направляється на вузол зв'язку $N_{4,3}$ і вузол зв'язку $N_{4,4}$, де сегмент проміжних "сміттєвих" даних вставляється між сегментами даних 1E і 1D
 35 для створення пакета даних 1174D у момент t_{17} з відповідним станом 1137. Пакети даних 1174A, 1174B, 1174C, і 1174D, кожен з яких формується в різних станах і в різний час (особливо в моменти часу t_{14} , t_{15} , t_{16} , і t_{17}), дескремблюються та змішуються у вузол зв'язку $N_{f,f}$. При цьому вони використовують операцію дескремблювання та змішування 1161 для повторного
 40 створення вихідного дескрембльованого пакета даних 1055 у момент часу t_f . Усі вузли фіксують, що робити для того, щоб обробити вхідний пакет даних за станом пакета або за іншим ідентифікатором, що відповідає набору розділених секретів, відомих вузлу, або тому, що окремий сервер, який називається сигнальним сервером до вузла, априорі фіксує, що робити, коли надходить окремий пакет.

Так і у випадку статичної решітчастої передачі даних, в активній решітчастій передачі даних, шляхи даних можуть бути різної довжини та виявляти різні затримки поширення. У результаті
 45 чого пакети даних можуть надійти на останній вузол зв'язку $N_{f,f}$ раніше за інші. У таких випадках, відповідно до даного винаходу, пакети даних повинні тимчасово зберігатися у вузлі зв'язку $N_{f,f}$, поки не надійдуть інші зв'язані пакети даних. І хоча зображення показує, що остаточне встановлення та відновлення вихідного пакета даних 1055 відбувається у вузлі зв'язку $N_{f,f}$, на
 50 практиці остаточне повторне встановлення пакета може відбуватися у пристрої, такому як настільний комп'ютер, ноутбук, стільниковий телефон, планшет, телеприставка, автомобіль, холодильник або інший апаратний пристрій, підключений до мережі. Інакше кажучи, щодо решітчастої передачі даних немає жодної розбіжності між вузлом зв'язку та пристроєм,
 55 підключеним до вузла зв'язку, тобто вузол зв'язку $N_{f,f}$ можна розглядати настільним комп'ютером замість істинного сервера з великою пропускнуою здатністю. З'єднання пристрою з розглядуваною хмарою SDNP, , тобто кінцевим з'єднанням, докладніше розглядається далі в цьому додатку.

Як зазначалося раніше, згадана вище динамічна маршрутизація може бути об'єднана з
 60 одним або декількома згаданими методами SDNP, як описано, включаючи скремблювання,

шифрування або їхні комбінації. Одна така зашифрована операція змішування та розділення 1180, показана на рис. 81А, виконує послідовність спеціальних операцій SDNP на множинних вхідних пакетах даних з вузлів зв'язку $N_{a, b}$, $N_{a, d}$, $N_{a, f}$ і $N_{a, h}$, що включають операції дешифрування 1032, які виконуються на кожному вхідному пакеті даних, змішуючи та розділяючи пакети даних за допомогою операції змішування та розділення 1148, яка йде за повторним шифруванням нових пакетів даних за допомогою операції шифрування 1026, і пересилаючи ці пакети по решітчастій мережі. Відповідно до зображення, вхідні пакети даних раніше були зашифровані і містять нерозбірливі зашифровані пакети 1181А, 1183А та інші, які не показані. Ключі шифрування, необхідні для дешифрування зашифрованих входів, специфічні для часу, стану та алгоритмів шифрування, використовуються для того, щоб кожен вхідний пакет проходив операцію дешифрування 1032 до виконання дешифрування, як розділений секрет, ключі, наявні у незашифрованому пакеті даних, відправлені специфічним пакетом даних або комюніке, або ключі, які надаються через інші канали зв'язку. Як описано нижче, ключі можуть бути симетричними або асиметричними. Тема обміну ключами розглядається нижче.

Після дешифрування пакети даних стають незашифрованими пакетами 1182А, 1184А та іншими, які не показані, потім змішуються вузлом зв'язку $N_{a, j}$ у довгий пакет 1185, що також включає незашифрований текст, і потім розділяються на нові незашифровані пакети 1182В, 1184В та інші, які не показані. Використовуючи нові різні ключі шифрування на основі конкретного часу або стану, пакети даних потім зашифровуються для створення нових зашифрованих пакетів 1181В, 1183В та інших, які не показані, відправлених на інші вузли зв'язку. Як показано на рис. 81В, послідовність виконання кратних незалежних операцій дешифрування 1032 на кожному вході, за яким йде операція змішування 1089, разом складає "дешифроване змішування решітчастих входів", символічно представлене операцією дешифрованого змішування 1090. Операція "розділення та шифрування" для решітчастих виходів, показана на рис. 81С, включає послідовність розділення пакета даних за допомогою операції розділення 1106, за якою йде створення кратних незалежних операцій шифрування 1026 для кожного виходу. Для зручності послідовність може бути представлена символічно операцією розділення та шифрування 1091.

На рис. 82А показано приклад повторного шифрування, рескремблювання та повторного розділення пакетів даних із кратних вузлів зв'язку $N_{a, b}$, $N_{a, d}$, $N_{a, f}$ і $N_{a, h}$ для решітчастої передачі даних відповідно до винаходу. При використанні операції повторного шифрування, рескремблювання, змішування та розділення 1201 на вхідні пакети даних, що надходять у вузол зв'язку $N_{a, j}$, кожен вхідний пакет даних дешифрується операцією дешифрування 1032, дескремблюється операцією дескремблювання 928, змішується операцією змішування 1089, і згодом розділяється на множинні нові пакети даних шляхом операції розділення 1106. Кожен пакет даних потім незалежно знову скремблюється за допомогою операції скремблювання 926, знову зашифровується за допомогою шифрування 1026 і потім посилається далі за допомогою решітчастої мережі. Відповідно до зображення, вхідні пакети даних попередньо були зашифровані і містять нерозбірливий зашифрований текст 1194А, 1197А та інші, які не показані.

Інформація про час і стан, спільні секрети, числові початкові значення, алгоритми і ключі дешифрування необхідна для дескремблювання та дешифрування зашифрованих входів, специфічних для часу, стану та алгоритмів. Вони використовуються для того, щоб кожен вхідний пакет проходив операцію дешифрування 1032 до проведення операції дешифрування та дескремблювання 928, як спільні секрет, ключі або числові початкові значення, наявні у незашифрованому пакеті даних, відправлені специфічним пакетом даних або комюніке, або ключі і числові початкові значення, доставлені іншим каналом зв'язку. Ключі можуть бути симетричними або асиметричними. Тема обміну ключами та доставка числових початкових значень розглядається нижче. Усі вузли фіксують, що робити, щоб обробити вхідний пакет даних за станом пакета або за іншим ідентифікатором, коли початкове значення відповідає набору спільних секретів, відомих вузлу, або окремому серверу, який називається сигнальним сервером, до вузла, що фіксує, що робити, коли надходить окремий пакет.

Після дешифрування незашифровані пакети 1195А, 1198А та інші, які не показані, потім дескремблюються за допомогою операції дескремблювання 928 для створення відповідних дескрембльованих незашифрованих пакетів 1196А, 1199А та інших, які не показані. При операції змішування 1089 дескрембльовані незашифровані пакети змішуються вузлом зв'язку $N_{a, j}$ у довгий пакет 1220, що згодом розділяється на нові дескрембльовані незашифровані пакети 1196В, 1199В та інші, які не показані, під час операції розділення 1106. Потім скремблюють заново шляхом операції скремблювання 926 за допомогою нових початкових чисел, що відповідають поточному часу або стану для формування скрембльованих незашифрованих пакетів 1195В, 1198В та інших, які не показані. Використовуючи нові, різні

ключі шифрування на основі цього конкретного часу або стану, пакети даних знову зашифровуються шляхом операцій шифрування 1026 для формування нових зашифрованих текстів 1194B, 1197B та інших, які не показані, і потім відправляються на інші вузли зв'язку.

Відповідно до опису даного винаходу, комунікаційна система SDNP може містити будь-яку послідовність шифрування, скремблювання, змішування, розділення, дескремблювання та дешифрування. Принаймні теоретично, якщо виконується послідовність відбувається у відомій послідовності, математично описаній у вигляді функцій $y = H\{G[F(x)]\}$, де внутрішня функція F виконується першою та найвіддаленіша функція H виконується останньою, а потім у порядку відновлення вихідних даних x антифункція повинна виконуватися у зворотній послідовності, де H-1 виконується першою та F-1 виконується останньою, тобто $x = F^{-1}\{G^{-1}[H^{-1}(y)]\}$. Дана послідовність операції "першим прийшов, останнім обслужений" повинна скасовувати зміни та відновлювати вихідний вміст у тому випадку, якщо в ході процесу не видаляються або не вставляються дані в пакети. Якщо дані видаляються або вставляються в пакети, скремблований або зашифрований файл засмічений і не може бути відновлений. Наприклад, змішування даних, зашифрованих шляхом різних шифрувальних методів вихідних даних, які не можуть бути незашифрованими без первісного відновлення вихідних компонентів. Одна ключова перевага динамічного решітчастого зв'язку з використанням передачі даних SDNP – затінення всього вмісту шляхом динамічного змішування, розділення та перемаршрутизації декількох сеансів зв'язку губиться, якщо даний вузол зв'язку не має можливості змішувати або розділяти пакети у міру необхідності.

Тому один варіант здійснення зв'язку SDNP забезпечує незалежне скремблювання та шифрування на пакетах даних, що виходять із окремих виходів вузлів зв'язку, а не для змішування пакетів даних до операцій скремблювання та шифрування. Відповідно, якщо пакети даних, що входять у вузол зв'язку, зашифровуються, скремблюються, то вони повинні бути незалежно дескрембловані та не зашифровані до змішування, тобто до формування довгого змішаного пакету. Таким чином, краща послідовність операцій для вхідних пакетів полягає в послідовному дешифруванні, дескремблюванні та змішуванні вхідних даних на кожному вході у вузол зв'язку або в альтернативній послідовності з метою дескремблювання, дешифрування та змішування вхідних даних.

Перший випадок показано на рис. 82B, де операція дешифрування, дескремблювання та змішування ланкових виходів схематично зображена операцією DUM 1209 і символічно операцією DUM 1210, включає незалежне формування для кожного входу послідовності операції дешифрування 1032, операції дескремблювання 928, і потім змішування кінцевих пакетів даних шляхом операції змішування 1089. Окремі перемикачі 1208A і 1208B, присутні на кожному вході, використовуються для переадресації, за необхідності, пакетів даних навколо однієї з операцій 1032 або однієї з операцій дескремблювання 928, відповідно. Наприклад, якщо обидва перемикачі на певному вході відкриті, тоді всі пакети даних повинні пройти через супутню операцію дешифрування 1032 і операцію дескремблювання 928. Після цього пакет даних буде неодмінно дешифрований і дескремблований. Якщо обидва перемикачі закриті, операції є "укороченими", і дані не обробляються ні операцією дешифрування 1032, ні операцією дескремблювання 928, тобто дані, які направляються на операцію змішування 1089, не змінюються.

Якщо перемикач 1208A закритий і перемикач 1208B відкритий, тоді дані направляються на операцію дешифрування 1032, але проходять через операцію дескремблювання 928, що означає, що вхідний пакет даних буде дескремблований, але не дешифрований. З іншого боку, якщо перемикач 1208A відкритий і перемикач 1208B закритий, то дані проходять через операцію дешифрування 1032, але будуть спрямовані убік операції дескремблювання 928, що означає, що вхідні пакети даних будуть дешифровані, але не дескрембловані. Тому що операції дешифрування 1032 і операції дескремблювання 928 зазвичай реалізуються в програмному забезпеченні, що не має фізичних перемикачів, які відводять сигнал. Перемикачі 1208A і 1208B символічно представляють операцію програмного забезпечення. Зокрема, якщо перемикач, паралельний операції, відкритий, то застосовне програмне забезпечення виконує операцію, і якщо перемикач, паралельний операції, закритий, то застосовне програмне забезпечення не виконує операцію, а просто проходить вхід і вихід без змін. Мовою електроніки, функція "закривається" закритим перемикачем, отже сигнал залишається неопрацьованим. Комбінації підсумовано в наступній таблиці істинності, де перемикач 1208A одночасно з операцією дешифрування 1032 відомий як перемикач A і перемикач 1208B одночасно з операцією дескремблювання 928 відомий як перемикач B.

Перемикач А	Перемикач В	Дешифрування	Дескремблювання	Вплив пакета даних
Відкритий	Відкритий	Так	Так	Дешифрування потім дескремблювання
Закритий	Відкритий	Немає	Так	Тільки дескремблювання
Відкритий	Закритий	Так	Немає	Тільки дешифрування
Закритий	Закритий	Немає	Немає	Пакет даних не змінюється

Зворотна функція, а саме операція розділення, скремблювання та шифрування, показана на рис. 82С схематично операцією SSE 1209 і символічно операцією SSE 1213, включає поділ операцією розділення 1106, за якою йде незалежно здійснювана операція дескремблювання 926 одразу за операцією шифрування 1026. Перемикачі 1211В і 1211А, наявні на кожному вході, використовуються для відведення, за необхідності, пакетів даних на операцію скремблювання 926 або операцію шифрування 1026, відповідно. Наприклад, якщо обидва перемикачі 1211В і 1211А на певному вході відкриті, тоді всі пакети даних повинні проходити та повинні оброблятися операцією скремблювання 926 і операцією шифрування 1026. Після цього пакет даних буде неодмінно дешифровуватися та скремблюватися. Якщо обидва перемикачі закриті, то операції "замикаються" і дані проходять через перемикачі 1211В і 1211А, і не обробляються ні операцією скремблювання 926, ні операцією шифрування 1026. Це означає, що дані, які проходять через операцію розділення, на виході є незмінними.

Якщо перемикач 1211В закритий і перемикач 1211А відкритий, то дані будуть відведені від операції скремблювання 926, але будуть оброблені операцією шифрування 1026. Це означає, що вихідний пакет даних буде зашифрований, але не скремблований. В іншому випадку, якщо перемикач 1211В відкритий і перемикач 1211А закритий, то дані будуть оброблені за допомогою операції скремблювання 926, але будуть відведені від операції шифрування 1026. Це означає, що вихідні пакети даних будуть скрембловані, але не зашифровані.

Як зазначено вище, оскільки операції скремблювання 926 і операції шифрування 1026 зазвичай реалізуються в програмному забезпеченні, яке не має фізичних перемикачів, що відводять сигнал, то перемикачі 1211В і 1211А символічно представляють операцію програмного забезпечення. Зокрема, якщо перемикач, паралельний операції, відкритий, то застосовне програмне забезпечення виконує операцію; і якщо перемикач, паралельний операції, закритий, то застосовне програмне забезпечення не виконує операцію, а просто проходить вхід і вихід без змін. Мовою електроніки, функція "замикається" закритим перемикачем, отже сигнал залишається неопрацьованим. Комбінації підсумовано в наступній таблиці істинності, де перемикач 1211В одночасно з операцією скремблювання 926 відомий як перемикач В і перемикач 1211А одночасно з операцією шифрування 1026 відомий як перемикач А.

Перемикач В	Перемикач А	Скремблювання	Шифрування	Вплив пакета даних
Відкритий	Відкритий	Так	Так	Скремблювання, потім шифрування
Закритий	Відкритий	Немає	Так	Тільки шифрування
Відкритий	Закритий	Так	Немає	Тільки скремблювання
Закритий	Закритий	Немає	Немає	Пакет даних не змінюється

Комбінація множинного входу DUM 1209 і множинного виходу SSE 1212 формує універсальний елемент для забезпечення безпечного зв'язку відповідно до винаходу, який називається медіавузлом SDNP 1201, показаним на рис. 83А. Відповідно до зображення, дані, що надходять на будь-які множинні входи, можуть бути спочатку дешифровані операцією дешифрування 1032, або операція дешифрування 1032 може бути відведена. Пакет даних може бути потім дескремблований операцією дескремблювання 928, або операція дескремблювання 928 може бути відведена. Різні вхідні сигнали після обробки можуть бути змішані за допомогою операції змішування 1089 і згодом розділені на нові пакети шляхом операції розділення 1106. Кожні індивідуальні вихідні пакети даних потім скремблюються операцією скремблювання 926, або, навпаки, операція скремблювання 926 відводиться, і потім зашифровується операцією шифрування 1026, або, навпаки, операція шифрування 926 може бути відведена.

Назва "медіа-вузол" відображає застосування програмного забезпечення зв'язку даного вузла зв'язку, або "програмного комутатора" відповідно до даного винаходу, зокрема, для

перенесення, маршрутизації та обробки контенту, що представляє в реальному часі голос, текст, музику, відео, файли, код і т.д., тобто медіа-контент. Медіа-вузол SDNP також представлений символічно для зручності як медіа-вузол SDNP $M_{a,j}$, розміщений на сервері 1215, як показано на рис. 83B. При використанні того самого коду можливі всі комбінації обробки сигналу з використанням показаного медіа-вузла SDNP, включаючи такі приклади:

- "Прохід одним маршрутом", де один вхід направляється на один вихід "як є" або, навпаки, шляхом вставки або видалення "сміттєвих" пакетів або розбирання вхідного пакета даних на трохи коротші пакети даних. Дана функція, показана на рис. 83C схематично та символічно як операція проходу одним маршрутом 1217A, придатна, коли медіа-вузол працює просто як сигнальний ретранслятор у мережі. "Сміттєва" та розбірна функції 1053 і 1052, як показано на рисунку, є невід'ємною властивістю операції змішування пакета 1061 і операції розділення пакета 1057, і включені тут тільки для зручності.

- "Надлишкова реплікація маршруту", де один вхід копіюється та відправляється "як є" на два або більше виходи або, навпаки, шляхом вставки або видалення "сміттєвих" пакетів або розбору вхідного пакета даних на коротші пакети даних перед пересиланням ідентичних копій та/або послідовностей даних на два або більше виходи. Дана функція, показана схематично та символічно на рис. 83D як операція надлишкової реплікації маршруту 1217B, придатна при реалізації "маршрутизації перегонів" для VIP клієнтів або термінового зв'язку, тобто відправлення двох копій різними шляхами та використання першої, яка надходить у пункт призначення. "Сміттєва" та розбірна функції 1053 і 1052 є невід'ємною властивістю операції змішування пакета 1061 і операції розділення пакета 1057, і включені тут тільки для зручності.

- "Одномаршрутне скремблювання", де один вхід скремблюється та направляється на один вихід незалежно від того, чи був попередньо зашифрований пакет. Як показано на рис. 83E, одномаршрутне скремблювання застосовується для зв'язку "першої милі" між клієнтом і хмарою або комюніке перед розділенням пакетів даних або змішуванням для багатомаршрутної або решітчастої передачі даних. Функція, представлена схематично та символічно як операція одномаршрутного скремблювання 1217C, включає операцію розділення одного вхідного пакета 1057; у цьому випадку використовується тільки для вставок і видалення "сміттєвих" даних та для розбору, при якій потрібна тільки операція скремблювання 1268B.

- "Одномаршрутне дескремблювання" – зворотна операція для одномаршрутного скремблювання, показана символічно як операція одномаршрутного дескремблювання 1217D на рис. 83F, використовується для повернення скрембльованого пакета у стан дескремблювання, незалежно від того, чи був пакет попередньо зашифрований до скремблювання. Функція включає серію комбінацій операції дескремблювання 1226A, за якою йде операція одномаршрутного змішування 1061, використовується для "сміттєвих" вставок та видалень, і для розбору пакета.

- Виконуючи послідовно дві попередні функції одномаршрутного дескремблювання та скремблювання, "одномаршрутне рескремблювання", показане схематично та символічно як операція одномаршрутного рескремблювання 1216C на рис. 83G, використовується для динамічного відновлення пакета скремблювання в єдиний маршрут.

- "Одномаршрутне шифрування", де один вхід зашифровується та направляється на один вихід, незалежно від того, чи був попередньо скрембльований пакет. Ця функція, представлена схематично та символічно як операція одномаршрутного шифрування 1217E на рис. 83H, використовується для зв'язку "першої милі" поза хмарою або для комюніке до розділення або змішування пакетів даних для багатомаршрутної або решітчастої передачі даних. Функція, відповідно до зображення, включає операцію розділення пакета з єдиним входом 1057; у цьому випадку використовується тільки для "сміттєвих" вставок або видалень і для розбору, за якими йде тільки операція шифрування 1226D.

- Зворотною для одномаршрутного шифрування операцією є операція "одномаршрутного дешифрування", яку показано символічно як одномаршрутну операцію дешифрування 1217F на рис. 83I, використовується для повернення зашифрованого пакета в його незашифрований стан, незалежно від того, чи був пакет попередньо скрембльований до шифрування. Функція включає серію комбінацій операції дешифрування 1226C, за якою йде операція одномаршрутного змішування 1061, використовується для "сміттєвих" вставок та видалень, і для розбору пакета.

- Виконуючи послідовно дві попередні функції одномаршрутного дешифрування та шифрування, "одномаршрутне повторне шифрування", показане схематично та символічно як операція одномаршрутного повторного шифрування 1216D на рис. 83J, використовується для динамічного відновлення пакета шифрування в єдиний маршрут.

- "Одномаршрутне скрембльоване шифрування", де один вхід скремблюється, зашифровується та направляється на один вихід. Функція, представлена схематично та символічно як операція одномаршрутного скрембльованого шифрування 1217G на рис. 83K, застосовується для зв'язку "першої милі" поза хмарою або для комюніке до розділення та змішування пакетів даних для багатомаршрутної або решітчастої передачі даних. Функція, як показано на рисунку, включає операцію розділення пакета з єдиним входом 1057. У цьому випадку використовується тільки для "сміттєвих" вставок і видалень та розбору, за якими йде операція скремблювання та шифрування 1226E.

- Зворотною для операції одномаршрутного скрембльованого шифрування є операція "Одномаршрутне дескрембльоване дешифрування", показана символічно як операція одномаршрутного дескрембльованого дешифрування 1217G на рис. 83L, що використовується для повернення скрембльованого зашифрованого пакета у вихідний дескрембльований та незашифрований стан. Функція включає низку комбінацій операції дешифрування дескремблювання 1226D, за якою йде операція одномаршрутного змішування 1061, використовувана для "сміттєвих" вставок та видалень, і розбору пакета.

- Виконуючи послідовно попередні функції одномаршрутного дешифрування, дескремблювання, скремблювання та шифрування, функція "Одномаршрутне репакетування", показана схематично та символічно як операція одномаршрутного репакетування 1216E на рис. 83M, застосовується для динамічного відновлення скремблювання та шифрування пакета в єдиний маршрут.

- "Решітчастий вхід шлюзу SDNP", іменований "шлюзом SDNP з єдиним входом і множинними виходами", схематично та символічно показано як операцію з єдиним входом і множинними виходами 1216F на рис. 83N. Тут єдине введення розділяється або направляється на кілька виведень для багатомаршрутної або решітчастої передачі даних, незалежно від того, чи був пакет попередньо скрембльований або зашифрований. Ця функція застосовується для ініціювання решітчастої маршрутизації дескремблювання дешифрування в шлюзі SDNP, включаючи "сміттєву" та розбірну функції 1053 і 1052 як невід'ємну властивість операції розділення пакета.

- Зворотною для попередньої функції решітчастого входу в шлюз є функція "Решітчастий вихід пакетного шлюзу", також відомий як "шлюз SDNP із множинними входами і єдиним виходом", яку показано схематично та символічно як операцію з множинними входами і єдиним виходом 1216G на рис. 83O. Тут єдине введення розділяється та направляється на кілька виведень для багатомаршрутної або решітчастої передачі даних, незалежно від того, чи був пакет скрембльований або зашифрований. Функція використовується для повторного складання компонентів пакетів повідомлення в шлюзі SDNP для зв'язку останньої милі або для міжхмарного транзитного переходу, тобто для завершення решітчастої маршрутизації SDNP, і вибірково містить у собі "сміттєві" та розбірні функції 1053 і 1052 як невід'ємні функції операції змішування пакета.

- "Скрембльований вхід у шлюз SDNP" показано символічно як операцію скремблювання з єдиним входом і множинними виходами 1217H на рис. 83P, де єдиний вхід розділяється, окремо скремблюється для кожного виходу, і направляється на множинні виходи для багатомаршрутної або решітчастої передачі даних, незалежно від того, чи був пакет попередньо зашифрований. Ця функція застосовується для ініціювання скрембльованої решітчастої маршрутизації в шлюзі SDNP, вибірково включаючи випадкову та розбірну функції (не показано) як вбудовані функції операції розділення.

- Зворотною для попередньої функції скремблювання входу в шлюз є функція "Дескрембльований вихід шлюзу SDNP", також відома як "дескремблювання із множинними входами, єдиним виходом із шлюзу SDNP", яку показано символічно як операцію дескремблювання з множинними входами і єдиним виходом 1217J на рис. 83P. Тут множинні решітчасті входи спочатку незалежно дескремблюються, а потім змішуються та направляються на єдиний вихід або користувачу, незалежно від того, чи був пакет зашифрований. Функція використовується для складання та дескремблювання компонентних пакетів повідомлення у шлюзі SDNP для зв'язку "останньої милі" або для міжхмарного транзитного переходу, тобто для завершення решітчастої маршрутизації SDNP, і вибірково містить у собі "сміттєві" та розбірні функції (не показано) як вбудовані функції операції розділення пакета.

- "Зашифрований вхід у шлюз SDNP" показано символічно як операцію шифрування з єдиним входом і множинними виходами 1217K на рис. 83Q. Тут єдиний вхід розділяється, зашифровується незалежно для кожного виходу; потім направляється на множинні виходи для багатомаршрутної або решітчастої передачі даних, незалежно від того, чи скремблювався пакет. Ця функція застосовується для ініціювання решітчастої маршрутизації шифрування в

шлюзі SDNP, вибірково включаючи "сміттєві" та розбірні функції (не показано) як вбудовані функції операції розділення.

- Зворотною для попередньої функції шифрування входу в шлюз є функція "Дешифрований вихід шлюзу SDNP", яку показано символічно як операцію дешифрування з множинними входами і єдиним виходом 1217L на рис. 83Q. Тут множинні решітчасті входи спочатку незалежно дешифруються для кожного входу, а потім змішуються та направляються на єдиний вихід або користувачу, незалежно від того, чи скремблювався пакет. Функція використовується для складання та дешифрування компонентних пакетів повідомлення в шлюзі SDNP для зв'язку "останньої милі" або для міжмарного транзитного переходу, тобто для завершення решітчастої маршрутизації SDNP, і вибірково містить у собі "сміттєві" та розбірні функції (не показано) як вбудовані функції операції змішування пакета.

- "Скремблований зашифрований вхід у шлюз SDNP" символічно показано як операцію скремблювання-шифрування з єдиним входом і множинними виходами 1217M на рис. 83R. Тут єдиний вхід розділяється, потім скремблюється та згодом зашифровується незалежно для кожного виходу, і, нарешті, направляється на множинні виходи для багатомаршрутної або решітчастої передачі даних. Ця функція застосовується для ініціювання зашифрованої решітчастої маршрутизації в шлюзі SDNP, і вибірково включає випадкову та розбірну функції (не показано) як вбудовані функції операції розділення.

- Зворотною для попередньої функції скремблювання-шифрування входу в шлюз є функція "Дескремблований дешифрований вихід шлюзу SDNP", яку символічно показано як операцію дескремблювання-дешифрування з множинними входами і єдиним виходом 1217N на рис. 83R. Тут множинні решітчасті входи спочатку дешифруються, потім дескремблюються незалежно для кожного входу, потім змішуються та направляються на єдиний вихід або користувачу. Ця функція застосовується для складання, дешифрування та дескремблювання компонентних пакетів повідомлення в шлюзі SDNP для зв'язку "останньої милі" або для міжмарного транзитного переходу, тобто для завершення решітчастої маршрутизації SDNP, і вибірково включає випадкову та розбірну функції (не показано) як вбудовані функції операції змішування пакета.

- "Решітчасте рескремблювання" символічно показано як операцію дескремблювання-скремблювання з множинними входами та множинними виходами 1216A на рис. 83S. Тут багатомаршрутні або решітчасті входи спочатку дескремблюються для кожного входу, незалежно від того, чи шифрується пакет, зливаються в довгий пакет даних або еквівалентний, якщо можливо, видаляючи "сміттєві" пакети. Довгий пакет даних потім розділяється на множинні нові пакети даних, по можливості вставляючи "сміттєві" дані. Кожен пакет даних потім незалежно скремблюється та, нарешті, направляється на множинні виходи для багатомаршрутної або решітчастої передачі даних. Функція використовується для оновлення скремблювання до нового стану або часових умов, тобто для спрощення "рескремблювання" пакета даних в момент, коли пакети даних перетинають хмару SDNP.

- "Решітчасте повторне шифрування" символічно показано як операцію дешифрування-шифрування з множинними входами та множинними виходами 1216B на рис. 83S. Тут багатомаршрутні або решітчасті входи спочатку дешифруються для кожного входу, незалежно від того, чи скремблювався пакет, зливаються в довгий пакет даних або еквівалентний, якщо можливо, видаляючи "сміттєві" пакети. Довгий пакет даних потім розділяється на множинні нові пакети даних, по можливості вставляючи "сміттєві" дані. Кожен пакет даних потім незалежно дешифрується та, нарешті, направляється на множинні виходи для багатомаршрутної або решітчастої передачі даних. Функція використовується для відновлення шифрування до нового стану або часових умов, тобто для спрощення "повторного шифрування" пакета даних в момент, коли пакети даних перетинають хмару SDNP.

- "Решітчасте репакетування" попередньо показано у схемі на рис. 83A і в символічній формі на рис. 83B. Тут багатомаршрутні або решітчасті входи спочатку дешифруються та згодом дескремблюються незалежно для кожного входу, а потім зливаються в довгий пакет даних або еквівалентний, якщо можливо, видаляючи "сміттєві" пакети. В одному прикладі довгий пакет даних повинен містити незашифрований звичайний текст або формат даних, відправлений користувачем. Згодом довгий пакет даних розділяється на множинні нові пакети даних, по можливості вставляючи "сміттєві" дані. Кожен пакет даних потім незалежно скремблюється, шифрується та, нарешті, направляється на множинні виходи для багатомаршрутної або решітчастої передачі даних. Функція використовується для відновлення як скремблювання, так і шифрування до нового стану або часових умов, тобто для спрощення "репакетування" пакета даних в момент, коли пакети даних перетинають хмару SDNP.

Перераховані вище переваги не припускають обмеження можливих перестановок або комбінацій, при яких використовується описаний медіа-вузол SDNP. Наприклад, кількість вхідних і вихідних каналів, тобто кількість медіа-вузлів SDNP, приєднаних до певного медіа-вузла SDNP, може варіюватися в межах від одного до дюжини з'єднань на один пристрій.

Чотири входи та виходи показані для зручності. На рис. 84А схематична діаграма, яка представляє потік сигналів, ілюструє зв'язок між будь-якими вузлами, такими як медіа-вузли $M_{a,b}$, $M_{a,j}$ and $M_{a,h}$, що включають комп'ютерні сервери 1220B, 1220J і 1220H відповідно, усі використовують програмне забезпечення зв'язку SDNP. Даний рисунок показує два з'єднання між двома будь-якими медіа-вузлами – одне, під'єднане від виходу з медіа-вузла, наприклад, $M_{a,b}$, до входу в інший медіа-вузол, наприклад, $M_{a,j}$ і друге з'єднання від виходу з останнього медіа-вузла $M_{a,j}$ до входу першого медіа-вузла $M_{a,b}$. Цей рисунок представляє рівень мережевого з'єднання 3, але не PHY або рівень передачі даних, що фактично може включати один кабель, лінію зв'язку по коаксіальному кабелю, кручену пару, Ethernet, або супутниковий канал між медіа-вузлами зв'язку. Через мережевий рівень надання немає ризику зворотного живлення, "станів гонитви", або нестійкості, що виникає через те, що вихід пристрою під'єднаний до входу іншого пристрою та що вихід пристрою під'єднаний до входу першого пристрою, тобто схематично мережа не описує електричного зворотного зв'язку.

Для того щоб представити мережу зв'язку або хмару SDNP 1114 відповідно до винаходу, як показано на рис. 84B, масив комп'ютерних серверів, що включає сервери 1220B, 1220D, 1220F, 1220H, 1220J, 1220S і 1220Q, кожен сервер, який використовує програмне забезпечення для введення в роботу медіа-вузла SDNP 1215, створює захищену мережу з відповідними медіа-вузлами $M_{a,b}$, $M_{a,d}$, $M_{a,f}$, $M_{a,h}$, $M_{a,j}$, $M_{a,s}$, та $M_{a,q}$, що представляють частину вузлів великої безпечної хмари.

Комп'ютерні сервери необов'язково повинні використовувати одну операційну систему (OS), за умови, що програмне забезпечення, використовуване на медіа-вузлі SDNP 1215, повинно включати виконувану програму, що відповідає програмному апарату OS. Виконуваною програмою є комп'ютерне програмне забезпечення, використовуване на зазначеній апаратній платформі, що виконує специфічні прикладні функції. Виконувана програма створюється за допомогою складання "вихідної програми". У міру того, як вихідна програма зорієнтована як логічно організована послідовність операцій, алгоритмів і команд, то вихідна програма перетворюється у виконувану програму, фактичну функціональність якої важко або неможливо визначити. Процес односторонній – вихідна програма може генерувати виконувану програму, але виконувана програма не може використовуватися для визначення виникнення вихідної програми. Це важливо для запобігання розкраданню операційної системи, тому хакери можуть тільки реверсувати фактичну програму.

Вихідна програма є не виконуваною, що використовує мову та синтаксис програмістів, але не є машинною програмою, яка повинна виконуватись у певній операційній системі. Під час компіляції, для генерованої виконуваної програми характерна одна операційна система iOS, Android, Windows 9, Windows 10, MacOS і т.д... Виконувана програма однієї операційної системи не зможе використовувати іншу. Вихідна програма, проте, може використовуватися для генерування виконуваної програми. Тому вихідна програма мережі SDNP доступна тільки розробникам вихідної програми, а не операторам мережі, що використовують виконувану програму SDNP.

Зв'язність вузлів у мережі, здебільшого, характерна для стандартних протоколів Ethernet, Wi-Fi, 4G і DOCSIS, описаних у розділі передмови даного застосування, передбачає загальну структуру для з'єднання пристроїв способом, що не залежить від виробників або OS. У робочому стані з'єднання мережі доставляє та передає пакети даних операційної системи сервера комп'ютера туди та назад, що направляє їх туди та назад від програмного забезпечення SDNP, яке накладається поверх комп'ютерної OS. Таким чином, заснована на медіа-вузлі SDNP функція зв'язку програмного комутатора може бути реалізована на будь-якому пристрої, незалежно від виробника, та може відповідати будь-якій великій підтримуваній операційній системі, включаючи UNIX, LINUX, MacOS 10, Windows 7, Windows 8 і т.д.

Іншим принципом є те, що хмара SDNP не має ні центрального керуючого пристрою, ні єдиного пристрою, який вирішує маршрутизацію пакетів, а також вузлової точки, яка має повне уявлення про пакети даних, що посилаються: які вони, куди направляються і як вони змішуються, розділяються, скремблюються та шифруються. Навіть оператор мережі не має повної картини трафіку даних у мережі. На рис. 84B представлено мережу комп'ютерів в одній хмарі. Значення знаходження в одній хмарі є суб'єктивним і довільним терміном і не повинне обмежувати універсальність описаного винаходу. Друга хмара, що включає медіа-вузли $M_{b,b}$, $M_{b,e}$, $M_{b,f}$, $M_{b,g}$, $M_{b,j}$, $M_{b,s}$, і $M_{b,t}$ (не показано), може включати різну географію або мати різних

сервісних провайдерів. Наприклад, Amazon може використовувати "Хмару А", тоді як Microsoft – "Хмару В", а приватна компанія або ISP може використовувати "Хмару С". У цілому, внутрішньовузлова можливість з'єднання краща і інтенсивніша всередині хмари, ніж для міжхмарових з'єднань, яких менше за кількістю і які потребують використання істинних Internet-сумісних IP адрес для зв'язку, замість застосування тимчасової пакетної маршрутизації, визначеної транслятором мережесовієднаних адрес (NAT).

Відносно представлення функцій, виконуваних будь-яким SDNP, один принцип включення або обходження функції віртуальними перемикачами – або виконуючи функцію, або залишаючи дані незмінними, можна рівною мірою застосовувати до дискусії вище чи на альтернативному пристрої, де функції скремблювання та шифрування заміщаються для того, щоб здійснювати дескремблювання до дешифрування та здійснювати шифрування до скремблювання. Для стислості, ці альтернативні потоки даних не ілюструються окремо від розуміння того, що послідовність може змінюватися доти, поки не буде здійснено зворотну функцію в протилежній операційній послідовності. Оскільки обробка пакета даних здійснюється в програмному забезпеченні, ця послідовність може бути змінена шляхом зміни послідовності алгоритму на основі даного випадку або періодично. Наприклад, щомісяця, щодня, щогодини або в порядку надходження запитів на з'єднання, часу або стану.

Як описано вище, будь-яка послідовність скремблювання, шифрування та змішування може застосовуватися настільки довго, наскільки вихідні дані відновлюються в такому самому зворотному порядку та такому самому масиві даних. Зміни вмісту між операціями без змін у дескремблюванні та дешифруванні можуть призвести до безповоротної втрати даних і незворотного ушкодження даних. З іншого боку, пакет може навіть скремблюватися більше, ніж один раз або шифруватися більше, ніж один раз у згрупованому порядку доти, поки дотримується правило зворотної послідовності для відновлення вихідних даних. Наприклад, додаток клієнта може зашифрувати повідомлення за допомогою власного методу для створення зашифрованого тексту при вході в шлюз SDNP. Шлюз медіа-вузла може зашифрувати пакет другий раз для передачі даних мережі. Даний метод працює доти, поки останній шлюз не дешифрує шифрування мережі на підставі "пакет-пакет" до того, як відбудеться дешифрування додатка клієнта.

Крім випадку шифрування на клієнті, з метою запобігання ризику розкрадання даних і втрати пакета, в одній реалізації відповідно до даного винаходу, існують такі положення для здійснення зв'язку на основі SDNP:

- SDNP скремблювання пакета повинне здійснюватися в додатку клієнта SDNP або при введенні хмари SDNP у шлюз медіа-вузла SDNP.

- В ідеалі, шифрування SDNP повинне відбуватися на кожному мережевому сегменті між двома медіа-вузлами SDNP, тобто пакет даних дешифрується до маршрутизації та дешифрується одразу при вході в наступний медіа-вузол SDNP.

- Принаймні рескремблювання повинне відбуватися щоразу, коли пакет даних входить або залишає хмару SDNP для зв'язку "останньої милі" або для міжхмарових транзитних переходів. Якщо пакет даних зашифрований в SDNP, він повинен бути дешифрований до того, як буде дескремблюваний, а потім скремблюється знову до того, як він буде зашифрований знову.

- Перед змішуванням бажано дешифрувати та дескремблювати вхідні пакети даних. Дешифрування та дескремблювання змішаних довгих пакетів може спричинити ушкодження даних. Також переважніше скремблювати і дешифрувати дані після розділення. Дешифрування та скремблювання змішаних довгих пакетів може спричинити ушкодження даних.

- "Сміттєві" пакети повинні бути вилучені з вхідних пакетів даних після дешифрування та дескремблювання, але перед змішуванням. "Сміттєві" видалення на змішаних довгих пакетах можуть спричинити ушкодження даних. Також переважніше вставляти "сміттєві" дані після розділення, але до скремблювання та шифрування. "Сміттєві" вставки на змішаних довгих пакетах можуть спричинити ушкодження даних.

- Окремо від шифрування положення користувача, рескремблювання (тобто дескремблювання, а потім скремблювання) як правило, не повинно проводитися на зашифрованих даних.

- Вставки "сміттєвих" даних повинні виконуватися узгоджено для зручності вставки та видалення.

- Вхідні пакети даних повинні бути дешифровані та дескремблювані відповідно до часу, стану і алгоритмів, при яких відбувається їхнє шифрування та скремблювання. Вихідні пакети даних повинні бути зашифровані та скремблювані відповідно до поточного часу, асоційованим станом і відповідним алгоритмом.

- Незашифровані пакети доцільно відновлювати тільки всередині медіа-вузлів. Усі пакети скремблюються, зашифровуються, змішуються, розділяються та/або містять "сміттєві" сегменти даних, поки вони перебувають на шляху між медіа-вузлами.

Хоча вищезгадані способи представляють можливі методи відповідно до винаходу, вони не 5 призначені для обмеження можливої комбінації або послідовності функцій SDNP. Наприклад, зашифровані пакети можуть потім скремблюватися, отже той самий пакет даних дескремблюється до дешифрування.

При впровадженні скремблювання здійснюється тільки всередині додатка клієнта SDNP, а не за допомогою медіа-вузлів у хмарі SDNP. У таких випадках безпечний внутрішньовузловий 10 зв'язок являє собою послідовність шифрувань і дешифрувань, як показано на рис. 84C. Тут функціональні компоненти SDNP у медіа-вузлі $M_{a,h}$, що включають операцію розділення 1106, операцію шифрування 1225A, операцію змішування 1089 і операцію дешифрування 1225B, показані експліцитно, тоді як медіа-вузли SDNP $M_{a,f}$ і $M_{a,j}$ зображені так, що виконують функцію ланкового повторного шифрування 1216B медіа-вузла SDNP тільки символічно.

У роботі дані, що надходять на медіа-вузла $M_{a,j}$ з іншого медіа-вузла (не показано) спочатку 15 направляються на операцію дешифрування 1225B на один із виходів з медіа-вузла $M_{a,h}$ і на операцію змішування 1089. Тут, якщо вони надходять в один і той самий час, пакети поєднуються з пакетами даних, що надходять з медіа-вузла $M_{a,f}$ незалежно, отже вони обробляються іншою операцією дешифрування 1225B. Після змішування пакети даних розділяються на нові та різні комбінації з різними адресами, заснованими на алгоритмі 20 розділення, що виконується операцією розділення 1106. Індивідуальні виходи потім незалежно зашифровуються окремими операціями шифрування 1225A, а потім направляються на медіа-вузли $M_{a,f}$ і $M_{a,j}$ та на інші медіа-вузли в мережі.

Упродовж цієї маршрутизації довгий пакет, який існує між операцією змішування 1089 і 25 операцією розділення 1106, у результаті може містити пакети даних з того самого сеансу зв'язку. Один пакет даних, що переміщається від медіа-вузла $M_{a,f}$ в медіа-вузол $M_{a,j}$ через медіа-вузол $M_{a,h}$. Інший пакет даних, що переміщається з медіа-вузла $M_{a,j}$ через медіа-вузол $M_{a,h}$ на медіа-вузол $M_{a,f}$ у той самий час, але в іншому напрямку. Через точне керування маршрутизацією в мережі SDNP відповідно до винаходу, описаного детальніше далі в цьому 30 додатку, довгий пакет даних може в будь-який час містити будь-яку комбінацію залежного та незалежного контенту; навіть дані або звукові фрагменти з того самого дуплексного сеансу зв'язку, що йде в протилежних напрямках. Якщо дані не надходять у той самий час, тоді пакети даних проходять послідовно через медіа-вузол у протилежних напрямках без використання того самого довгого пакета. У будь-якому разі, немає ніякої взаємодії або погіршення продуктивності 35 в медіа-вузлі SDNP, що несе множинні сеанси зв'язку в дуплексному режимі.

Незважаючи на те, що спочатку дана унікальна форма мережевого зв'язку може здатися 40 заплутаною, представляючи передачу даних способом, зображеним на рис. 84D, вона швидко показує простоту зв'язку даних у медіа-вузлі SDNP, навіть якщо медіа-вузол підтримує два напрямки дуплексного зв'язку одночасно. Наприклад, пакети даних, показані у вигляді заштрихованих ліній, що надходять на медіа-вузол $M_{a,j}$, спочатку проходять через операцію дешифрування 1032, потім операцію змішування 1089, операцію розділення 1106 і операцію шифрування 1026 і, нарешті, виходять з медіа-вузла $M_{a,j}$ і надходять на медіа-вузол $M_{a,h}$ у 45 заново зашифрованому стані; і потім, повторюють ту саму послідовність, але в новому часі та стані. Нарешті, пакети даних з медіа-вузла $M_{a,h}$ надходять у медіа-вузол $M_{a,f}$, де вони дешифруються, змішуються, розділяються та повторно зашифровуються, а потім відправляються на інший медіа-вузол у хмарі. Одночасно дані, що проходять в іншому напрямку, показані як незаштриховані лінії, надходять у медіа-вузол $M_{a,f}$, де вони дешифруються, змішуються, розділяються та повторно зашифровуються, потім відправляються на медіа-вузол $M_{a,h}$ і, нарешті, надходять через медіа-вузол $M_{a,j}$ на інші медіа-вузли в хмарі 50 SDNP.

Зв'язок "останньої милі". Канал зв'язку між клієнтом і хмарою SDNP описано тут як зв'язок "останньої милі". Термін "остання миля" містить у собі "першу милю", з'єднання між джерелом 55 виклику та хмарою, тому що увесь зв'язок є незмінно двостороннім і включає відправлене повідомлення та відповідь, або можливий дуплексний зв'язок. Власне, термін "остання миля" у значенні, як тут вживається, означає будь-яке з'єднання між клієнтом і хмарою SDNP незалежно від того, чи ініціював клієнт дзвінок або була викликана людина, тобто реципієнт. Приклад зв'язку "останньої милі" показано на рис. 85A. Тут хмара SDNP 1114 включає мережу комп'ютерних серверів 1118, які використовують програмне забезпечення для керування медіа-вузлами SDNP $M_{a,b}$, $M_{a,d}$, $M_{a,f}$, $M_{a,h}$, $M_{a,j}$, $M_{a,s}$, і $M_{a,q}$, що разом представляють, щонайменше, 60 частину вузлів захищеної хмари. Зокрема, у показаному прикладі комп'ютерний сервер 1220H,

що є медіа-вузлом SDNP $M_{a, h}$, працює як медіа-вузол шлюзу SDNP, з'єднаний прямо або безпосередньо з базовою станцією LTE 17 і з'єднаний через вежу стільникового зв'язку 18 і радіолінію 13 до стільникового телефону 32 як клієнт. Як використаний тут, термін "вузол шлюзу" або "медіа-вузол шлюзу" стосується медіа-вузла, що з'єднується з вузлом поза мережею SDNP, зазвичай це пристрій клієнта, такий як стільниковий телефон або комп'ютер, і в цьому випадку з'єднання між вузлом шлюзу та пристроєм клієнта є з'єднанням "останньої милі".

Приклад, де безпечний вузол шлюзу SDNP з'єднується з небезпечною "останньою милею", показано на рис. 85B, наприклад, вузол шлюзу SDNP з'єднується з телефоном, що не має встановленого на ньому додатку SDNP. Відповідно до зображення, стільниковий телефон 32 під'єднаний шляхом радіозв'язку 28 до решітчастої вежі 18, що відправляє та одержує пакети даних від стільникового телефону 32 і перетворює їх у кабельний зв'язок, такий як Ethernet, волоконно-оптичний кабель, коаксіальний кабель, мідний кабель і т.д., що використовує базову станцію LTE 17. Незважаючи на те, що передача пакетів даних здійснюється двоспрямовано на єдиному мережевому PHY рівні 1 зв'язку, провідної, кабельної, радіо або супутникової лінії зв'язку, потік даних представлений окремо для пакетів, відправлених зі стільникового телефону 32 на медіа-вузол SDNP $M_{a, h}$ і навпаки. Відповідно до зображення, "остання миля" небезпечна доти, поки використовуваний додаток у стільниковому телефоні має вбудоване шифрування та викликувана особа використовує той самий додаток з тим самим шифруванням.

У процесі роботи пакети даних посилають зі стільникового телефону на медіа-вузол шлюзу SDNP $M_{a, h}$, не дешифруються та не скремблюються, тому що ці функції відключені, тобто замкнуті та, як такі, не показані. Замість цього вхідні пакети даних надходять прямо на операцію змішування 1089, змішуючи їх з іншими пакетами, а потім розділяючи їх на множинні виходи для решітчастої передачі даних за допомогою операції розділення 1106. Потім кожен із цих виходів захищається за допомогою операції скремблювання 926 і операції шифрування 1026 до передачі даних. Один вихід, показаний як приклад, направляється на медіа-вузол $M_{a, f}$, у сервері 1220F. Так само повідомлення може бути оброблено медіа-вузлом $M_{a, f}$ для міжхмарового зв'язку, як описано вище, і направляється далі на інший медіа-вузол, наприклад, медіа-вузол $M_{a, j}$ комп'ютерного сервера 1220J.

Потоки даних з хмари на стільниковий телефон з медіа-вузла $M_{a, f}$, у сервері 1220F та з інших медіа-вузлів обробляються у зворотній послідовності, починаючи з операцій дешифрування 1032 і дескремблювання за допомогою операції дескремблювання 928, а потім – змішуючись з іншими вхідними пакетами в тимчасовий довгий пакет шляхом операції змішування 1089. Потім довгий пакет розділяється на частини шляхом операції розділення 1106, направляючи деякі пакети в мережі та розділяючи пакети, які повинні бути відправлені на стільниковий телефон 32. Ці пакети можна відправляти разом або розбирати та відправляти послідовно в окремі пакети назад на базову станцію LTE 17 і далі на стільниковий телефон 32.

Пакети даних, що проходять по мережі, можуть бути повторно зашифровані та рескрембльовані, як описано вище. В іншому випадку, в одному варіанті пакети даних залишаються скрембльованими без рескремблювання по всій хмарі, але їх можна повторно зашифрувати на кожному медіа-вузлі. При такій системі одноразового скремблювання та одноразового дескремблювання скремблювання відбувається у вузлі шлюзу, де пакети входять у хмару; і дескремблювання відбувається у вузлі шлюзу, де пакети залишають хмару, тобто в медіа-вузлі шлюзу "першої" та "останньої милі". Хоча, як зазначалося вище, медіа-вузол, з'єднаний з "першою" або "останньою милею", можна назвати вузлом шлюзу. Насправді він містить те саме програмне забезпечення медіа-вузла SDNP і функціональність, як і будь-який інший медіа-вузол у хмарі, але функціонує по-різному при контакті з клієнтом.

Іншою опцією для здійснення одноразового скремблювання і одноразового дескремблювання зв'язку SDNP є здійснення скремблювання у пристрої клієнта за допомогою програмного забезпечення. Як показано на рис. 85C, у з'єднанні між стільниковим телефоном 32 і медіа-вузлом SDNP $M_{a, f}$ на комп'ютерному сервері 1220F, медіа-вузол SDNP $M_{a, h}$ виступає як медіа-вузол шлюзу між клієнтом і хмарию SDNP, де медіа-вузол шлюзу SDNP $M_{a, h}$ включає операцію змішування 1089, операцію розділення 1106, операцію шифрування 1225A, операцію скремблювання 1226B, операцію дешифрування 1225B і операцію дескремблювання 1226A. Як було зазначено вище, будь-який медіа-вузол, вузол зв'язку, позначений як вузол M, здатний виконувати будь-яку комбінацію всіх операцій безпеки, тобто змішування та розділення, шифрування та дешифрування, скремблювання та дескремблювання і т.д. У процесі роботи пакети даних скремблюються в стільниковому телефоні 32 за допомогою програмного забезпечення SDNP, переміщуються по радіозв'язку 28 у вежу LTE 18, де базова станція LTE 17 перетворює сигнали в Ethernet, оптичний кабель або в іншу провідну лінію зв'язку на вузол шлюзу SDNP. Залежно від локального носія частини цього зв'язку можуть містити трафік через

приватний NAT або включати дані, що переміщуються через Internet. Потім пакети даних направляються з базової станції LTE 17 на медіа-вузол SDNP $M_{a,h}$, що виступає як вузол шлюзу SDNP.

Потім вхідний пакет даних направляється для проходження операції 1216H, і згодом змішується з іншими вхідними пакетами даних за допомогою операції змішування 1089, а потім розділяється операцією розділення 1106 з пакетами даних зі стільникового телефону 32, спрямованими на медіа-вузол $M_{a,f}$ через операцію шифрування 1225A. Таким чином, дані, що проходять через хмару, зашифровуються за допомогою шлюзу, але скремблюються за допомогою додатка клієнта SDNP. І, навпаки, зашифрований і скремблований трафік даних із хмари SDNP направляється через медіа-вузол $M_{a,f}$, проходить через операцію дешифрування 1225B, змішується операцією змішування 1089 і розділяється на нові пакети операцією розділення 1106, витягаючи пакети даних зі стільникового телефону 32, як їхнього пункту призначення, і направляючи пакети даних на стільниковий телефон 32, немодифікований перехідною операцією 1216H. Таким чином, цілий зв'язок скремблюється від початку до кінця, але тільки зашифровується всередині хмари SDNP.

Модифікація вищезгаданого методу усе ще забезпечує скремблювання як в "останній милі", так і в хмарі, але скремблювання "останньої милі" відрізняється від скремблювання в хмарі. Як показано на рис. 85D, у з'єднанні між стільниковим телефоном 32 і медіа-вузлом SDNP $M_{a,f}$ на комп'ютерному сервері 1220F, медіа-вузол SDNP $M_{a,h}$ виступає як вузол шлюзу між клієнтом і хмарою SDNP, де медіа-вузол SDNP $M_{a,h}$ включає операцію змішування 1089, операцію розділення 1106, операцію скремблювання та шифрування 1226C, операцію дешифрування та дескремблювання 1226D, операцію скремблювання 1226B і операцію дескремблювання 1226A. Під час роботи пакети даних скремблюються в стільниковому телефоні 32 за допомогою програмного забезпечення SDNP, переміщуються по радіозв'язку 28 у вежу LTE 18, і базова станція LTE 17 перетворює сигнали в Ethernet, оптичний кабель або в іншу провідну лінію зв'язку на вузол шлюзу SDNP. Залежно від локального носія частини цього зв'язку зі стільникового телефону на базову станцію 7 можуть містити трафік через приватний NAT або включати дані, що переміщуються через Internet. Потім пакети даних направляються з базової станції LTE 17 на медіа-вузол SDNP $M_{a,h}$, що виступає як вузол шлюзу SDNP.

Потім вхідний пакет даних направляється на операцію дескремблювання 1226A і згодом змішується з іншими вхідними пакетами даних за допомогою операції змішування 1089, а потім розділяється операцією розділення 1106 з пакетами даних зі стільникового телефону 32, спрямованих на медіа-вузол $M_{a,f}$ через операцію скремблювання та шифрування 1226C. Таким чином, дані, що проходять через хмару, зашифровуються та скремблюються за допомогою вузла шлюзу, але іншим способом, ніж скремблювання, використовуваним додатком клієнта SDNP для безпеки "останньої милі". І, навпаки, зашифрований і скремблований трафік даних із хмари SDNP направляється через медіа-вузол $M_{a,f}$ через операцію дешифрування та дескремблювання 1226D. Потім змішується операцією змішування 1089 і розділяється на нові пакети операцією розділення 1106, витягаючи пакети даних зі стільникового телефону 32, як їхнього пункту призначення, і направляючи пакети даних на стільниковий телефон 32 за допомогою операції скремблювання 1226B. Пакети даних, що входять у стільниковий телефон 32, дескремблюються додатком з підтримкою SDNP. Таким чином, зв'язок у хмарі як зашифровується, так і скремблюється в медіа-вузлах, тоді як "остання миля" скремблюється у вузлі шлюзу та у додатку телефону на відміну від скремблювання хмари. Одним важливим аспектом скремблювання та дескремблювання пакетів даних у телефоні є метод, використовуваний для передачі інформації про стан, числові ключі або розділені секрети між хмарою та клієнтом. Це питання розглядається далі в цьому документі.

Фрагментована передача даних. Відповідно до даного винаходу, мережа комп'ютерних серверів, що використовують програмне забезпечення для виконання функцій медіа-вузла SDNP, полегшує глобальний безпечний зв'язок із широким спектром пристроїв на основі фрагментації даних у зв'язку з комутацією пакетів. Як показано на рис. 86, хмара SDNP 1114, що включає мережу комп'ютерних серверів, які використовують програмне забезпечення для оперування медіа-вузлами SDNP $M_{a,b}$, $M_{a,d}$, $M_{a,f}$, $M_{a,h}$, $M_{a,j}$, $M_{a,s}$, and $M_{a,q}$ та іншими, не показаними, може бути приєднана до великої кількості пристроїв і клієнтів, включаючи: (а) базову станцію LTE 17 з радіовузлами 28 до стільниковому телефону 32 і планшету 33. Базова станція 17 також може бути з'єднана за допомогою радіозв'язку з будь-яким пристроєм, що підтримує стандарт LTE; (б) публічним Wi-Fi 100 з Wi-Fi-антенною 26, що забезпечує радіозв'язок Wi-Fi 29 з ноутбуком 35 або планшетами, стільниковими телефонами, електронними книгами та іншими пристроями, підключеними до Wi-Fi, включаючи інтернет-пристрої; (в) кабель CMTS 101, під'єднаний оптичним кабелем або коаксіальним кабелем до кабельного модему 103 і потім до

персонального комп'ютера 36 або домашньої станції Wi-Fi, мережевих пристроїв і т.д.; (г) кабель CMTS 101, під'єднаний оптичним кабелем або коаксіальним кабелем до телевізійної приставки TV STB 102 і потім до HDTV 39; (д) провідне з'єднання з інтернет-роутерами 66A, 66B, 66C; (е) професійні радіомережі 14, такі як TETRA і EDAC, приєднані через радіовежі 15 до портативної радіостанції дуплексного зв'язку 16B, базових станцій 16A і професійних транспортних засобів 40; (ж) відомчою телефонною станцією PBX 8 і стаціонарними телефонами 9; і (з) за допомогою PSTN шлюзу 3 до традиційних телефонних мереж з комутацією каналів POTS. Відповідно до зображення, будь-який медіа-вузол SDNP може функціонувати як вузол шлюзу.

Проста ілюстрація передачі пакетів даних показана на рис. 87, на якому зображені приклади зв'язку, заснованого на хмарі SDNP між планшетом 33 і автомобілем 1255, що включає пакети даних 1056, послідовно 2A, 2B, 2C, 2D, 2E і 2F, і між ноутбуком 35 і стільниковим телефоном 32, що включає пакети даних 1055, послідовно 1A, 1B, 1C, 1D, 1E, і 1F. Інший пакет даних 1250, послідовно як 3A, 3B, 3C, 3D, 3E і 3F; пакет даних 1252, послідовно як 4A, 4B, 4C, 4D, 4E і 4F; і пакет даних 1251, послідовно як 5A, 5B, 5C, 5D, 5E і 5F, також передаються через мережу одночасно з пакетами даних 1255 і 1256. Короткі пакети представляють складові частини в різний час упродовж передачі для ілюстрації динамічного характеру мережевої передачі даних.

У показаному прикладі дані кожного пакета скремблюються таким чином, що послідовність сегментів даних може бути у випадковому порядку або випадково виявитися в порядку зростання. Сегменти даних одного комюніке або сеансу зв'язку можуть також чергуватися з незв'язаними сегментами даних. У результаті дуже малоймовірно, щоб пакет даних, який одноразово входить у хмару SDNP, не змішувався з іншими незв'язаними сегментами даних. Фактично в будь-якому конкретному пакеті даних, що проходять через два медіа-вузли SDNP, змішування незв'язаних сегментів даних і скремблювання порядку цих пакетів є нормальною умовою. Завдяки великій кількості або сеансу зв'язку та пакетам даних, що проходять через хмару одночасно, існує ймовірність того, що всі дані, що залишилися в одному пакеті даних, будуть статистично вилучені. За відсутності достатніх даних операція змішування в медіа-вузлах надає "сміттєві" дані. Включення різних сегментів даних незв'язаних між собою даних, відповідно до зображення, ілюструє принцип змішування комюніке та сеансів зв'язку в пакетах даних упродовж передачі даних SDNP, але не точно відображає реальну кількість і частоту незв'язаних даних або "сміттєвих" сегментів даних і заповнювача, присутнього у пакетах даних.

Рис. 88A демонструє початок сеансу зв'язку в часі t_0 і відповідному стані 990 з ноутбука 35 на стільниковий телефон 32, починаючи з пакета даних 1055 і незв'язаних пакетів даних 1056 і 1250 через 1252, що входить у мережу через різні вузли шлюзів, включаючи $M_{a,q}$, $M_{a,h}$, $M_{a,b}$, і $M_{a,s}$. Як показано на рис. 88B, у часі t_1 і відповідному стані 991 пакет даних 1055 розділяється на кілька частин пакетів даних. Один такий пакет даних 1261A, що включає сегменти даних 1A і 1B у порядку зростання, але змішується з незв'язаними сегментами даних, направляється на медіа-вузол $M_{a,b}$. Пакет даних 1261B, що включає сегменти даних 1D, 1C і 1F у скремблованому порядку, а також змішуваний з незв'язаними сегментами даних, направляється на медіа-вузол $M_{a,j}$, і пакет 1261C, що включає сегмент даних 1E, направляється на медіа-вузол $M_{a,h}$.

Як показано на рис. 88C, у часі t_2 і відповідному стані 992 дані розділяються на кілька комбінацій частин пакетів даних. Зокрема, пакет даних 1261A розділяється на нові пакети даних 1262A і 1262B, де пакет даних 1262A, що включає сегмент даних 1A та інші сегменти даних, направляється на медіа-вузол $M_{a,s}$, тоді як пакет даних 1262B, що включає сегмент даних 1B, направляється на медіа-вузол $M_{a,d}$. Пакет даних 1261B також розділяється на компоненти пакетів даних 1262C і 1262D, де пакет даних 1262C, що включає сегменти даних 1C і 1F у порядку зростання, але, змішуваний з незв'язаними сегментами даних, направляється на медіа-вузол $M_{a,d}$, тоді як компонент пакетів даних 1262D, що включає сегмент даних 1D, направляється на медіа-вузол $M_{a,f}$. При цьому пакет даних 1262E, що включає сегмент даних 1E, продовжує перехід один або змішується з незв'язаними пакетами даних (не показано) на медіа-вузол $M_{a,f}$.

Як показано на рис. 88D, у часі t_3 і відповідному стані 993 пакет даних 1263A, що включає сегмент даних 1A, і пакет даних 1263C, що включає сегменти даних 1D і 1E, доставляються на медіа-вузол $M_{a,d}$, тоді як пакет даних 1263B, що включає сегменти даних 1B, 1C і 1F, очікує їхнього прибуття на той самий медіа-вузол $M_{a,d}$. Як показано на рис. 88E, у часі t_4 і відповідному стані 994 медіа-вузол $M_{a,d}$ змішує пакети даних 1263A, 1263B і 1263C, відновлюючи вихідний пакет даних 1055, і направляє пакет даних 1055 на стільниковий телефон 32 разом або вроздріб. Короткий опис передачі пакетів даних між ноутбуком 35 і стільниковим телефоном 32 показано на рис. 88F.

Як показано на рис. 89A, незалежно від і одночасно зі зв'язком між ноутбуком 35 і стільниковим телефоном 32, планшет 33 зв'язується з автомобілем 1255, починаючи з часу t_0 і відповідного стану 990, коли пакет даних 1056 входить у безпечну хмару 1114. Як показано на рис. 89B, під час t_1 і у відповідному стані 991 вхідний пакет даних 1056 розділяється на

компоненти пакетів даних 1261D і 1261E, де пакет 1261D, що включає сегменти даних 2B і 2C у скрембльованому, але у випадковому порядку зростання, направляється на медіа-вузол $M_{a,q}$, і пакет 1261E, що включає сегменти даних 2E, 2F, 2A і 2D у скрембльованому порядку, направляється на медіа-вузол $M_{a,j}$.

Як показано на рис. 89C, у часі t_2 і у відповідному стані 992 пакет даних 1261D модифікується, скремблюючи порядок даних і вставляючи сегменти даних з інших джерел для створення пакета даних 1262F. Аналогічно, пакет даних 1261E розділяється медіа-вузлом $M_{i,j}$ на кілька пакетів даних 1262G, 1262H і 1262J. Пакет даних 1262J, що включає сегмент даних 2A, направляється на медіа-вузол $M_{a,f}$. Скрембльований пакет даних 1262H, що включає сегменти даних 2D і 2E, що змішується з числом незв'язаних сегментів даних, направляється на медіа-вузол $M_{a,d}$. Також, у часі t_2 пакет даних 1262G, що включає сегмент даних 2F, направляється на медіа-вузол $M_{a,s}$.

Як показано на рис. 89D, у часі t_3 і у відповідному стані 993 пакет даних 1263D, що включає сегменти даних 2B і 2C у порядку зростання, направляється на вузол $M_{a,s}$, де пакет даних 1263E, що включає сегмент даних 2F, очікує прибуття інших пакетів. Одночасно пакет даних 1263G направляється на медіа-вузол $M_{a,d}$, де пакет даних 1263F, що включає сегменти даних 2D і 2E у порядку убуття, очікує. Це показує, що в мережі SDNP пакети даних можуть бути відправлені негайно або за бажанням можуть тимчасово затриматися. Як показано на рис. 89E, у часі t_4 і у відповідному стані 994 пакет даних 1264B, що включає сегменти даних 2D, 2A і 2E у порядку скремблювання, направляється на медіа-вузол $M_{a,s}$, де пакет даних 1264A, що включає сегменти даних 2B, 2C і 2F, очікує. Як показано на рис. 89F, під час t_4 останній пакет даних 1056 збирається та направляється на автомобіль 1255, або, навпаки, усі компоненти сегментів даних останнього пакета даних 1056 направляються в незмішаній формі на автомобіль 1255 і знову збираються там. Стисле викладення маршрутизації пакетів даних 1056 із планшета 33 на автомобіль 1255 показано на рис. 89G.

Відповідно до зображення, пакети даних, що проходять через хмару SDNP, переносять одночасно кілька сеансів зв'язку на різні відстані, динамічно змінюючи контент з одного медіа-вузла SDNP на наступний. Змішування або розділення незв'язаних між собою сегментів даних не несе несприятливого впливу, не призводить до втрати даних та не створює жодних перешкод під час сеансу зв'язку. Наприклад, як показано на рис. 87, пакет даних 1257 містить сегменти даних 1C і 1F, спрямовані на стільниковий телефон 32, сегменти даних 2D і 2E, спрямовані на автомобіль 1255, та інші незв'язані сегменти даних і "сміттєві" дані, усі з яких доставляються в різні місця призначення, не піддаються тимчасовому обміну пакетами даних з іншими незв'язаними сегментами даних.

До того ж, оскільки жоден пакет даних не містить повного слова, звуку або сеансу зв'язку, фрагментація даних і решітчаста маршрутизація, використовувані медіа-вузлами SDNP відповідно до даного винаходу, роблять вміст пакета даних незрозумілим і невразливим для атаки "людина посередині". Як показано на рис. 90, під час t_1 злоумисник-порушник 630, що аналізує трафік пакетів даних при переході в і з медіа-вузла $M_{a,j}$, бачить тільки зашифровані пакети 1270A, 1271A, 1272A і 1273A. Малоімовірно, коли зашифровані файли порушуються, базовий вміст відкритих текстів з пакетів 1270B, 1271B, 1272B і 1273B включає скрембльоване неповне поєднання сегментів даних. Ця умова даних зберігається тільки на долю секунди, перш ніж нові пакети даних пройдуть через той самий медіа-вузол. Навіть без скремблювання та змішування обмежений час, доступний для дешифрування пакета даних перед його повторним шифруванням, рескремблюванням, повторним розділенням або репакетуванням, робить неможливими навіть ті атаки, при яких використовують суперкомп'ютери.

Рисунок 91A ілюструє динамічний стан медіа-передачі даних SDNP, використовуючи час як основу передачі даних. Дані, показані тут, такі самі, як накладення даних, показане на діаграмі мережі на рис. 87. У зображенні, заснованому на часі, пакет даних 1056 із планшета 33 розділяється на пакети даних 1261A, 1261B і 1261C. Під час t_2 пакет 1261A розділяється на нові пакети даних 1262A та 1262B і пакет даних 1261B розділяється на нові пакети даних 1262C та 1262D; і пакет даних 1261C оновлюється до пакета даних 1262E без зміни контенту. Під час t_3 пакет даних 1262A оновлюється до пакета даних 1263A без зміни його контенту; і пакети даних 1262B та 1262C змішуються в пакет даних 1263B, тоді як пакети даних 1262D і 1262E змішуються в пакет даних 1263. Під час t_4 пакети даних 1263A, 1263B і 1263C змішуються для відновлення пакета даних 1055.

Передача даних SDNP може бути також представлена у формі таблиці. Наприклад, таблиця 1279, показана на рис. 91B, показує обробку пакетів даних у часі t_3 , показуючи вихідні медіа-вузли, вхідні пакети, час, коли вхідні пакети були зашифровані, час, коли вхідні пакети були скрембльовані, останній час, коли пакети даних змішалися та розділилися, тобто злилися, і отримані вихідні пакети. Медіа-вузол використовує дану інформацію, щоб знати, що робити із вхідними пакетами даних, як репакетувати дані і як повторно зашифрувати або рескремблювати дані, якщо це необхідно.

Як показано на рис. 91C, ще одним аспектом динамічного характеру медіа-передачі даних SDNP є здатність тимчасово утримувати пакети в медіа-вузлі, очікуючи прибуття інших пакетів. Використовуючи ті самі дані, як показано вище на рис. 87, даний механізм ілюструє пакет 1056 на основі часу. У часі t_1 вхідний пакет даних 1056 скремблюється, а потім розділяється на пакет даних 1261D, що включає сегменти даних 2B і 2C, і пакет даних 1261E, що включає пакети 2A, 2D, 2E і 2F. Під час t_2 комюніке розпадається на чотири частини, пакети даних 1262F, 1262G, 1262H і 1262J, останні три є результатом поділу пакета даних 1261E на пакет даних 1262G, що включає сегменти даних 2F; пакет даних 1262H, що включає сегменти даних 2D і 2E; і пакет даних 1262J, що включає сегмент даних 2A. Пакет даних 1261D, що включає сегменти даних 2B і 2C, переміщається по мережі з незмінним вмістом, тобто як пакет даних 1262F у часі t_2 і як пакет даних 1263D у часі t_3 . Аналогічно, у момент часу t_3 пакет даних 1262J, що включає сегмент даних 2A, залишається з незмінним вмістом як пакет даних 1263G.

Щоб представити пакет даних, який тимчасово зберігається в медіа-вузлі, рис. 91C ілюструє пакет даних, що переміщається з заданого медіа-вузла на той самий медіа-вузол з відповідними коефіцієнтами наростання часу. Наприклад, між часом t_3 і часом t_4 пакет даних 1263E, що включає сегмент даних 2F, так само як і його попередній пакет даних 1262G, переміщається з медіа-вузла $M_{a,s}$ на той самий медіа-вузол $M_{a,s}$, тобто пакет не переміщається. Незважаючи на стаціонарний стан пакета даних, шифрування та скремблювання може змінитися, щоб відбити оновлений час, схематичне зображення вмісту пакета даних 1263E, що переміщається з вихідного медіа-вузла $M_{a,s}$ на медіа-вузол з ідентичним місцем призначення $M_{a,s}$ у момент часу t_4 означає, що він зберігається в пам'яті медіа-вузла $M_{a,s}$.

Аналогічно, між часом t_3 і часом t_4 пакет даних 1263F, що включає сегменти даних 2D і 2E, так само як і його попередній пакет даних 1262H, переміщається з медіа-вузла $M_{a,d}$ на медіа-вузол $M_{a,d}$. Це знову означає, що пакет не переміщається та зберігається тимчасово в пам'яті. У момент часу t_4 вхідний пакет даних 1263D змішується в медіа-вузлі $M_{a,s}$ з пакетом даних 1263E, що зберігався в пам'яті з моменту часу t_3 , що призводить до нового об'єднаного пакета 1264A, який включає з'єднані дані сегментів 2B, 2C і 2F. Цей новий пакет даних 1264A залишається на зберігання в медіа-вузлі $M_{a,s}$, очікуючи більше вхідних даних. Тим часом, у момент часу t_4 у медіа-вузлі $M_{a,d}$ пакети даних 1263F і 1263G змішуються та направляються на медіа-вузол $M_{a,s}$, як пакет даних 1264B, що включає сегменти даних 2A, 2D і 2E. У момент часу t_4 пакет вхідних даних 1264B змішується зі стаціонарним пакетом даних 1264A, що очікує в медіа-вузлі $M_{a,s}$ з моменту часу t_4 , створюючи вихідний пакет даних 1056, що відправляється на автомобіль 1255.

Як описано, у способах, показаних відповідно до винаходу, дані можуть проходити через хмару SDNP або залишатися нерухомими в певному медіа-вузлі, очікуючи надходження вхідних даних перед обробкою.

Транспортна команда та керування. Для того щоб медіа-вузлу дізнатися, як обробляти вхідні пакети даних, він повинен якимось чином отримати інформацію щодо алгоритмів, початкових чисел і ключів, які будуть використовуватися в скремблюванні, дескремблюванні, шифруванні, дешифруванні, змішуванні, розділенні, вставці та видаленні "сміттєвих" і розбірних пакетів даних. Ця важлива інформація може передаватися різними способами або їхньою комбінацією:

- Направляючи розділені секрети на медіа-вузол як частини встановлення програмного забезпечення SDNP або нових версій.

- Направляючи контрольні дані через медіа-вузли перед відправленням контенту.

- Направляючи контрольні дані через медіа-вузли як частину пакета даних.

- Направляючи контрольні дані через канал даних, окремий від медіа-вузлів, які обмінюються інформацією. Наприклад, через мережу "сигнального сервера", що працює паралельно до медіа-вузлів.

- Зберігаючи інформацію про ідентичність пристроїв, приєднаних до мережі SDNP, і їх відповідні IP або SDNP адреси на серверах SDNP, відмінні від сигнальних серверів або серверів, що працюють як медіа-вузли, які несуть контент.

Наприклад, як показано на рис. 92A, у момент часу t_3 , що відповідає стану 993, пакет даних 1262B, що включає сегмент даних 1B, пакет даних 1262C, що включає сегменти даних 1C і 1F, і

пакет даних 1262Н, що включає незв'язані сегменти даних, входять у медіа-вузол $M_{a,d}$. Після входження в медіа-вузол вхідні пакети даних 1262В, 1262С і 1262Н, які для ясності показані в дешифрованій формі, спочатку обробляються операціями дешифрування та дескремблювання. Потім пакети даних 1262В, 1262С і 1262Н змішуються, включаючи збір частин, видаляючи "сміттєві" біти, щоб створити вихідний пакет даних 1263В, що включає сегменти даних 1В, 1С і 1F. Щоб виконати дане завдання, комп'ютерний сервер 1220D, що є хостом для медіа-вузла $M_{a,d}$, повинен спочатку отримати певну інформацію щодо часу та відповідних станів, використовуваних для створення вхідних пакетів даних. Ця інформація може міститися в пакеті даних як заголовок, або може бути відправлена заздалегідь на медіа-вузол із сигнального вузла або іншого медіа-вузла. Як описано в таблиці на рис. 91В, ці вхідні пакети даних були востаннє зашифровані в момент часу t_2 . Пакети були скрембльовані востаннє в момент часу t_1 , що відповідає стану 1301А, або, можливо, у момент часу t_2 , що відповідає стану 1301В. Дана інформація повинна бути надана на вузол $M_{a,d}$ для правильної обробки вхідних даних відповідно до умов, використовуваних для створення пакетів даних. Інформація про стан у моменти часу t_1 і t_2 використовується для створення відповідних ключів D 1306А і 1306, необхідних для пакетного дешифрування вхідних пакетів за допомогою генератора ключів D₁ 1305А і генератора ключів D₂ 1305В. Генератори ключів дешифрування виконуються за допомогою програмного забезпечення, розташованого на сервері DMZ, приєднаному до вузла зв'язку $M_{a,d}$. Загальні вказівки з експлуатації та генерування шифрування та дешифрування ключів були описані в передмові до цього документа. На противагу статичному шифруванню, шифрування в мережі SDNP є динамічним. Це означає, що єдиний шлях для створення правильного ключа дешифрування – це інформація про файл, який було зашифровано. Ця інформація передається як час і стан, доставлені разом із вхідним пакетом даних, або, навпаки, до надходження пакета, і використовується для вибору потрібного алгоритму шифрування для генерування асоційованого генератора ключів. Алгоритми шифрування і їхні асоційовані генератори ключів дешифрування зберігаються як розділені секрети на безпечному сервері DMZ, приєднаному до вузла зв'язку $M_{a,d}$.

Незважаючи на те, що пакети даних можуть бути зашифровані, для ілюстрації вони показані в їхньому незашифрованому вигляді. Така сама інформація про стан також застосовується до множинних початкових станів генератора 1303 для створення відповідних множинних початкових станів 1304А і 1304В для визначення алгоритмів, використовуваних у моменти часу t_1 і t_2 для створення пакетів даних. Початкові стани можуть генеруватися двома способами. У першому випадку початкові стани генеруються, використовуючи програмне забезпечення, розташоване на сервері DMZ, приєднаному до медіа-вузла, де відбувається скремблювання, змішування та шифрування переданих пакетів даних. У таких випадках початкові стани повинні бути доставлені на вузол зв'язку $M_{a,d}$ до надходження пакета даних.

У другому випадку час створення вхідного пакета, що надходить на вузол зв'язку $M_{a,d}$ як частина заголовка вхідного пакета даних або в окремому пакеті, доставлених заздалегідь даних. Потім час подається в числовий генератор 1303, розташований на сервері DMZ, приєднаному до вузла зв'язку $M_{a,d}$. Незалежно від того, де вони генеруються локально або на джерелі, а потім подаються, генеровані початкові стани подаються в селектор 1307, що містить таблиці скрембльованих алгоритмів 1308А, алгоритмів змішування 1308В, і алгоритмів шифрування 1308С. Крім початкових станів або інформації про стан, пов'язаний з пакетами даних, тобто, утримуваних у межах заголовка пакета або доставленого заздалегідь пакета даних, алгоритми використовуються для створення вхідних пакетів даних, не переносяться або утримуються всередині пакета, але присутні локально або всередині медіа-вузла $M_{a,d}$ або на безпечному сервері, до якого у медіа-вузла є доступ. Дані алгоритми, що зберігаються локально як розподілені (спільні) секрети для спеціальної ділянки 1302А, у цьому випадку зони Z1, поєднуються з кожним медіа-вузлом однієї й тієї ж зони. Знаючи час і стан, коли пакет даних був створений, медіа-вузол $M_{a,d}$ здатний визначити, як кожні з пакетів 1262В, 1262С і 1262Н були створені і як скасувати процес відновлення даних простого тексту кожного з пакетів 1262В, 1262С і 1262Н. Наприклад, як дешифрувати зашифрований пакет, дескремблювати скрембльований пакет і т.д. Використання спільних секретів, а також те, як вони розподіляються, описано далі в додатку.

Ключі дешифрування 1306А і 1306В функціонують разом з обраним алгоритмом шифрування 1309С для дешифрування зашифрованого тексту у простий текст. Зокрема, алгоритм шифрування 1309С представляє послідовність математичних кроків, які можуть використовуватися для перетворення пакета даних із зашифрованого тексту у простий текст. Ключі дешифрування 1306А і 1306В потім вибирають спеціальну комбінацію тих кроків, які використовуються в дешифруванні пакета, кожний з яких відповідає стану та часу, коли були

востаннє зашифровані вхідні пакети даних. Якщо обидва вхідні пакети були зашифровані одночасно, необхідний тільки один ключ дешифрування. Хоча наведене вище посилання стосується алгоритму "шифрування" 1309C, буде зрозуміло, що алгоритм шифрування визначає зворотне – алгоритм дешифрування. За винятком певних типів шифрування, що використовують "асиметричні" ключі, більшість алгоритмів є симетричними, означаючи, що зворотний алгоритм, використовуваний для зашифровування та скремблювання пакета даних, може бути використаний для дешифрування або дескремблювання пакета даних і відновлення вихідного контенту. На специфічному прикладі, показаному на рис. 92A, для кожного моменту часу та стану, що відповідають вхідним пакетам даних 1262B, 1262C і 1262H, виходам селектору 1307, необхідний обраний алгоритм шифрування 1309C для дешифрування вхідного пакета, необхідний обраний алгоритм скремблювання 1309A для дескремблювання вхідного пакета, і необхідний обраний алгоритм 1309B для об'єднання пакетів у певному порядку та видалення "сміттєвих" даних. Таким чином, алгоритми шифрування, скремблювання та змішування, обрані селектором 1307, використовуються для здійснення операцій дешифрування, дескремблювання та змішування, відповідно, на пакетах даних 1262B, 1262H і 1262C сервером комп'ютера 1220D на медіа-вузлі $M_{a, d}$. Тому обробка даних на медіа-вузлі залежить від часу та стану вхідних пакетів даних і від обраних алгоритмів. Наприклад, обраний алгоритм змішування 1309B може впорядкувати вхідні пакети, які будуть об'єднані в один довгий пакет у послідовності спадання часу зважаючи на те, коли був створений пакет. Наприклад, найдавніший пакет розміщується на початку довгого пакета, а найновіші пакети даних - наприкінці. Або, навпаки, дані можуть бути наведені в хронологічному порядку сегментів даних, як показано в пакеті даних 263B, тобто сегмент даних 1B перед 1C, сегмент даних 1C перед 1F, і т.д. Тому обробка вхідних пакетів даних потребує інформації про час і стан, пов'язані зі створенням вхідних пакетів, а не поточний час та поточний стан. Без попереднього перехоплення інформації про стан і час вхідних пакетів навіть хакер, який має доступ до таблиць алгоритму та поточних станів, не зможе декодувати, дешифрувати, прочитати або інтерпретувати вхідні дані на медіа-вузлі. Як було зазначено вище, вибір алгоритмів селектора 1307 і генерації ключа за допомогою генераторів ключів 1305A і 1305B залежить від географічного положення або "підмережі", де пакети даних були створені, яка зображена на рисунку як зона інформації 1302A і позначена як "Зона Z1". Використання зон буде описано далі в цьому документі.

На противагу попередній ілюстрації, яка показує керування вхідними пакетами даних, керування вихідним пакетом даних, зображеним на рис. 92B, залежить не від колишніх часів і станів, а від поточного часу і його відповідного стану. Відповідно до зображення, у момент часу t_3 і його відповідного стану 1301C генератор початкових станів 1303 виробляє початкові стани 1304C, використані селектором 1307 для вибору відповідних алгоритмів для розділення, скремблювання та шифрування з таблиць алгоритмів скремблювання 1308A, алгоритмів змішування 1308B та алгоритмів шифрування 1308C. Оскільки алгоритм змішування 1308B зазвичай є симетричною функцією, зворотний алгоритм, що застосовується для змішування, використовується для розділення. У цьому випадку, розділяючи довгий пакет даних на множинні пакети, готові до передачі даних. У двоканальному або триканальному зв'язку пункти призначення для всіх згенерованих пакетів передаються вузлу з сигнального сервера, що керує маршрутизацією пакета. В одноканальному зв'язку медіа-вузли самі повинні емулювати функцію сигнального сервера, відображаючи власний маршрут між операторами, які викликають.

Та сама інформація про стан 1301C подається в генератор ключів E_3 1305C для створення ключа E 1306C, необхідного для шифрування вихідних пакетів даних, і в генератор початкових чисел 1303 для створення початкових чисел 1304C, використовуваних для вибору алгоритму шифрування 1309C з таблиці 1308C. Ключ E_3 функціонує разом з обраним алгоритмом шифрування 1308C для шифрування звичайного тексту в зашифрований текст. Зокрема, алгоритм шифрування представляє послідовність математичних кроків, які можуть використовуватися для перетворення пакета даних зі звичайного тексту в один з мільйонів, більйонів або трильйонів можливих зашифрованих результатів. Потім ключ шифрування вибирає специфічну комбінацію тих кроків, які застосовуються для шифрування пакета.

У схемі шифруванні з симетричними ключами, в якій використовується алгоритм AES (з англ. Advanced Encryption Standard - покращений стандарт шифрування), описаний в http://en.wikipedia.org/wiki/advanced_encryption_standard, ключ, який використовується для шифрування файлу, також використовується для його дешифрування. У такому разі варто генерувати ключ локально як поділюваний секрет, наявний всередині кожного вузла, наприклад, використовуючи генератор ключів E_3 1305C. Якщо симетричний ключ повинен бути поданий на

медіа-вузол мережі, варто доставити ключ через відмінні від медіа канали зв'язку. Наприклад, пакети даних і контент. Багатоканальний зв'язок обговорюється далі в цьому додатку.

Іншим засобом для поліпшення безпечної доставки симетричних ключів є їхня доставка на медіа-вузли в момент часу, незалежного від комюніке. Наприклад, на один тиждень раніше дешифрувати ключ з іншим рівнем шифрування або розділити ключі на дві частини, доставлені в різні моменти часу. Інший метод включає використання алгоритму поділу ключа в генераторі ключа Ез 1305С, де частина ключа залишається локально на кожному медіа-вузлі як поділюваний секрет, тобто не представлений у мережі, а інша частина знаходиться у відкритому доступі. Безпека підвищується, і кіберпірат не може визначити, скільки бітів має ключ, тому що він бачить тільки частину ключа. Незнання довжини ключа унеможливорює знаходження правильного ключа, тому що довжина ключа та кожний елемент ключа повинні вгадуватися.

У разі використання асиметричного шифрування або алгоритму шифрування з відкритим ключем, генератор ключів Ез 1305С одночасно генерує пари ключів – одну для шифрування, іншу для дешифрування, заснованого на стані 1301С або на часі t_3 . Ключ дешифрування зберігається в медіа-вузлі як поділюваний секрет, тоді як ключ шифрування безпечно та відкрито направляється на медіа-вузол, готовий відправити пакет даних на нього. Складністю використання симетричних ключів мережі в режимі реального часу є те, що ключ шифрування потрібно згенерувати та направити на всі медіа-вузли перед запуском пакета даних, який містить контент на медіа-каналі, інакше пакет даних може надійти до надходження ключа для його дешифрування і дані стануть застарілими, тобто їх пізно використовувати. Опис використання та керування асиметричними та відкритими ключами шифрування доступний у багатьох текстах і в онлайн-публікаціях, таких як http://en.wikipedia.org/wiki/public-key_cryptography. В той час як технологія шифрування з відкритим ключем - загальновідома, описаний додаток включає унікальну інтеграцію криптографії в мережу реального часу та систему зв'язку.

Алгоритми, числові значення та ключі шифрування є згенерованими для поточної зони підмережі 1307А, у цьому випадку зони Z1. Ключ шифрування 1306С, заснований на даній зоні та поточному часі t_3 , разом з обраним алгоритмом поділу 1309В, обраним алгоритмом скремблювання 1309А і обраним алгоритмом шифрування 1309С, відправляється на медіа-вузол М_a, розміщений на комп'ютерному сервері 1220D для створення двох виходів. Це вихід пакета даних 1263С, що включає незв'язані сегменти даних, відправлені далі в момент часу t_3 , і вихід пакета даних 1263В, що містить сегменти даних 1В, 1С і 1F, що зберігаються до моменту часу t_4 до маршрутизації на наступному медіа-вузлі. Інструкції щодо того, зберігати пакет даних або сегмент даних тимчасово чи відправляти його на наступний медіа-вузол, можуть бути доставлені одразу на медіа-вузол декількома способами. В одному випадку, вхідний пакет даних може містити інструкції про час, протягом якого вони будуть зберігатися, і інформацію про умови, за яких їх можна направляти далі. В іншому випадку, сервер, тобто інший канал зв'язку, може надавати медіа-вузлу інструкції про те, що робити. Використання сигнальних серверів в багатоканальному безпечному зв'язку описано далі в цьому додатку.

Як показано на рис. 93, для того щоб вибрати алгоритм з таблиці алгоритмів, які можуть бути алгоритмами скремблювання/дескремблювання, шифрування/ дешифрування або змішування/розділення, селектор 1307 повинен шукати за списком алгоритмів і адресами пам'яті 1308D, порівнюючи їх з адресом 1304D, згенерованим генератором початкових чисел 1303 у момент часу t_x і у відповідному стані 1301D. Коли генерований стан адреси 1304D відповідає елементу в таблиці алгоритмів 1308D, обраний алгоритм 1309D виведенняється з процедури пошуку для використання. Наприклад, якщо генератор початкових чисел 1303 генерує адресу 1304D, що має значення "356", тоді селектор 1307 виявить відповідний елемент з таблиці, а саме "зрушення фаз 2" і виведе його як обраний алгоритм 1309D.

З метою запобігання систематичному відстеженню, список алгоритмів та їхні відповідні адреси пам'яті регулярно переставляються, наприклад, щодня або щогодини так, що та сама адреса не задіює той самий алгоритм, навіть коли він випадково повторюється. Як показано на рис. 94, таблиця алгоритмів для дня 318 у зоні Z1 включає таблицю адрес алгоритму 1308D, використовуваного для скремблювання та дескремблювання в зоні Z1 на 318 день, тобто таблицю адрес алгоритму 1308Е, використовуваного для розділення або змішування пакетів даних у зоні Z1 на 318 день, і, таблицю адрес алгоритму 1308F, використовуваного для шифрування або дешифрування в зоні Z1 на 318 день. Потім на задану дату події 1311 і часу 1310 повторно призначають адресну операцію 1312 і перетасовують, тобто змішують список алгоритмів і адрес, створюючи три нові таблиці, що включають таблицю адрес алгоритму 1308G для скремблювання та дескремблювання в зоні Z1 на 319 день, друга таблиця – таблиця адрес

алгоритму 1308H для змішування та розділення в зоні Z1 на 319 день, і третя таблиця для шифрування та дешифрування в зоні Z1 на 319 день, тобто таблиця адрес алгоритму 1308J. Відповідно до зображення, наприклад, на 318 день, "режим перегрупування 5" має відповідну адресу пам'яті 359, але через день адреса змінюється на 424. Таким чином, таблиця перетворення між адресами та алгоритмами перетасовується, щоб уникнути злому.

Зони і мости. Для глобальної комунікації із запобіганням доступу хакерів та кіберпіратів до всієї хмари захищеної динамічної мережі зв'язку і протоколу (SDNP) та мережі, іншим способом реалізації даного винаходу є комунікаційна мережа SDNP, поділена на "зони". У цьому випадку зона є частиною мережі, тобто "підмережею", в якій кожна зона має власні налаштування, команди керування та безпеки, включаючи точні і окремі алгоритми, а також алгоритмічні таблиці, що визначають змішування і розділення, шифрування і дешифрування, а також скремблювання і дескремблювання, використовувані в зоні, окремі ключі шифрування та визначені початкові числові значення. Зазвичай, комунікаційні сервера, керовані програмним забезпеченням SDNP, в межах однієї зони мають однакові налаштування і працюють в незалежному від зони розміщенні режимі.

Кожна підмережа може складатися з різних серверних хмар, в яких може працювати програмне забезпечення SDNP, встановлене різними Інтернет-провайдерами або хостинговими компаніями, наприклад, Microsoft, Amazon, Yahoo, або воно може складатися з особистих хмар чи трансляторів мережевої адреси (NAT), наприклад, орендованих особистих хмар, що складаються з темного оптоволокна з заданою смугою частот. Також вигідно взаємодіяти з операторами зв'язку, які надають послуги з організації "останньої милі", наприклад, компанія Comcast в Північній Каліфорнії (місцева телефонна мережа, що надає послуги загального зв'язку), а також місцевими підприємствами стільникового зв'язку як з окремими зонами. Основною перевагою, яка забезпечується при поділі на зони, є те, що при найгіршому сценарії, коли дуже здібний кіберпірат тимчасово зламує налаштування безпеки SDNP, можна обмежити географічно їх атаку меншою підмережею, не даючи доступу до наскрізної передачі даних. Фактично, зони характеризуються можливістю пошкодження унаслідок кібератаки.

Приклад використання зон проілюстровано на рис. 95A, де хмара 1114, що складається з комп'ютерів-серверів, на яких встановлено програмне забезпечення SDNP, поділена на дві підмережі, а саме підмережу 1318A, яка складається з "зони Z1", і мережу 1318C, яка складається з "зони Z2". Відповідно до зображення, підмережа 1318A складається з медіа-вузлів $M_{a, w}$, $M_{a, s}$, $M_{a, j}$, $M_{a, b}$, $M_{a, q}$, і $M_{a, f}$ разом із $M_{b, d}$ та $M_{b, h}$, тоді як підмережа 1318C складається з медіа-вузлів $M_{c, j}$, $M_{c, n}$, $M_{c, v}$, $M_{c, u}$, і $M_{c, z}$, до неї віднесені також медіа-вузли $M_{b, d}$ і $M_{b, h}$. Тоді як медіа-вузли з головним нижнім індексом "a", тобто M_a , унікальні для зони Z1, медіа-вузли з головним нижнім індексом "c", тобто M_c , унікальні для зони Z2, медіа-вузли $M_{b, d}$ і $M_{b, h}$, що знаходяться на комп'ютерах-серверах 1220 D і 1220H, унікальні тим, що вони належать до обох підмереж 1318A і 1318C. Програмне забезпечення SDNP, що працює на комп'ютерах-серверах 1220D і 1220H, має розуміти, як обмінюватися інформацією з іншими медіа-вузлами як у зоні Z1, так і в зоні Z2. Такі пристрої функціонують як "мости" між двома підмережами і за необхідності повинні передавати дані з захищених файлів у зоні Z1 до даних, сформатованих відповідно до захищених файлів у зоні Z2, і навпаки.

Функцію передачі, що здійснюється в мостовому медіа-вузлі $M_{b, d}$, проілюстровано на рис. 95B, на якому зображено потік даних із зони Z1 в зону Z2, де операція DUM (з англ. Decryption-Unscrambling-Splitting – дешифрування-дескремблювання-розділення) 1210, яка виконується на мостовому комп'ютері-сервері 1220D, на якому розташований медіа-вузол $M_{b, d}$, забезпечує дешифрування, дескремблювання та змішування для підмережі 1318A, зона Z1, використовуючи алгоритмічні таблиці 1308K, для створення довгого пакету, який вона перетворить в операцію SSE 1213, те саме відбувається в медіа-вузлі $M_{b, d}$, що виконує розділення, скремблювання та шифрування для підмережі 1318C, зона Z2, з використанням алгоритмічних таблиць 1308L. Повнодуплексна версія мостового медіа-вузла $M_{b, d}$ показана на рис. 95C, де проілюстровано, що мостовий медіа-вузол $M_{b, d}$ виконує двосторонню передачу даних та їх передачу з зони Z1 в зону Z2 і навпаки. Для передачі даних із зони Z1 в зону Z2 мостовий комп'ютер-сервер 1220D з програмним забезпеченням SDNP, на якому знаходиться медіа-вузол $M_{b, d}$, виконує DUM-операцію 1210 з пакетами даних при їх виході з зони Z1 (підмережа 1318A) з подальшою SSE-операцією 1210 з пакетами даних, що входять у зону Z2 (підмережа 1318C). Навпаки, для передачі даних із зони Z2 в зону Z1 мостовий комп'ютер-сервер 1220D з програмним забезпеченням SDNP виконує DUM-операцію 1210 з пакетами даних при їх виході з зони Z2 (підмережа 1318C) з подальшою SSE-операцією 1213 з пакетами даних, що входять у зону Z1 (підмережа 1213A). Усі чотири операції з даними, що виконуються

на мостовому медіа-вузлі $M_{b,d}$, виконуються в програмному забезпеченні, встановленому на тому самому серверному хості, в даному випадку це комп'ютер-сервер 1220D.

Повністю інтегрований медіа-вузол $M_{b,d}$ моста з SDNP, показаний на рис. 95C, виконує як DUM-, так і SSE-операції у двох різних зонах, тобто в зонах Z1 і Z2, все це виконується на комп'ютері-сервері 1202D. Таке повністю інтегроване рішення може бути реалізоване лише тоді, коли обидві з'єднані підмережі знаходяться у одного й того ж Інтернет-провайдеру або в одній хмарі. Водночас, якщо підмережі знаходяться в різних хмарах або зберігаються у різних провайдерів послуг, як показано у вигляді підмереж 1318A і 1318C на рис. 95D, то необхідно передбачити комунікаційний міст між двома комп'ютерами-серверами, які не знаходяться в одній і тій самій хмарі. Відповідно до зображення, мостовий комунікаційний зв'язок 1316B з'єднує мостовий медіа-вузол $M_{b,h}$ з SDNP, що працює в зоні Z1, з мостовим медіа-вузлом $M_{b,u}$, що працює в зоні Z2, проте зона Z1 функціонує в хмарі 1114, тоді як зона Z2 функціонує в хмарі 1315. Використання того самого методу, показаного раніше на рис. 95C, стає проблематичним за наявності декількох хмар, оскільки мостовий комунікаційний зв'язок 1316B, що переміщається між хмарами, стає незахищеним і чутливим до прочитання і кібератак. На рис. 95E показано випадок, коли DUM-операція, що виконується мостовим медіа-вузлом $M_{b,h}$, який знаходиться на комп'ютері-сервері 1220H в підмережі 1318A і зоні Z1, направляє пакети даних через мостовий комунікаційний зв'язок 1316B в мостовий медіа-вузол $M_{b,u}$, що знаходиться на комп'ютері-сервері 1220U, в підмережу 1318C і зону Z2 для передачі, проте через те, що інформація, яка передається, являє собою дешифрований і дескрембльований довгий пакет з DUM-операції мостового медіа-вузла $M_{b,h}$, переміщення з однієї хмари в іншу залишається незахищеним і схильним до кібератак.

Вирішенням цієї проблеми є залучення двох повнодуплексних мостових інтерфейсних медіа-вузлів, по одному в кожній хмарі, як показано на рис. 95F, із захищеною передачею даних між інтерфейсами. При передачі даних із зони Z1 в зону Z2 пакети даних, які виходять із зони Z1, що знаходиться в підмережі 1318A, перетворюються в одноканальні дані зони Z2, при цьому передбачено їх скремблювання і шифрування. Ця функція потребує наявності доступу медіа-вузла $M_{b,d}$ як до зони Z1, так і до зони Z2, початкових значень, ключів шифрування, алгоритмічних таблиць та інших засобів безпеки. Вся обробка проводиться на комп'ютері-сервері 1220D, встановленому в підмережі 1318A, а не в зоні Z2 хмари призначення. Після цього захищені дані передаються з мостового інтерфейсного медіа-вузла $M_{b,d}$ в підмережі 1318A на мостовий інтерфейсний медіа-вузол $M_{b,u}$ в підмережі 1318C з використанням безпечного мостового комунікаційного зв'язку 1316A. Після надходження на мостовий інтерфейсний медіа-вузол $M_{b,u}$ пакети даних обробляються відповідно до інформації, що зберігається в зоні Z2, і прямують звідти в підмережу 1318C.

Навпаки, при передачі даних із зони Z2 в зону Z1 пакети даних, що надходять із зони Z2 і підмережі 1318C на медіа-вузол $M_{b,u}$, перетворюються в одноканальні дані зони Z1, при цьому передбачено їх скремблювання і шифрування. Ця функція потребує наявності доступу медіа-вузла $M_{b,d}$ як до зони Z1, так і до зони Z2, початкових значень, ключів шифрування, алгоритмічних таблиць та інших засобів безпеки. Вся обробка здійснюється на комп'ютері-сервері 1220U, встановленому в підмережі 1318C, а не в зоні Z1 хмари призначення. Після цього захищені дані передаються з мостового інтерфейсного медіа-вузла $M_{b,u}$ в підмережі 1318C на мостовий інтерфейсний медіа-вузол $M_{b,d}$ в підмережі 1318A з використанням безпечного мостового комунікаційного зв'язку 1316C. Після надходження на мостовий інтерфейсний медіа-вузол $M_{b,d}$ пакети даних обробляються відповідно до інформації, що зберігається в зоні Z1, і прямують звідти в підмережу 1318A. Хоча безпечні мостові комунікаційні зв'язки 1316A і 1316C зображені у вигляді окремих ліній, лініями є окремі канали передачі зв'язку на 3-му рівні мережі, вони не призначені для збереження відповідності окремим дротам, кабелям або каналам передачі даних на програмно-апаратному комплексі або опису 1-го фізичного рівня. Замість цього приймальний мостовий вузол може передавати дані з зони відправки Z1 в зону призначення Z2, якщо на приймальному мостовому вузлі зберігаються спільні для зон Z1 і Z2 розподілені секрети.

Функціонування шлюзу SDNP. У попередньому розділі представлено опис "моста" у вигляді медіа-вузла або пари медіа-вузлів, що забезпечують обмін інформацією між окремими підмережами, мережами або хмарами. Аналогічно, "шлюзний медіа-вузол" SDNP, описаний у цьому розділі, забезпечує комунікаційний зв'язок між хмарою SDNP і пристроєм клієнта, наприклад, стільниковим телефоном, автомобілем, планшетом, ноутбуком або Інтернет-пристроєм. Функціонування шлюзового медіа-вузла показано на рис. 96A, де комп'ютер-сервер 1220F у хмарі 1114 з SDNP, де знаходиться медіа-вузол $M_{b,f}$ з SDNP, функціонує як шлюзний медіа-вузол з SDNP між підмережею 1318A і "останньою милею" 1318D з планшетом 33. На

відміну від підмережі 1318A, "остання миля" 1318D може здійснюватися через Інтернет, особисту хмару, кабельне телевізійне з'єднання або шляхом стільникового зв'язку. У з'єднанні користувача з вузлом доступу Інтернет-провайдера не можна точно керувати маршрутизацією, як це було в підмережі 1318A. Наприклад, шлюзний медіа-вузол $M_{b,f}$ з'єднаний з сервером 65A з'єднаним пристроєм 1317, проте за межами цієї точки маршрутизація в базовій станції Wi-Fi загального користування 100 керується місцевими IP-маршрутизаторами. Радіозв'язок Wi-Fi 29 від антени Wi-Fi 26 до планшета 33 також керується місцевим пристроєм, часто розташованим в аеропорту, готелі, кафе, будівлі органу влади, амфітеатрі або в іншому громадському місці.

Замість цього "остання миля" може бути д्रो́тяним з'єднанням з базовою станцією мережі передачі даних (LTE) 17 з радіозв'язком 28 між антеною 18 і планшетом 33. Через його нечітку маршрутизацію і наявність доступу бажано не ділитися налаштуваннями безпеки і секретами, використовуваними в хмарі SDNP, з пристроями, використовуваними для маршрутизації з'єднань "останньої милі". "Остання миля" 1318d не має доступу до інформації в зоні Z1, замість цього для керування налаштуваннями безпеки воно використовує окрему зону U2. Для з'єднання між хмарою 1114 і "останньою милею" шлюзний медіа-вузол $M_{b,f}$ має бути забезпечений доступом до налаштувань безпеки як зони Z1, так і зони U2, полегшуючи обмін інформацією між хмарним інтерфейсом 1320 і клієнтським інтерфейсом 1321. Для забезпечення безпечного з'єднання "останньої милі", на клієнті, в цьому випадку планшеті 33, також має бути встановлене клієнтське програмне забезпечення SDNP 1322. Шлюзний вузол з SDNP $M_{b,f}$ складається з хмарного інтерфейсу 1320, що полегшує обмін інформацією між медіа-вузлами всередині хмари 1114, і клієнтського інтерфейсу 1321, що полегшує обмін даними у з'єднанні "останньої милі". Як показано на рис. 96B, хмарний інтерфейс 1320 складається з двох ліній передачі даних, тобто SSE 1213 та DUM 1210. Клієнтський інтерфейс 1321, показаний на рис. 96C, також складається з двох ліній передачі даних, одна з яких призначена для передачі даних, що надходять із шлюзу клієнту, а інша – для передачі даних у зворотному напрямку від клієнта до шлюзу. Зокрема, передача даних від шлюзу до клієнта послідовно включає операцію з розділення 1106, виконувану по одному маршруту, використовувану для введення "сміттєвих" даних у потік інформації з подальшим скремблюванням пакету 926 і що закінчується шифруванням 1026. У зворотному напрямку потік даних від клієнта до шлюзу включає в себе послідовно дешифрування 1302, дескремблювання пакету 928, а також операцію змішування 1089, виконувану по одному маршруту, виконувану з метою видалення "сміттєвих" даних з інформаційного потоку.

Ролі операцій змішування і розділення при передачі інформації одним каналом, наприклад, по лінії з'єднання "останньої милі", двоїсті. По-перше, що важливо, потік інформації в режимі реального часу ділять на безліч послідовних частин пакетів, кожна з яких має свої ідентифікаційні мітки і, можливо, змінну довжину, з метою протидії легкому виявленню. Тому сформований у результаті послідовний інформаційний потік потребує окремого утримання деяких частин пакетів даних при відправленні перших пакетів. Оскільки частота передачі даних у хмарі SDNP вимірюється сотнями гігабіт у секунду, формування серій відбувається практично миттєво, для цього потрібні наносекунди. У зв'язку "останньої милі" швидкість передачі даних нижча (хоча в сучасних системах вона залишається дуже високою), наприклад, близько двох гігабіт у секунду. Додаткові затримки не виникають, оскільки Wi-Fi, 4G/LTE, а також DOCSIS 3 завжди передають дані послідовно.

Друга необхідність забезпечення змішування для передачі по одному каналу обумовлена тим, що операція змішування по одному маршруту використовується також для того, щоб ввести "сміттєві" дані в частини пакетів різними способами для аналізу змішаної інформації способом, описаним вище для рис. 67J.

Як показано на рис. 96D, для безпечного обміну інформацією по "останній милі" клієнт повинен запустити клієнтське програмне забезпечення 1322. За наявності стільникового телефону або планшета це програмне забезпечення повинне працювати на операційній системі пристрою, наприклад, Android або iOS. У разі використання настільного комп'ютера або ноутбука клієнтське програмне забезпечення працює на операційній системі комп'ютера, наприклад, MacOS, Windows, Linux або Unix. У разі, якщо трапляється обмін інформацією з пристроєм користувача, наприклад, Інтернетом речей (IoT), на який не можна встановити клієнтське програмне забезпечення SDNP, як інтерфейс можна використовувати апаратний пристрій з вбудованим клієнтським програмним забезпеченням. Функції, пов'язані з обміном інформацією, що виконуються клієнтом 1322, полягають в обробці пакетів даних, які надходять, шляхом виконання операції дешифрування 1032, дескремблювання пакету 928, а також видалення "сміттєвої" інформації з використанням операції змішування по одному маршруту 1089 для відновлення корисної інформації, що міститься в пакетах. Далі вміст використовується

в додатках 1336, у тому числі дані, які використовуються в аудіо-файлах CODEC, файлах формату MPEG, зображеннях, файлах, що не є медіа-файлами, а також програмним забезпеченням.

5 Функції, пов'язані з обміном інформацією, що виконуються клієнтом 1322 з вихідними пакетами даних, полягають у додаванні "сміттєвих" даних при виконанні операції розділення по одному маршруту 1026, скремблюванні пакету 926, а також при виконанні остаточної операції шифрування 1026 з метою підготовки пакету даних до передачі по "останній милі" до шлюзу. У клієнтському програмному забезпеченні 1322 змішування 1089 по одному маршруту алгоритмічно видаляє "сміттєві" дані з потоку інформації, яка надходить, тоді як завдання операції розділення 1026 по одному маршруту полягає в додаванні "сміттєвих" даних у пакети даних.

10 Функціонування безпечного шлюзового вузла $M_{b, f}$ з SDNP детальніше показано на рис. 97A, де хмарний інтерфейс 1320 і клієнтський інтерфейс 1321 приймають ті пакети даних, які надходять від медіа-вузла $M_{a, h}$, виконуючи дешифрування, дескремблювання і змішування з використанням DUM-операції 1210 відповідно до налаштувань безпеки зони Z1, внаслідок чого з'являється пакет даних 1330, що являє собою дескремблований простий текст. Далі пакет даних 1330 передається на клієнтський інтерфейс 1321, який також працює у шлюзовому медіа-вузлі $M_{b, f}$, що введення пакети зі "сміттєвими" даними 1053 в ході виконання операції одномаршрутного розділення 1106, що використовується для введення "сміттєвих" даних 1053 в пакети даних, але з використанням налаштувань безпеки зони U2, а не налаштувань безпеки зони Z1, використовуваних хмарию. Далі пакет даних скремблюється за допомогою операції скремблювання 926, при цьому для створення пакету даних 1329 знову використовуються налаштування безпеки зони U2, які використовуються в "останній милі".

20 У показаному прикладі операція скремблювання 926 використовує алгоритм, з використанням якого дійсні сегменти даних скремблюються, проте кожним другим сегментом даних є сегмент "сміттєвих" даних. Далі операція шифрування виконується також у клієнтському інтерфейсі 1321, при цьому також використовуються налаштування безпеки зони U2 з метою створення зашифрованого тексту 1328, що надсилається. Поля даних можна шифрувати індивідуально і окремо від "сміттєвих" даних (як показано), альтернативний шлях реалізації полягає в тому, щоб цілісний пакет даних 1329 можна було шифрувати з метою отримання одного довгого шифрованого тексту. Зрештою, пакет шифрованих даних пересилається, тобто "експортується" клієнтові через єдиний канал передачі інформації.

25 Паралельно дані, отримані через єдиний канал зв'язку "останньої милі", що складаються з зашифрованого тексту 1327, дешифруються операцією дешифрування 1032 з використанням налаштувань безпеки U2, включаючи алгоритми, ключі дешифрування та ін., при цьому виходить пакет даних у вигляді скремблованого простого тексту 1326, що складається з сегментів скремблованих даних, змішаних із сегментами "сміттєвих" даних. В одному з способів реалізації цього винаходу пакети "сміттєвих" даних, що містяться в цьому вхідному пакеті даних 1326, не розміщуються в тих самих слотах, що і у вихідному пакеті даних 1329 у вигляді скремблованого простого тексту. Зокрема, у прикладі з вихідними даними кожен другий пакет складається з "сміттєвих" даних, тоді як у пакеті даних, що надходять, кожен 3-й і 4-й слот, а також уся їх сукупність містять "сміттєві" дані.

40 Далі пакет даних 1326 у вигляді скремблованого простого тексту обробляється з використанням налаштувань безпеки зони U2 шляхом проведення операції дескремблювання пакетів 928 з подальшим здійсненням операції змішування 1089 з метою відновлення вихідного порядку даних і видалення пакетів "сміттєвих" даних, тобто видалення з 1053 "сміттєвих" даних, що призводить до отримання пакету дешифрованих і дескремблованих даних 1325. Далі цей пакет даних передається з клієнтського інтерфейсу 1321 у хмарний інтерфейс 1320 для проведення операцій розділення, скремблювання і шифрування, характерних для хмари, з використанням SSE-операції 1213 перед передачею даних, отриманих діленням на фрагменти, в різних пакетах даних решітчастої маршрутизації до медіа-вузла $M_{b, h}$ та в інші точки.

45 Як показано нижче на рис. 97B, шлюзний медіа-вузол SDNP $M_{b, f}$ використовує програмне забезпечення для полегшення повнодуплексного обміну інформацією як у хмарному інтерфейсі 1320 відповідно до налаштувань безпеки зони Z1, так і в клієнтському інтерфейсі 1321 відповідно до налаштувань безпеки зони U2. З'єднання "останньої милі" 1355 від клієнтського інтерфейсу 1321 до планшета 33 через базову станцію передачі даних (LTE) 27, радіовежу LTE 18, а також радіозв'язок 28 безпечне, оскільки інформація, що передається, скремблована і зашифрована, а сміттєві дані введені в пакети даних. Для інтерпретації пакетів даних, що надходять, і забезпечення можливості безпечного реагування на пристрої клієнта (в даному

випадку це планшет 1322) має працювати прикладне програмне забезпечення 1322, сумісне з SDNP.

Обробка пакетів даних у клієнтському інтерфейсі з SDNP показана детальніше на рис. 98, де клієнтський вузол $C_{2,1}$ здійснює безпечний обмін інформацією з шлюзним вузлом $M_{b, d}$ з SDNP шляхом повнодуплексного обміну даними між клієнтським інтерфейсом 1321 і клієнтом з SDNP 1322, при якому обидва знаходяться в безпечній зоні U2. У ході операції пакети даних, що надходять з клієнтського інтерфейсу 1321, підлягають дешифруванню при проведенні операції дешифрування 1032, дескремблювання у ході операції дескремблювання 928, а також проводиться видалення "сміттєвих" даних з використанням операції розділення 1089; усе це здійснюється перед обробкою додатками 1336. І навпаки, відправка пакетів даних з додатків 1336 супроводжується операцією змішування 1026 з метою вкладення "сміттєвої" інформації, далі проводиться скремблювання шляхом проведення операції скремблювання 926, шифрування проведенням операції шифрування 1106; потім дані надходять на клієнтський інтерфейс 1321.

З використанням методів, описаних у цьому розділі, безпечний обмін інформацією між двома і великою кількістю клієнтів, що статично або динамічно маршрутизується через решітчасту мережу, може передбачати будь-яку комбінацію алгоритмів змішування, розділення, шифрування і скремблювання, керування роботою яких у різних зонах здійснюється за допомогою окремих ключів, визначених початкових числових значень, а також різномірних, пов'язаних з безпекою секретів. Як показано на рис. 99A, решітчаста мережа, до складу якої входять комп'ютери-сервери 1118, на яких працюють медіа-вузли з SDNP на основі програмного забезпечення, включає комп'ютери-сервери 1220F і 1220D, на яких знаходяться шлюзові медіа-вузли $M_{b, f}$ і $M_{b, d}$. Безпека в підмережі 1318A визначається налаштуваннями безпеки для зони Z1. Шлюзний медіа-вузол $M_{b, d}$ підключається до клієнтського вузла $C_{1,1}$, що знаходиться на зовнішньому пристрої (в даному випадку це стільниковий телефон 32), доступ до якого здійснюється по "останній милі" 1318E. Безпека в "останній милі" визначається налаштуваннями безпеки для зони U1. Аналогічно, шлюзний медіа-вузол $M_{b, f}$ підключається до клієнтського вузла $C_{2,1}$, що знаходиться на планшеті 33, крім того, підключення здійснюється по "останній милі" 1318D. Безпека в "останній милі" визначається налаштуваннями безпеки для зони U2.

Відповідно до зображення, обмін інформацією з використанням операції шифрування 1339, показаної у вигляді замка, забезпечує безпеку в усій мережі і в лініях "останньої милі". Для забезпечення безпеки "останньої милі" шифрування обов'язково проводиться на пристроях користувача. Замість цього пакети можна шифрувати повторно або шифрувати двічі з використанням шлюзних медіа-вузлів; інший варіант реалізації передбачає дешифрування і повторне шифрування на кожному медіа-вузлі в решітчастій транспортній мережі. Один із варіантів реалізації, описаних у даному розділі, спрямований на сприяння багаторівневій безпеці. Наприклад, на рис. 99A у лініях зв'язку "останньої милі" 1318D та 1318E передбачено лише шифрування, тобто однорівнева або одновимірна безпека. У мережі 1318A обмін інформацією здійснюється з забезпеченням двовимірної або дворівневої безпеки, що передбачає комбінування шифрування з роботою решітчастої системи, яка передбачає статичний поділ, передачу декількома маршрутами і змішування. У разі якщо налаштування безпеки з часом змінюються, наприклад, "у динамічному режимі" при проходженні пакетів даних по мережі, реалізують додатковий рівень безпеки, тобто двовимірну або дворівневу безпеку в лінії зв'язку "останньої милі" і тривимірну безпеку в хмарі з SDNP.

Як показано на рис. 99B, додаткове проведення скремблювання в мережі 1318A підвищує рівень безпеки до вищого рівня багаторівневої безпеки, що передбачає комбінацію передачі інформації по решітчастій мережі з шифруванням і скремблюванням. Зокрема, при такому підході обмін інформацією між клієнтським вузлом $C_{2,1}$ і клієнтським вузлом $C_{1,1}$ включає додаткове проведення операції скремблювання 926 на шлюзовому медіа-вузлі $M_{b, f}$ і операції дескремблювання 928 на шлюзовому медіа-вузлі $M_{b, d}$. Після передачі через мережу 1318A пакети даних розшифровують, дескремблюють шляхом проведення операції дескремблювання 928, а потім змішують. Хоча цей підхід забезпечує багатовимірну безпеку в мережі 1318A, він не забезпечує багаторівневої безпеки в "останній милі", в якому, зважаючи на передачу даних одним каналом, без скремблювання їх безпека забезпечується лише шифруванням.

Інший спосіб реалізації цього винаходу, який показано на рис. 99C, поширює технологію забезпечення багаторівневої безпеки шляхом комбінування шифрування і кодування як для мережі 1318A, так і для зв'язку "останньої милі" 1318D до клієнтського вузла $C_{2,1}$. Обмін інформацією між клієнтським вузлом $C_{2,1}$ і клієнтським вузлом $C_{1,1}$ передбачає проведення операції скремблювання 926 в клієнтському вузлі $C_{2,1}$ і операції дескремблювання 928 у

шлюзному медіа-вузлі $M_{b, d}$. Обмін інформацією між клієнтським вузлом $C_{1,1}$ і клієнтським вузлом $C_{2,1}$ здійснюється з проведенням операції скремблювання в шлюзному медіа-вузлі $M_{b, d}$ і операції дескремблювання 928, здійснюваною в клієнтському вузлі $C_{2,1}$. Проте в лінії зв'язку "останньої милі" 1318E між клієнтським вузлом $C_{1,1}$ і шлюзним медіа-вузлом $M_{b, d}$, передбачено лише шифрування. Подібний випадок може мати місце, якщо в клієнтському вузлі $C_{2,1}$ працює захищена прикладна програма, сумісна з SDNP, але в клієнтському вузлі $C_{1,1}$ передбачено лише стандартне шифрування.

Інший спосіб реалізації цього винаходу, який показано на рис. 99D, передбачає реалізацію скремблювання для багатовимірної безпеки при обміні інформацією між клієнтами, тобто між кінцевими точками. Обмін інформацією між клієнтським вузлом $C_{2,1}$ і клієнтським вузлом $C_{1,1}$ передбачає виконання додаткової операції скремблювання 926 у клієнтському вузлі $C_{2,1}$ і операції дескремблювання 928 у клієнтському вузлі $C_{1,1}$. Обмін інформацією між клієнтським вузлом $C_{1,1}$ і клієнтським вузлом $C_{2,1}$ передбачає проведення додаткової операції скремблювання 926 у клієнтському вузлі $C_{1,1}$ і операції дескремблювання 928 у клієнтському вузлі $C_{2,1}$. У ході виконання клієнтський вузол $C_{1,1}$ здійснює скремблювання і шифрування вихідних пакетів даних, а також здійснює дешифрування і дескремблювання даних, що надходять, за допомогою програмного забезпечення SDNP, що працює на стільниковому телефоні 32. Аналогічно, клієнтський вузол $C_{2,1}$ здійснює скремблювання і шифрування пакетів даних, що надсилаються, а також здійснює дешифрування і дескремблювання даних, що надходять, за допомогою програмного забезпечення SDNP, яке працює на планшеті 33. Разом вони полегшують безпечний обмін інформацією між кінцевими точками із забезпеченням двохарової або двовимірної безпеки, тобто здійснюючи шифрування і скремблювання в лініях зв'язку "останньої милі" 1318D та 1318E, а також тривимірну або трирівневу безпеку в межах решітчастої сітки 1318A шляхом передачі через решітчасту мережу численними маршрутами. У разі якщо налаштування безпеки з часом змінюються, наприклад, "у динамічному режимі" при проходженні пакетів даних по мережі, реалізують додатковий рівень безпеки, тобто тривимірну або трирівневу безпеку в "останній милі" і чотирирівневу безпеку в хмарі з SDNP.

Можливим слабким місцем такого варіанту виконання є те, що методи скремблювання і початкові числові значення, використовувані клієнтом, використовуються також для забезпечення безпеки хмари SDNP. Як наслідок, налаштуваннями безпеки для зон U2, Z1 і U1 доводиться ділитися, що пов'язане з ризиком для всієї мережі і маршрутизації від кібератак на "останній милі". Один із відомих методів протидії незахищеності налаштувань безпеки хмари показано на рис. 99E, де "остання миля" 1318D передбачає скремблювання з використанням налаштувань безпеки зони U2, тоді як у хмарі для скремблювання інформації використовуються налаштування безпеки зони Z1. У даному прикладі клієнтський вузол $C_{2,1}$, що працює у вигляді додатка на планшеті 33, полегшує скремблювання 926 відповідно до налаштувань безпеки зони U2. Шлюзний медіа-вузол $M_{b, f}$, що знаходиться на комп'ютері-сервері 1220F, здійснює дескремблювання пакету даних, що надходить, з використанням налаштувань безпеки зони U2, після чого виконує повторне скремблювання пакетів даних, використовуючи налаштування безпеки зони Z1 для передачі інформації по решітчастій сітці 1318A. При такій організації налаштування безпеки зони Z1 хмари в "останній милі" 1318D ніколи не розкриваються.

Спосіб подальшого підвищення багаторівневої безпеки показано на рис. 99F, де скремблювання і шифрування здійснюють з використанням різних налаштувань безпеки у трьох різних зонах – "остання миля" 1318D, що з'єднує клієнтський вузол $C_{2,1}$ із шлюзним вузлом $M_{b, f}$, на якому використовуються налаштування безпеки зони U2, решітчастої сітки 1318A, до складу якої входять шлюзні медіа-вузли $M_{b, f}$ і $M_{b, d}$ і в якій використовуються налаштування безпеки зони Z1, а також "остання миля" 1318E, що сполучає шлюзний медіа-вузол $M_{b, d}$ з клієнтським вузлом $C_{1,1}$, на якому використовуються налаштування безпеки зони U2. Цей підхід забезпечує безпеку з'єднання між кінцевими точками і наскрізним шифруванням, наскрізне скремблювання, а також решітчасту маршрутизацію у хмарі, що є дворівневою або двовимірною безпекою в "останній милі", і тришарову або тривимірну безпеку в хмарі. У разі якщо налаштування безпеки з часом змінюються, наприклад, "у динамічному режимі" при проходженні пакетів даних по мережі, реалізують додатковий рівень безпеки, тобто тривимірну або дворівневу безпеку в "останній милі" і чотирирівневу безпеку в хмарі з SDNP.

При обміні інформацією між клієнтським вузлом $C_{2,1}$ і клієнтським вузлом $C_{1,1}$, тобто між планшетом 33 і стільниковим телефоном 32, додаток SDNP, що працює на клієнтському вузлі $C_{2,1}$, скремблює вихідний пакет даних, виконуючи операцію скремблювання 926 при налаштуваннях безпеки зони U2 з подальшим шифруванням. Пакет даних, що передається по одному каналу і перетинає "остання милю" 1318D, спочатку дешифрують, а потім проводять його дескремблювання шляхом проведення операції дескремблювання 928, здійснюваної

шлюзним медіа-вузлом $M_{b, f}$ з використанням налаштувань безпеки зони U2. Потім, використовуючи налаштування безпеки для зони Z1, шлюзний медіа-вузол $M_{b, f}$ здійснює розділення, скремблювання і шифрування даних для передачі по сітці в мережі 1318A з використанням налаштувань безпеки зони Z1. У шлюзовому медіа-вузлі $M_{b, d}$ відбувається дешифрування пакету даних, а також їх дескремблювання проведенням операції дескремблювання 928 з подальшим змішуванням з утворенням пакету даних для обміну інформацією по одному каналу з використанням налаштувань безпеки зони Z1. Далі шлюзний медіа-вузол $M_{b, d}$ здійснює повторне скремблювання і шифрування пакету даних, що передаються по одному каналу, використовуючи налаштування безпеки зони U1, і надалі направляє ці дані клієнтові $C_{1,1}$. Додаток SDNP, що працює на стільниковому телефоні 32, здійснює дешифрування і подальше дескремблювання, проводячи операцію дескремблювання 928, а остаточний пакет передається в місце призначення з використанням налаштувань безпеки зони U1.

Аналогічні явища відбуваються при передачі у зворотному напрямку, тобто при обміні інформацією між клієнтським вузлом $C_{1,1}$ і клієнтським вузлом $C_{2,1}$, а саме між стільниковим телефоном 32 і планшетом 33; додаток SDNP, що працює на клієнтському вузлі $C_{1,1}$, здійснює скремблювання пакету даних, що надсилаються, реалізуючи операцію скремблювання 926 з налаштуваннями безпеки зони U1 і подальшим шифруванням. Пакет даних, який передається по одному каналу і перетинає "останню милю" 1318E, спочатку дешифрують, а потім проводять його дескремблювання шляхом проведення операції дескремблювання 928, здійснюваної шлюзним медіа-вузлом $M_{b, d}$ з використанням налаштувань безпеки зони U1. Потім, використовуючи налаштування безпеки для зони Z1, шлюзний медіа-вузол $M_{b, d}$ проводить розподіл, скремблювання і шифрування даних для передачі по сітці в мережі 1318A з використанням налаштувань безпеки зони Z1. У шлюзовому медіа-вузлі $M_{b, f}$ відбувається дешифрування пакету даних, а також їх дескремблювання проведенням операції дескремблювання 928 з подальшим змішуванням з утворенням пакету даних для обміну інформацією по одному каналу з використанням налаштувань безпеки зони Z1. Далі шлюзний медіа-вузол $M_{b, f}$ здійснює скремблювання і шифрування пакету даних, що передаються по одному каналу, використовуючи налаштування безпеки зони U2, і надалі надсилає ці дані на клієнтський вузол $C_{2,1}$. Додаток SDNP, що працює на планшеті 33, здійснює дешифрування і подальше дескремблювання, проводячи операцію дескремблювання 928 з використанням налаштувань безпеки зони U2. Потім пакет даних передається клієнту, в даному випадку це планшет 33.

Як зазначалося вище, усі показані комунікаційні зв'язки забезпечують передачу зашифрованих даних незалежно від скремблювання та змішування, що ілюструється зображенням замка 1339. Докладного опису стадій шифрування та дешифрування для забезпечення наочності не наведено. В одному з варіантів реалізації здійснюється дешифрування та шифрування пакетів даних (тобто їхнє повторне шифрування) щоразу, коли дані перетинають новий медіа-вузол. Принаймні у всіх медіа-вузлах, що здійснюють повторне скремблювання, дані, які надходять перед дескремблюванням, дешифруються, після чого проводиться їх скремблювання та шифрування. Зведена інформація про можливу багаторівневу безпеку, якої можна досягти при транспортуванні по сітці, шифруванні та скремблюванні, коли у всіх випадках використовуються налаштування безпеки для тієї або іншої зони, наведена в таблиці нижче.

Метод забезпечення безпеки	Безпека в хмарі	Безпека в "останній милі"
Решітчаста маршрутизація в хмарі, Шифрування не проводиться, Скремблювання не проводиться	1-D	Відсутня
Решітчаста маршрутизація, Наскрізне шифрування, Скремблювання не проводиться	2-D	1-D
Решітчаста маршрутизація, Наскрізне скремблювання + Шифрування	3-D	2-D
Динамічна решітчаста маршрутизація, Наскрізне скремблювання + Шифрування	4-D	3-D
Динамічна решітчаста маршрутизація, Наскрізне скремблювання + Шифрування + "Сміттєві" дані	4-D	3.5-D

Як показано в наведеній вище таблиці, введення динамічних змін у шифрування та скремблювання при передачі даних згодом забезпечує додатковий рівень безпеки за рахунок

обмеження часу, упродовж якого кіберзлочин повинен перехопити пакет і "зламати код" для зчитування даних у пакеті. Динамічні зміни можуть відбуватися щодня, щогодини або за іншим розкладом чи від пакета до пакета, в останньому випадку зміни відбуваються приблизно кожні 100 мс. З наведеної вище таблиці також зрозуміло, що зв'язок "останньої милі" менш безпечний, ніж шлях передачі даних через хмару.

Одним із засобів підвищення безпеки з'єднань "останньої милі" є динамічне додавання фрагментів "сміттєвих" даних у потік даних і навіть відсилення пакетів, що складаються винятково зі "сміттєвих" даних, як принади, що змушує кіберзлочин витратити комп'ютерні ресурси на дешифрування марної інформації. Це поліпшення представлене у вигляді переходу від 3-D до 3.5-D, що означає, що додавання "сміттєвих" даних не є способом підвищення безпеки, надійним так само, як досягається при шифруванні, скремблюванні, а також передачі даних декількома маршрутами; водночас, цей спосіб все-таки забезпечує поліпшення, особливо якщо сміттєві дані, що вводяться, змінюються згодом і є відмінними у пакетах, які присилаються і надсилаються. Іншим важливим аспектом підвищення безпеки SDNP відповідно до цього винаходу є зазначення "неправильного напрямку", тобто маскуванню даного джерела та призначення при маршрутизації пакета; ця тема обговорюється далі в даному описі.

Передача секретів, ключів та початкових значень. Безпечний обмін інформацією на підставі SDNP ґрунтується на передачі між особами інформації для обміну, про яку сторонні особи не інформовані або значення чи мету якої вони не можуть зрозуміти. Окрім дійсного змісту переданих даних ця інформація може містити розподілені (спільні) секрети, алгоритми, ключі шифрування та дешифрування, а також числові початкові значення. "Спільний секрет" у значенні, використовуваному в цьому документі, являє собою інформацію, яку знають або якою діляться тільки певні особи. Це може бути, наприклад, перелік алгоритмів змішування, скремблювання та/або шифрування, ключ шифрування та/або дешифрування та/або генератор початкових даних, генератор чисел або інший спосіб вибору певних даних згодом. Наприклад, селектор 1307, зображений на рис. 92B, являє собою поділюваний секрет. Працюючи в поєднанні з розподіленими (спільними) секретами, числові початкові значення, які можуть ґрунтуватися на часі та/або стані, використовуються для вибору певних алгоритмів, для виклику різноманітних налаштувань та для виконання програм. Самі собою певні числові початкові значення не мають значення, проте у комбінації з поділюваним секретом, числові початкові значення можуть використовуватися для передачі динамічного повідомлення або події по мережі без розкриття його чи її значення або функції при перетині.

Аналогічно, для обміну зашифрованою інформацією шифрування потребує певного алгоритму, погодженого між особами, які нею обмінюються; це може бути, наприклад, поділюваний секрет, а також обмін одним або двома ключами, використовуваними для шифрування та дешифрування. У методах, що передбачають використання симетричного ключа, ключі шифрування та дешифрування однакові. Обмін симетричними ключами чутливий до атак, якщо ключ має більшу довжину, наприклад, 34 біта або 36 біт; тоді як, наявний проміжок часу для злому шифру короткий, наприклад, одна секунда або менше. Для будь-якого заданого алгоритму шифрування співвідношення між кількістю біт, використовуваних у симетричному ключі шифрування, і проміжком часу, упродовж якого ключ дійсний, являє собою міру стійкості шифрування. Симетричні ключі як такі можуть використовуватися в динамічній мережі за умови, що вони великі та що проміжок часу, упродовж якого потрібно зламати шифрування, короткий. Інший варіант полягає в тому, що алгоритми шифрування можна застосовувати у випадках, коли ключі шифрування та дешифрування різні або "асиметричні", і один ключ використовується для шифрування, а інший – для дешифрування. У відкритих каналах обміну інформацією асиметричні ключі кращі, оскільки здійснюється передача тільки ключа для шифрування, і цей ключ не дає інформації про ключ для дешифрування. Функціонування у взаємодії комбінації симетричних і асиметричних ключів кодування, числових початкових значень і спільних секретів при їхній динамічній зміні в часі забезпечує найвищу багатомірну безпеку обміну інформацією з SDNP. Існує безліч джерел інформації загального плану про шифрування інформації, наприклад, "Computer Security and Cryptography" by Alan G. Konheim (Wiley, 2007). Водночас адаптація шифрування до обміну інформацією в режимі реального часу не є безпосередньою, та інформацію про неї навряд чи можна знайти в загальнодоступних літературних джерелах. У багатьох випадках додавання шифру при обміні інформацією збільшує час очікування та затримку при передачі даних, знижуючи якість обслуговування, забезпечувану мережею.

Спільними секретами можна обмінюватися між клієнтськими вузлами та медіа-вузлами до початку дійсного комюніке, передачі, дзвінків і обміну даними. На рис. 100A показано, як розподілені (спільні) секрети можна передавати в сполученні з встановленням коду SDNP, що

виконується. У зоні Z1 безпечний пакет програмного забезпечення 1352A складається з виконуваного коду, 1351 і секретів 1350, якими поділилися, що перебувають у зоні Z1; сюди може входити генератор початкових станів 921, генератор чисел 960, алгоритми 1340, код шифрування 1022, код дешифрування 1030, а також їхні комбінації. Безпечний пакет програмного забезпечення 1352A для зони Z1, включаючи виконуваний код 1351 і розподілені (спільні) секрети 1350A, передається на медіа-сервери 1118 у хмарі 1114, а також на обидва "DMZ"-сервери, тобто 1353A і 1353B. Встановлення виконуваного коду 1351 у медіа-вузлах $M_{a, b}$ і $M_{a, f}$, а також інших вузлах на серверах 1118 виконується одночасно з встановленням спільних секретів для зони Z1, тобто секретів 1350A Z1 на окремих комп'ютерах, які в цьому документі називають DMZ-серверами 1353A і 1353B.

Термін "DMZ", який зазвичай являє собою скорочення, що означає "демілітаризована зона", у цьому випадку означає комп'ютер-сервер, до якого немає доступу безпосередньо через Інтернет.

DMZ-сервери можуть керувати одним або декількома серверами, які підключені до мережі і функціонують як медіа-вузли, але жоден медіа-сервер 1118 не має доступу до будь-якого з DMZ-серверів - DMZ-серверів 1353A, 1353B та інших (не показано). Розподіл усього програмного забезпечення та спільних секретів здійснюється в ході безпечного обміну інформацією, вони дійсні тільки упродовж короткого проміжку часу, позначеного у вигляді замка з годинниками 1354. Якщо доставка програмного забезпечення зроблена занадто пізно, то адміністратор SDNP повинен повторно надати повноваження на завантаження безпечного пакета програмного забезпечення 1352A для зони Z1 після персонального підтвердження ідентичності власника акаунту та його повноважень.

Уточнюємо, що опис DMZ-сервера як "комп'ютера-сервера, не підключеного до мережі Інтернет безпосередньо", означає, що не існує прямого електричного зв'язку між Інтернет і серверами. Тоді як файл 1352A в Z1 фактично можна завантажити на сервер або серверне господарство через Інтернет, встановлення файлу на DMZ потребує втручання адміністратора акаунту сервера або серверного господарства разом із власником акаунту. Перед встановленням файлів на DMZ адміністратор акаунту підтверджує ідентичність власника акаунту та правильність встановлення.

Після підтвердження встановлення адміністратор завантажує файл, що містить секрети Z1, на DMZ-сервер, користуючись локальною мережею (LAN), по якій комп'ютер адміністратора підключений безпосередньо до DMZ-сервера. Через це LAN не підключена до мережі Інтернет безпосередньо, а потребує авторизованого переходу через комп'ютер адміністратора після процесу строгої автентифікації. Встановлення спільних секретів здійснюється в одному напрямку, при цьому файли завантажуються на DMZ-сервери з неможливістю їхнього читання з доступом через Інтернет. Аналогічно, завантаження в Інтернет файлів з DMZ-сервера заборонено, що сприяє запобіганню доступу у режимі он-лайн і хакерству.

Процес встановлення спільних секретів аналогічний банківському рахунку, для якого можливість онлайн-банкінгу не забезпечена, а службовець банку може здійснити електронний банківський переказ у ручному режимі з дозволу клієнта. За рахунок відмови в доступі до Інтернет перехоплення спільних секретів вимагало б фізичного входу та місцевої атаки на серверне господарство, тобто в місці, де проведення LAN повинно бути ідентифіковане, підключене та втручання в яке повинно бути здійснене точно в момент передачі даних. Навіть у цьому випадку встановлюваний файл зашифрований і доступний тільки упродовж короткого проміжку часу.

Ту саму концепцію можна поширити на багатозонне встановлення програмного забезпечення, показане на рис. 100B, де сервер адміністрування SDNP 1355 використовується для надсилання безпечного пакета програмного забезпечення 1352A для зони Z1 на DMZ-сервер 1353A у вигляді секретів 1350A зони Z1, а також на медіа-сервери 1118 у хмарі 1114 у вигляді виконуваного коду 1351. Сервер адміністрування SDNP 1355 аналогічним способом використовується для розподілу безпечного пакета програмного забезпечення 1352B для зони Z2 на DMZ-сервер 1353B у вигляді секретів, що розподіляються, 1350B зони Z2, а також на медіа-сервери у хмарі 1315 у вигляді виконуваного коду 1351. Сервер адміністрування SDNP 1355 також направляє безпечний пакет програмного забезпечення 1352C, що включає виконуваний код 1351, до мостових медіа-вузлів $M_{b, f}$ у хмарі 1114 SDNP і $M_{b, n}$ у хмарі 1115, а розподілені (спільні) секрети 1350C – у зони Z1 і Z2, а також на DMZ-сервер 1353. Мости медіа-вузлів $M_{b, f}$ у хмарі SDNP 1114 і $M_{b, n}$ у хмарі SDNP 1115 отримують виконуваний код 1351 безпосередньо через сервер адміністрування 1355C, а розподілені (спільні) секрети зон Z1 і Z2 – з DMZ-сервера 1353C. Оскільки міст медіа-вузла $M_{b, f}$ виконує передачу між секретами Z1 і Z2, то тільки він (і жоден інший мостовий сервер не показаний) потребує доступу до спільних

секретів як Z1, так і Z2. В іншому випадку вузли в зоні Z1 потребують доступу тільки до спільних секретів зони Z1, а вузли в зоні Z2 потребують доступу тільки до спільних секретів зони Z2.

Важливо підкреслити, що тоді як сервер адміністрування SDNP 1355 направляє розподілені (спільні) секрети на DMZ-сервери 1353A, 1353B і 1353C, сервер адміністрування SDNP 1355 не має інформації про те, що буває з спільними секретами після доставки, він також не виконує команд і не керує спільними секретами після їхньої доставки. Наприклад, якщо переробляють список алгоритмів, тобто змінюють порядок їхнього розташування, при якому змінюється адреса певного алгоритму, сервер адміністрування SDNP 1355 не має інформації про те, яким чином здійснюється перестановка. Аналогічно, сервер адміністрування SDNP 1355 не є одержувачем числових початкових значень і обміну ключами між сторонами, що обмінюються інформацією, і через це не є пунктом керування. Фактично, як виявляється, жоден сервер у всій мережі SDNP не має повної інформації про пакет, маршрут його передачі, його налаштування безпеки, а також про його вміст. Таким чином, мережа SDNP однозначно є повністю розподіленою системою для безпечного загального обміну інформацією.

Доставка спільних секретів на DMZ-сервер, як показано на рис. 101A, здійснюється у вигляді чітко визначеного процесу, в результаті чого сервер адміністрування SDNP 1355 встановлює зв'язок з DMZ-сервером 1353A та виконує процес автентифікації для підтвердження того, що комп'ютер дійсно являє собою DMZ-сервер, авторизований SDNP. Процес може бути автоматичним або він може передбачати втручання та перевірки власників акаунтів способом, аналогічним банківському переказу. У кожному разі електронний сертифікат авторизації формується тільки після того, як автентифікація підтверджує автентичність DMZ-сервера 1353A, що дає можливість серверу адміністрування SDNP 1355 передавати свої секрети та код на DMZ-сервер 1353A. Після завантаження ці налаштування направляються на медіа-сервери 1361, 1362 і 1363, видаючи інструкції медіа-вузлам M₁, M₂ і M₃, відповідно, про те, як обробляти вхідні та вихідні пакети даних.

Той самий DMZ-сервер 1353A може керувати більш ніж одним медіа-сервером, тобто група медіа-серверів 1360 або низка DMZ-серверів можуть містити ті самі налаштування безпеки та розподілені (спільні) секрети. Усі медіа-вузли можуть функціонувати для одночасного перенесення медіа-файлів, вмісту та даних з використанням поділу часу та зрівноважування навантаження. Якщо навантаження на групу медіа-серверів 1360, пов'язане з обміном інформацією, падає, то медіа-вузол M₃ можна відключити від комп'ютера, що умовно показано у вигляді розімкнутих ключів 1365A та 1365B, при цьому залишаючи медіа-вузол M₂ функціонуючим, що умовно показано у вигляді замкнутих ключів 1364A та 1364B. Ключі не показують, що вхід і вихід певного сервера фізично відключені, а тільки те, що на сервері більше не працює додаток медіа-вузла, внаслідок чого відбувається економія енергії та відсутня необхідність плати за користування хостингом серверів, необхідності в чому немає. Відповідно до зображення, один DMZ-сервер 1353A може керувати роботою більше ніж одного медіа-сервера за рахунок завантаження інструкцій, команд і секретів з DMZ-сервера 1353A на будь-який сервер із групи серверів 1360, проте зворотна дія неможлива. Будь-яка спроба отримати інформацію, зробити запис, запит або перевірити вміст DMZ-сервера 1353A з медіа-сервера блокується міжмережним екраном 1366, а це означає, що вміст DMZ-сервера 1353A не можна перевірити або розкрити його з мережі Інтернет за допомогою медіа-вузла.

Приклад безпечного обміну інформацією відповідно до даного винаходу на підставі спільних секретів показано на рис. 101B, де перед обміном інформацією розподілені (спільні) секрети для зони Z1 були спрямовані сервером адміністрування (не показано) на усі DMZ-сервери в зоні Z1, у тому числі DMZ-сервери 1353A та 1353B. Такі розподілені (спільні) секрети можуть включати без обмежень генератор початкових даних 921, генератор випадкових чисел 960, алгоритми 1340, ключ шифрування 1022, а також ключ дешифрування 1030. При обміні інформацією між передавальним медіа-вузлом M_S і приймальним медіа-вузлом M_R, що перебувають на медіа-серверах 1118, DMZ-сервер 1353A передає розподілені (спільні) секрети на передавальний медіа-вузол M_S для формування пакета корисного навантаження 1341 і стану 920 з описом часу створення пакета корисного навантаження 1342. Перед передачею з медіа-вузла M_S пакет корисного навантаження 1342 також шифрується шляхом проведення операції шифрування 1339, представленої символічно у вигляді замка.

Після отримання безпечного пакета корисного навантаження 1342 медіа-вузол M_R, який приймає, проводить дешифрування пакета 1342, використовуючи ключ дешифрування 1030, що міститься в поділюваних секретах 1350, які відсилаються DMZ-сервером 1353B, після чого, використовуючи інформацію про стан 920 пакета даних 1342, відновлює дані 1341. При альтернативному способі реалізації числові початкові значення 929 можна надіслати заздалегідь, тобто перед передачею пакета корисного навантаження 1342 з медіа-вузла M_S,

який відсилає, на медіа-вузол M_R , який приймає, у вигляді числових початкових значень 929, що існують упродовж певного проміжку часу. Якщо ними не скористалися упродовж заданого проміжку часу або якщо відсилання пакета корисного навантаження 1342 затримане, то час існування початкових даних закінчується і вони самознищуються, приводячи медіа-вузол M_R у стан, коли він не може відкрити пакет корисного навантаження 1342.

Інший приклад безпечного обміну інформацією відповідно до даного винаходу, ґрунтується на поділюваних секретах у комбінації з початковими даними та ключем, включеним до складу пакета, що відсилається, як показано на рис. 101С. У цьому прикладі перед обміном інформацією розподілені (спільні) секрети 1350А для зони Z1 передаються на всі DMZ-сервери зони Z1, у тому числі сервери 1353А та 1353В. Такі розподілені (спільні) секрети можуть без обмежень включати генератор початкових даних 921, генератор випадкових чисел 960 і алгоритми 1340, але вони не містять ключів, зокрема, ключа шифрування 1022 і ключа дешифрування 1030. У ході обміну інформацією між медіа-вузлом M_S , який відсилає, і медіа-вузлом M_R , який приймає, що перебувають на медіа-серверах 1118, DMZ-сервер 1353А направляє секрети на медіа-вузол M_S , який відсилає, для формування пакета корисного навантаження 1342, що складається з даних 1341, стану 920 (із зазначенням часу формування пакета корисного навантаження 1342) і ключа шифрування (який використовують для шифрування пакетів корисного навантаження надалі). Перед маршрутизацією пакет корисного навантаження 1342 шифрується шляхом проведення операції шифрування 1339, представленої умовно у вигляді замка.

Після отримання безпечного пакета корисного навантаження 1342 медіа-вузол M_R , який приймає, проводить дешифрування пакета 1342, використовуючи ключ дешифрування 1030, що існує упродовж певного проміжку часу та надісланий заздалегідь, тобто до передачі корисного навантаження 1342 під час окремої передачі інформації між медіа-вузлом M_S , який відсилає, і медіа-вузлом M_R , який приймає. Цей пакет даних, відісланий раніше, можна убезпечити з використанням спільних секретів, наприклад, ще одного дешифрування, динамічного алгоритму, числових початкових значень або їхньої комбінації. Якщо кодом дешифрування 1030 не скористалися упродовж заданого проміжку часу або якщо відсилання пакета даних 1342 затримано, то термін дії коду дешифрування 1030 закінчується і він самознищується, приводячи медіа-вузол M_R у стан, коли він не може відкрити пакет корисного навантаження 1342. Хоча замість цього код дешифрування 1030 може бути включений у пакет корисного навантаження 1342, такий спосіб не вважається кращим.

Один із шляхів запобігання необхідності передачі всієї інформації, що стосується безпеки, із вмістом, полягає в розбивці та виділенні каналу, використовуваного для подачі командних сигналів і сигналів керування з каналу передачі медіа-файлів, використовуваного для передачі вмісту. Відповідно до даного винаходу, така "двоканальна" система обміну інформацією, показана на рис. 102, складається з медіа-каналу, підтримуваного медіа-серверами, і каналу передачі командних сигналів та сигналів керування, підтримуваного другою мережею комп'ютерів, що називаються у даному документі сигнальними серверами. У ході обміну інформацією сигнальний сервер 1365, на якому працює встановлене програмне забезпечення SDNP, функціонує як сигнальний вузол S_1 для передачі командних сигналів і сигналів керування, тоді як медіа-сервери 1361, 1362 і 1363, на яких працює встановлене програмне забезпечення SDNP, функціонують як медіа-вузли M_1 , M_2 і M_3 , відповідно, при передачі вмісту та медіа-файлів. При такій організації канал передачі медіа-файлів не передає командні сигнали та сигнали керування, та ці сигнали не обов'язково передавати по медіа-каналі у вигляді комбінації з корисним навантаженням або окремо у вигляді переданого заздалегідь пакета даних до передачі пакета даних, в якому знаходиться вміст повідомлення.

Під час роботи пакети передаються на сигнальний вузол S_1 з описом маршрутизації та налаштувань безпеки для пакетів медіа-файлів, надходження яких на групу серверів 1360 очікується. У цьому документі такі пакети спеціального призначення називаються "пакетами команд і керування". При обміні інформацією пакети команд і керування направляються на медіа-сервери 1361, 1362 і 1363, видаючи інструкції медіа-вузлам M_1 , M_2 і M_3 , відповідно, про те, яким чином обробляти пакети даних, які надходять та відсилаються. Ці інструкції комбінуються з інформацією, що зберігається на DMZ-сервері 1353А. Як описувалося вище, той самий DMZ-сервер може керувати роботою більше ніж одного медіа-сервера, наприклад, групою медіа-серверів 1360. Усі медіа-вузли можуть функціонувати для перенесення медіа-файлів, вмісту та даних з використанням поділу часу та зрівноважування навантаження. Якщо навантаження на групу медіа-серверів 1360, пов'язане з обміном інформацією, падає, то медіа-вузол M_3 можна відключити від комп'ютера, що умовно показано у вигляді розімкнутих ключів 1365А і 1365В, при цьому залишаючи медіа-вузол M_2 функціонуючим, що умовно показано у вигляді замкнутих

ключів 1365A та 1365B, при цьому залишаючи медіа-вузли M_1 і M_2 функціонуючими, що умовно показано у вигляді замкнутих ключів 1364A і 1364B. Ключі не показують, що вхід і вихід певного сервера фізично відключені, а тільки те, що на сервері більше не працює додаток медіа-вузла, внаслідок чого відбувається економія енергії та відсутня необхідність плати за користування хостингом серверів, необхідності в чому немає.

Відповідно до зображення, один DMZ-сервер 1353A може керувати роботою більше ніж одного медіа-сервера за рахунок завантаження інструкцій, команд і секретів з DMZ-сервера 1353A на будь-який сервер із групи серверів 1360, однак зворотна дія неможлива. Будь-яка спроба отримати інформацію, зробити запис, запит або перевірити вміст DMZ-сервера 1353A з сигнального сервера 1365 або медіа-серверів 1361, 1362 і 1363 блокується міжмережовим екраном 1366, а це означає, що вміст DMZ-сервера 1353A не можна перевірити або розкрити його з мережі Інтернет за допомогою медіа-вузла.

Таким чином, у системі двостороннього обміну інформацією подача команд і керування системою обміну даних передбачає використання іншого каналу передачі інформації, тобто унікальної маршрутизації, відмінної від передачі вмісту повідомлень. Мережа сигнальних серверів переносить усю інформацію, що стосується подачі команд і керування мережею, тоді як медіа-сервери містять дійсний вміст повідомлення. Пакети командних сигналів і сигналів керування можуть містити початкові значення, ключі, інструкції з маршрутизації, налаштування пріоритетності та ін., тоді як медіа-файли містять голосову, текстову інформацію, відеозаписи, електронні листи та ін.

Одна з переваг двоканального обміну інформацією полягає в тому, що пакети даних не містять інформації про джерела свого походження та місця призначення. Сигнальний сервер інформує всі медіа-сервери про те, що потрібно робити з кожним пакетом даних, який надходить, на підставі "необхідності знати", тобто як ідентифікувати пакет, який надходить, за адресою вузла, що направила його, або за "поштовим індексом" SDNP, що з ним робити та куди його відправити. За рахунок цього пакет ніколи не містить більше інформації про маршрутизацію, ніж та, яка стосується його останнього переходу і його наступного транзитного переходу в хмарі. Аналогічно, сигнальні сервери містять інформацію про командні сигнали та сигнали керування, але не мають доступу до вмісту пакета даних і обміну інформацією, переданою по каналу обміну медіа-файлами. Такий поділ керування без вмісту, а також вмісту без маршрутизації забезпечує вищий рівень безпеки, ніж у двоканальних мережах на основах SDNP.

Приклад безпечного двоканального обміну інформацією відповідно до даного винаходу показано на рис. 103A, де пакети даних про командні сигнали та сигнали керування, що складаються з початкових даних 929 і ключа дешифрування 1080, передаються сигнальними серверами 1365, тоді як медіа-файли та вміст передаються між медіа-серверами 1118. У цьому прикладі секрети 1350A зони Z1 перед передачею інформації відсилаються на всі DMZ-сервери зони Z1, у тому числі сервери 1353A і 1353B, де такі розподілені (спільні) секрети можуть містити без обмежень генератор початкових даних 921, генератор випадкових чисел 960 і алгоритми 1340, але не містять ключів, зокрема, ключа дешифрування 1030. Перед початком обміну інформацією сигнальний вузол S_s , що знаходиться на сигнальному сервері 1365, надсилає пакет командних сигналів і сигналів керування, що складається з числових початкових значень 929 і ключа дешифрування 1030, або інші налаштування безпеки на сигнальний вузол призначення S_d . Ця інформація в комбінації з спільними секретами та налаштуваннями безпеки, що зберігаються на DMZ-серверах 1353A та 1353B, надалі використовується для подачі інструкцій про те, яким чином медіа-вузол M_s , який направляє, повинен передавати шифроване корисне навантаження 1342 на медіа-вузол M_R , який приймає. Шифрування інформації з корисним навантаженням 1342 показано у вигляді замка 1339.

За рахунок цього крім переданих даних 1341 єдиними даними, пов'язаними з безпекою, що містяться в пакеті корисного навантаження 1342, є стан 920, що описує час формування пакета 1342. Після надходження пакета корисної інформації 1342 на медіа-вузол M_R , який приймає, відбувається її дешифрування з використанням ключа дешифрування 1030. Після дешифрування початкові значення 929 у комбінації з інформацією про стан 920 і спільними секретами 1350A, що надходять із DMZ-сервера 1353B, використовуються для дескремблювання, змішування та розділення пакета корисного навантаження 1342 та інших пакетів даних, що надходять, відповідно до описаного вище методу. Хоча пакет даних може нести інформацію про час своєї останньої зміни (інформація про стан особливо корисна для локального генерування ключів дешифрування), одночасне використання початкових даних, переданих по каналу передачі командних сигналів і сигналів керування, забезпечує можливість ідентифікації операцій розділення та дескремблювання, яким раніше піддавали пакет даних,

який надходить, тільки за часом, це не обов'язково операції, що проводилися на вузлі, що безпосередньо передуює даному.

В іншому способі реалізації, показаному на рис. 103B, числові початкові значення 929 доставляються заздалегідь, тобто раніше, ніж пакет із корисним навантаженням 1342, по каналу передачі медіа-файлів, але ключ дешифрування 1030 як і раніше передається каналом передачі сигналів. Комбінація або перестановки методів доставки можливі для забезпечення безпечного обміну інформацією. Альтернативний варіант полягає в тому, що початкові значення, ключі та інші динамічні налаштування безпеки згодом можуть змінюватися.

Для полегшення описаного вище наскрізного забезпечення безпеки у клієнта необхідно встановлювати також виконуваний код, розподілені (спільні) секрети та ключі, як правило, їх завантажують у вигляді додатка. Для запобігання розкриттю налаштувань безпеки, використовуваних у мережі SDNP, ці завантажені файли встановлюють в окремій зоні, відомій тільки клієнтові та хмарному шлюзовому вузлу, за допомогою якого вони передаються. Як показано на рис. 104, для забезпечення мобільному пристрою, наприклад, стільниковому телефону 32, можливості обмінюватися інформацією з використанням хмари SDNP, він повинен спочатку стати авторизованим клієнтом SDNP. Цей етап включає завантаження пакета програмного забезпечення 1352D зони U1 із сервера адміністрування SDNP 1355 на клієнтський вузол $C_{1,1}$, тобто стільниковий телефон 32, з використанням безпечної лінії завантаження 1354, що існує лише упродовж обмеженого проміжку часу. Якщо для завершення завантаження потрібно занадто багато часу або воно не відповідає певним критеріям автентифікації, які підтверджують, що користувач є реально існуючим пристроєм і не є комп'ютером хакера, який робить вигляд, що є клієнтом, то файл за жодних умов не дешифрується та не встановлюється на стільниковий телефон 32. У пакеті програмного забезпечення 1352D у зоні U1 міститься виконуваний код 1351, специфічний для операційної системи стільникового телефону 32 або іншого пристрою, на який встановлюється код, наприклад, iOS, Android, Windows, MacOS та ін., а також секрети 1350D зони U1, до яких можуть належати комбінація генератора початкових значень 921, генератор випадкових чисел 960, алгоритми 1340, код шифрування 1022 і код дешифрування 1030, усі вони специфічні для клієнтської зони U1.

Для будь-якої зони U1 зовнішній клієнтський вузол $C_{1,1}$ для обміну інформацією з зоною Z1 хмари 1114 SDNP, шлюзні вузли, такі як медіа-вузол $M_{a,d}$, повинні одержати інформацію, що стосується налаштувань безпеки як зони Z1, так і зони U1, що містяться в пакеті завантаження 1352E зон U1 і Z1. З використанням обмежених за часом і безпечних методів завантаження, позначених у вигляді замка 1354, секрети як зони Z1, так і зони U1 завантажуються по лінії зв'язку 1350C на DMZ-сервер 1353C, а виконуваний код 1351 завантажується по лінії зв'язку 1351 і встановлюється на медіа-вузол SDNP $M_{a,d}$, а також на інші медіа-вузли зони Z1, необхідні для створення шлюзних з'єднань між хмарою 1114 і зовнішніми клієнтами, наприклад, з'єднань, що підтримують можливість з'єднання "останньої милі". Після того, як медіа-вузол $M_{a,d}$ у зоні Z1 і клієнтський вузол $C_{1,1}$ у зоні U1 завантажені вмістом пакетів, завантаження 1352E і 1352D, відповідно, може бути забезпечений безпечний обмін інформацією 1306, що включає операцію шифрування 1339.

Оскільки обмін інформацією між безпечною хмарою в зоні Z1, що знаходиться на медіа-серверах 1118, і клієнтським вузлом $C_{1,1}$, що знаходиться на зовнішньому пристрої, наприклад, на мобільному телефоні 32 у зоні U1, може бути реалізований по одному каналу передачі інформації, необхідні деякі засоби для перетворення двоканального обміну інформацією, реалізованого в хмарі 1114, в одноканальний обмін інформацією, необхідний у "останній милі". Приклад функції, виконуваної шлюзним вузлом SDNP при реалізації перетворення двоканального обміну інформацією в одноканальний, показано на рис. 105A, де пакети командних сигналів і сигналів керування зони Z1, що надходять на сигнальний вузол S_d на сигнальному сервері 1365, комбінуються з вмістом медіа-файлів у шлюзовому медіа-вузлі M_R , забезпечуючи одноканальний обмін інформацією з пакетом корисного навантаження 1342, що містить дані 1341 і налаштування безпеки зони U2, включаючи стан 920, що показує час формування пакета 1342, числові початкові значення 929 і код шифрування 1022, які призначені для використання в шифруванні наступного пакета, тобто пакета, що повинен формувати вузол $C_{1,1}$.

Пакет корисного навантаження 1342 шифрується проведенням операції шифрування 1339. Для дешифрування пакета корисного навантаження 1342 необхідно користуватися ключем дешифрування 1030; цей ключ складається з одного з декількох спільних секретів 1350D зони U1, завантажених раніше в безпечний додаток і сховище даних 1359 разом з іншими секретами зони U1, такими як генератор початкових даних 921, генератор випадкових чисел 960 і алгоритми 1340. Замість цього, як показано на рис. 105B, можна спочатку доставити попередні

початкові значення 929, які використовуються для дескремблювання скрембльованого ключа дешифрування 1030, що у свою чергу використовують для дешифрування корисного навантаження 1342. Далі можна скористатися станом 920 для дешифрування або дескремблювання даних 1341, створюючи численні перешкоди для боротьби з порушеннями безпеки в каналах зв'язку "останньої милі".

Для запобігання розпізнаванню шаблону алгоритмів, повторно використовуваних клієнтом, адреса або код, використовуваний для вибору алгоритму зі списку алгоритмів, встановлених на пристрої-клієнті, відповідно до даного винаходу, регулярно змінюють, наприклад, щотижня, щодня, щогодини і т.д. Ця особливість, яка називається "перестановкою", проводиться способом, аналогічним перетасуванню колоди карт, і схожа на перестановку, виконувану в мережі. Перестановка забезпечує зміну місця розташування чисел, використовуваних для ідентифікації будь-якого заданого алгоритму в таблиці алгоритмів, незалежно від того, складається ця таблиця алгоритмів з методів скремблювання, змішування або шифрування. Як показано на рис. 106, для перестановки в будь-якій таблиці алгоритмів у клієнтському вузлі $C_{1,1}$, наприклад, встановлений на стільниковий телефон 32, з одночасним забезпеченням здатності хмари SDNP інтерпретувати нові адреси алгоритмів, сигнальний сервер 1365, на якому знаходиться сигнальний вузол S_s , направляє числові початкові значення 929 на клієнтський вузол $C_{1,1}$, що у свою чергу перенаправляє початкові значення в зону U1 генератора випадкових чисел 960. Результуюче число використовується для запуску алгоритму перестановки 1312, перетворюючи алгоритмічну таблицю 1368A зони U1 у нову алгоритмічну таблицю 1368F зони U1 і зберігаючи видозмінену таблицю в безпечних додатках і реєстрі даних 1359, який знаходиться на клієнтському вузлі $C_{1,1}$. Сигнальний сервер (не показано) формує числові початкові значення 929, виходячи з інформації про стан, отриманої із планованого часу 1310 і дати події 1311, використовуваних при складанні графіків процесу перестановки. Ту саму інформацію про стан і дату використовують для перестановок у таблиці на DMZ-сервері 1353A, забезпечуючи ідентичність і синхронізацію алгоритмічних таблиць хмари з алгоритмічними таблицями клієнта.

Удосконалений метод передачі налаштувань безпеки з хмари в клієнтський вузол припускає реалізацію двоканального обміну інформацією, як показано на рис. 107, де медіа-вузол M_R , що знаходиться на медіа-сервері 1118, направляє числові початкові значення 929 на клієнтський вузол $C_{1,1}$, а сигнальний вузол S_d , який знаходиться на окремому сигнальному сервері 1365, направляє ключ дешифрування 1030 на клієнтський вузол $C_{1,1}$. Перевага цього методу полягає в тому, що ключ дешифрування 1030 надходить від іншого джерела з адресою пакета SDNP, відмінною від адреси числових початкових значень 929 і пакета корисного навантаження 1342. Можливим недоліком є те, що, незважаючи на наявність різних шляхів обміну інформацією, у багатьох випадках імовірно те, що обидва мережеві канали будуть підтримуватися тим самим засобом, наприклад, одиничним з'єднанням Wi-Fi або LTE зі стільниковим телефоном 32. Скремблювання або шифрування коду дешифрування 1030 перед його передачею з сигнального сервера 1365 на клієнтський вузол $C_{1,1}$ може значною мірою нівелювати цей недолік, щоб його можна було перехопити або прочитати, здійснюючи прослуховування пакета.

При роботі числові початкові значення 929, передані каналом передачі медіа-файлів з медіа-вузла M_R на клієнтський вузол $C_{1,1}$, використовуються для вибору алгоритму дешифрування з алгоритмічної таблиці 1340 і розблокування безпеки на ключі дешифрування 1030, зображеному у вигляді замка 1339C. Після розблокування ключ дешифрування використовують для розблокування шифрування, піддаючи пакет корисного навантаження 1342 проведенню операції шифрування 1339B. Потім числові початкові значення 929 у сукупності з секретами 1350D зони U1 використовуються для відновлення даних 1341 для використання клієнтським вузлом $C_{1,1}$.

Якщо реалізують асиметричний обмін ключами, як показано на рис. 108, то DMZ-сервер 1353A створює пару асиметричних ключів, до якої входять секретний ключ дешифрування 1030A і відкритий ключ шифрування 1370A. Ключ дешифрування 1030A залишається секретним на DMZ-сервері у вигляді секрету зони Z1, а відкритий ключ шифрування 1370A передається через сигнальний вузол S_d на сервер обміну ключами 1369. На сервері обміну ключами 1369 ключ шифрування 1370A зберігається доти, поки він потрібний, після чого цей сервер передає його на клієнтський пристрій 1335. Коли клієнтський вузол $C_{1,1}$ формує пакет даних з корисним навантаженням 1342 для його відправлення на медіа-вузол M_R , він спочатку завантажує ключ шифрування 1370A зони Z1 із сервера обміну ключами 1369. Тоді як сигнальний сервер може пропустити ключ шифрування на клієнтський вузол $C_{1,1}$ безпосередньо, використання сервера обміну ключами 1369 забезпечує множинні переваги. Перша вигода від використання загальнодоступного сервера обміну ключами полягає в тому, що забезпечується маскування

при прямому розгляді, тобто "безпека в числах". Оскільки загальнодоступний сервер обміну ключами може генерувати мільйони ключів шифрування, у людини, яка здійснює втручання, не залишається способу довідатися, який саме ключ потрібно запитувати для отримання незаконного доступу до неавторизованого обміну інформацією. Навіть якщо вона якимсь дивом зможе вибрати потрібний ключ, ключ шифрування дає їй можливість лише шифрування повідомлень, а не їхнього дешифрування. По-третє, розподіл загальнодоступних ключів звільняє сигнальний сервер від обов'язку розподіляти ключі та підтверджувати доставку. Нарешті, при використанні загальнодоступного сервера обміну ключами у кіберпірата не залишається способу відстежити, звідки надійшов ключ шифрування, ускладнюючи виявлення пристрою, що викликає, через їхній сигнальний сервер.

Після отримання ключа шифрування 1370A вузол $C_{1,1}$ на клієнтському пристрої 1335 шифрує пакет корисного навантаження 1342, використовуючи обраний алгоритм шифрування, та ключ шифрування 1371Y. Оскільки медіа-вузол M_R має доступ до ключа дешифрування 1030 з DMZ-сервера 1353A, він може відкрити пакет корисного навантаження 1342 і прочитати файл. Секрети 1350D зони U1, навпаки, містять ключ дешифрування, що відповідає ключу шифрування (не показано), які надійшли з клієнтського вузла $C_{1,1}$ на сервер обміну ключами 1369. Коли медіа-вузол M_R формує пакет даних для клієнтського вузла $C_{1,1}$, він завантажує ключ шифрування 1370A зони U1 і потім проводить шифрування пакета корисного навантаження 1342 для його відсилання на клієнтський вузол $C_{1,1}$. З огляду на те, що стільниковий телефон 32 має доступ до секретів зони U1, у тому числі до ключа дешифрування 1030 зони U1, він може проводити дешифрування та читати пакет корисного навантаження 1342.

Відповідно до даного винаходу, користуючись описаними вище методами і їхніми комбінаціями, можна реалізувати безпечний обмін інформацією, включаючи доставку програмного забезпечення, спільних секретів, алгоритмів, генераторів випадкових чисел, числових початкових значень, а також асиметричних або симетричних ключів шифрування.

Передача пакета SDNP. Іншим аспектом способу безпечного обміну інформацією відповідно до даного винаходу є неможливість особи, що здійснює кібератаку, визначити, звідки надійшов пакет даних або пакет командних сигналів і сигналів керування, а також їхнє місце призначення, тобто істинне джерело та місце призначення сховані, при цьому відкривається тільки джерело та місце призначення окремого переходу. До того ж, у межах однієї хмари SDNP використовувані SDNP-адреси не є реальними IP-адресами, що існують у мережі Інтернет, це лише локальні адреси, що мають значення в окремій хмарі SDNP, аналогічно NAT-адресам. На противагу передачі даних у мережі NAT, при маршрутизації даних у мережі SDNP адреси SDNP у субпакеті даних переписуються після кожного переходу з вузла у вузол. Більш того, медіа-вузол не має жодної інформації про маршрутизацію пакета даних крім того, з якого останнього медіа-вузла він прийшов і на який медіа-вузол він піде далі. Протоколи відрізняються між собою так, як описано вище в прикладах одноканального та двоканального обміну інформацією, однак концепції маршрутизації однакові.

Одноканальна передача. Один із прикладів одноканального обміну інформацією представлено на рис. 109, де пакети даних передаються через решітчасту мережу SDNP, до якої підключені планшет 33 і стільниковий телефон 32, на кожному з яких працює додаток 1335, сумісний з SDNP. При безпечному обміні даними між клієнтським вузлом $C_{2,1}$ і клієнтським вузлом $C_{1,1}$ дані переміщуються по одноканальній лінії зв'язку "останньої милі", маршрутизація якої проведена в зоні U2 від клієнтського вузла $C_{2,1}$ до медіа-вузла $M_{a, f}$, після чого проводиться маршрутизація по решітчастій мережі в зоні Z1 хмари SDNP від шлюзового медіа-вузла $M_{a, f}$ до шлюзового медіа-вузла $M_{a, d}$; все це завершується в одноканальній лінії зв'язку "останньої милі" в зоні U1 від медіа-вузла $M_{a, d}$ до клієнтського вузла $C_{1,1}$. Пакет даних 1374B показує IP-адресацію, при якій пакет направляють із IP-адреси Addr TB на IP-адресу Addr MF, IP-адреса медіа-сервера 1220F.

Адреси ліній зв'язку "останньої милі" є дійсними IP-адресами. Після входу в зону Z1 хмари IP-адреса джерела в пакеті SDNP 1374F змінюється на псевдо-IP-адресу SDNP Addr MF, тобто адресу типу "NAT", що не має значення в мережі Інтернет. Якщо уявити спрощено, що маршрутизація в мережі являє собою єдиний перехід, то адреса місця призначення також є псевдо-IP-адресою, у цьому випадку це SDNP Addr MD. У лінії зв'язку "останньої милі" в зоні U1 адреса, зазначена в пакеті 1374G SDNP, перетвориться в дійсні IP-адреси, при цьому адресою джерела є IP Addr MD, а адресою місця призначення – IP Addr CP. При передачі пакетів у режимі реального часу всі пакети медіа-файлів використовують протокол UDP, а не TCP. Як відзначалося вище, корисне навантаження в різних зонах неоднакове: у лінії зв'язку "останньої милі" в зоні U2 корисне навантаження пакета медіа-файлів 1374B SDNP складається з пакета U2 SDNP, у решітчастій мережі та хмарі SDNP зони Z1 корисне навантаження пакета медіа-

файлів 1374F SDNP являє собою пакет Z1 SDNP, а в лінії зв'язку "останньої милі" в зоні U1 пакета медіа-файлів 1374G – пакет U1 SDNP. Тому, на відміну від обміну інформацією в мережі Інтернет, пакет медіа-файлів SDNP являє собою мінливе корисне навантаження, адреса, формат і зміст якого змінюються, і яке переміщається по мережі обміну інформацією.

На рис. 110A-110F показано ряд блок-схем, що ілюструють, яким чином здійснюється одноканальний обмін інформацією SDNP. При одноканальному обміні інформацією через порт (ad hoc) особи, що обмінюються інформацією, здійснюють обмін нею одним каналом, тобто каналом передачі медіа-файлів, з послідовністю, що забезпечує виконання сесії, а потім передачу даних або голосового повідомлення. Як показано на етапі 1380A рис. 110A, клієнт запускає додаток 1335, сумісний з SDNP, і починає діалог з будь-яким медіа-сервером SDNP, встановленим за замовчуванням, із числа медіа-серверів SDNP, перерахованих у таблиці 1375. Кожен із серверів SDNP, встановлених за замовчуванням, у цьому випадку – це медіа-сервер 1120S, на якому знаходиться медіа-вузол $M_{a, s}$, використовується як перший номер для контакту у всіх випадках, коли авторизований клієнт бажає здійснити виклик або створити сесію, користуючись мережею SDNP. При одноканальному обміні інформацією сервер 1220S виконує дві функції, функціонуючи як сервер за замовчуванням для першого контакту з новими ініціаторами виклику та одночасно виконуючи функцію медіа-сервера для передачі вже зроблених викликів. При альтернативному варіанті реалізації використовується окремий призначений "сервер імен" для функціонування як пристрій для першого контакту не тоді, коли ініціюється виклик, а завжди при першому підключенні пристроїв, тобто при їхній реєстрації в мережі. Використання сервера імен відповідно до даного винаходу описано далі в даній заявці.

Клієнтський додаток 1335, сумісний з SDNP, може бути безпечним додатком, робота якого дозволена SDNP, наприклад, персональною системою обміну особистими повідомленнями або безпечною електронною поштою, що функціонує на стільниковому телефоні, планшеті або ноутбуці. Замість цього клієнт може бути безпечним апаратним пристроєм, на якому працює вбудоване програмне забезпечення SDNP. До складу пристроїв з убудованим SDNP може входити автомобільний термінал віддаленого управління, касовий термінал для операцій із кредитними картками, спеціально призначений клієнт Інтернету речей, дозвіл на роботу якого наданий SDNP, або маршрутизатор SDNP. Маршрутизатор SDNP, описаний у даному документі, є периферичним апаратним пристроєм загального призначення, використовуваним для підключення будь-якого пристрою, на якому програмне забезпечення SDNP не працює, до безпечної хмари SDNP, приміром, до будь-якого ноутбука, планшета, електронного пристрою зчитування, стільникового телефону, ігрової приставки, а також пристрою з можливістю підключення до локальної мережі, Wi-Fi або Bluetooth.

Після контакту клієнтського додатка 1335 з одним із серверів SDNP, встановлених за замовчуванням, він надалі перенаправляється до шлюзового вузла SDNP. Шлюзний вузол можна вибирати, виходячи з його фізичної наближеності місця розташування клієнта до сервера, найменшого Інтернет-трафіку або за ознакою приналежності до ліній з найменшою затримкою поширення та мінімальною затримкою доставки даних. На стадії 1380B встановлений за замовчуванням сервер 1220S SDNP перенаправляє запит на з'єднання від клієнта на шлюзний медіа-сервер 1220F, що є найкращим варіантом вибору з SDNP, на якому знаходиться шлюзний медіа-вузол $M_{a, f}$ з SDNP. Далі шлюзний медіа-вузол $M_{a, f}$ проводить автентифікацію сертифіката 1357, що належить обом особам, підтверджує користувача, встановлює, чи є виклик безкоштовним або містить ознаки необхідності внесення плати та, за необхідності, підтверджує стан платоспроможності акаунту, після чого запускає серію SDNP.

На етапі 1380C клієнтський додаток 1335 відсилає початковий пакет 1374A SDNP із запитом адреси та інформації про маршрутизацію виклику до місця призначення, тобто особі або пристрою, якій був адресований виклик, з використанням запиту на маршрутизацію 1371, що надсилається до шлюзового медіа-серверу 1220F. З огляду на те, що пакет 1374A SDNP, який містить запит на маршрутизацію, є пакетом командних сигналів і сигналів керування, а не обміном даними в режимі реального часу (тобто пакетом даних), його доставка здійснюється з використанням протоколу TCP, а не UDP. Запит на маршрутизацію 1371 може встановлювати необхідність доставки інформації клієнтському додатку 1335 у будь-якому числі форматів, включаючи телефонний номер, адресу SDNP, IP-адреси, URL, а також спеціальний код SDNP, наприклад, "поштова адреса" SDNP пристрою призначення (у даному випадку це стільниковий телефон 32). Тому запит на маршрутизацію являє собою запит на інформацію про особу, якій адресований виклик, тобто будь-яку інформацію, необхідну для визначення місця призначення виклику, що включає в себе, наприклад, "поштову адресу" SDNP, їх IP-адресу або адресу SDNP.

На етапі 1380D рис. 110B шлюзний медіа-вузол SDNP $M_{a, f}$ здійснює пошук хмари SDNP 1114 і отримує адресу місця призначення, а це означає, що медіа-вузол $M_{a, f}$ ідентифікує особу, якій адресований виклик, і отримує необхідну інформацію для адресації виклику, що може являти собою, наприклад, "поштову адресу", IP-адресу або адресу SDNP особи, якій адресований виклик; далі на етапі 1380E шлюзний медіа-вузол $M_{a, f}$ передає інформацію про маршрутизацію, шлях, яким буде передано виклик, а також ключі шифрування, необхідні для переходу по певній зоні до клієнтського додатка 1335. Коли клієнт (планшет 33) одержує адресу місця призначення, на етапі 1380F (планшет 33) ініціює виклик з пакетом даних SDNP 1374B. Звукові хвилі, що виникають внаслідок голосової передачі даних, 1383A, перетворюються у формат цифрової інформації за допомогою аудіокодеку (не показано) і передаються в додаток 1335. Комбінування аудіо-інформації з маршрутизацією адреси та іншою інформацією, що накопичується в субпакеті SDNP, додаток 1335 формує пакет SDNP 1374B для маршрутизації по лінії зв'язку "останньої милі" від "IP Addr TB" до "IP Addr MF", і починає передачу пакета на медіа-вузол $M_{a, f}$. субпакет SDNP, вбудований у корисне навантаження 1372 пакета даних 1374B, може містити інформацію про терміновість, переваги щодо доставки, протоколи безпеки, а також специфікації типу даних. Оскільки маршрутизація пакета даних SDNP 1374B по лінії зв'язку "останньої милі" здійснюється з використанням IP-адреси, передача пакета здійснюється аналогічно звичайному Інтернет-трафіку, за винятком того, що дійсний вміст даних скрембльований та зашифрований з використанням налаштувань безпеки SDNP зони U2, а субпакет SDNP, який міститься в корисному навантаженні U2 SDNP 1372, що містить дані, також має специфічний формат, що відповідає безпечному динамічному протоколу мережі для зони U2. Безпечний динамічний протокол мережі для зони U2 являє собою сукупність спільних секретів, специфічно застосовних для обміну інформацією з рухом даною конкретною зоною, наприклад, початкові значення зони U2, розраховані з використанням специфічного генератора початкових даних зони U2, тобто методу генерування вихідних даних з використанням алгоритму, описаного раніше в прикладі, представленому на рис. 51A, але з використанням налаштувань безпеки, таблиць та ін., специфічних для зони U2. Аналогічно, алгоритми шифрування та скремблювання зони U2 ґрунтуються на налаштуваннях безпеки, специфічних для зони U2. Пакети, що пересилаються планшетом 33, скремблюються та зашифровуються описаним вище способом, виходячи зі стану (часу) і того, що ці пакети містять ключі дешифрування та початкові значення, які ідентифікують стан (час), коли вони були сформовані, забезпечуючи можливість дескремблювання та дешифрування пакетів медіа-вузлом $M_{a, f}$ з використанням налаштувань безпеки, заданих для зони U2.

Таким чином, кожен вузол ідентифікує кожен пакет, одержуваний ним, за його ярликом. Після ідентифікації пакета вузлом він виконує з пакетом необхідні операції дешифрування, дескремблювання, змішування, скремблювання, шифрування та розділення відповідно до інструкцій сигнального сервера по роботі з ним і в заданому порядку. Алгоритми або інші методи, використововувані в цих операціях, можуть ґрунтуватися на стані, наприклад, часі формування пакета, або початкових значеннях, які генеруються відповідно до алгоритму, певним станом. При виконанні кожної операції вузол може використовувати стан або початкові значення для вибору певного алгоритму або методу з таблиці, що зберігається в його пам'яті. Знову ж таки, відповідно до інструкцій, виданих сигнальним сервером, вузол присвоює кожному пакету ярлик, після чого забезпечує маршрутизацію пакета при його передачі на наступний вузол у ході його руху мережею SDNP. Звісно, мається на увазі, що якщо вхідні пакети були змішані та/або розділені, то пакети, що пересилаються вузлом, зазвичай не є тими самими пакетами, які він отримує, оскільки деякі сегменти даних могли бути переміщені в інші пакети, а сегменти даних з інших пакетів могли бути додані. Таким чином, після розділення пакета будь-який одержуваний у результаті пакет здобуває власний ярлик і переміщується власним маршрутом, не маючи жодної інформації про те, яким чином його "дочірні вузли" будуть проводити ті самі процеси, рухаючись до того самого кінцевого місця призначення. Вузол не має інформації про те, яким чином рухається кожен з пакетів, за винятком його найближчого переходу.

В одноканальних системах SDNP шлюзний та інші медіа-вузли повинні виконувати потрібне призначення, імітуючи завдання, виконувані сервером імен і сигнальним сервером. Фактично одноканальні, двоканальні та триканальні системи відрізняються цими трьома функціями: передача пакета, передача сигналу та "ім'я" виконуються на тих самих серверах в одноканальній системі, на двох типах серверів у двоканальній системі та трьох типах серверів у триканальній системі. Самі по собі функції в системах всіх трьох типів ідентичні.

У розподілених системах сервери, що виконують сигнальну функцію, мають інформацію про кінцеве місце призначення пакетів, однак жоден одиничний сервер не має інформації про увесь

маршрут руху пакетів. Наприклад, вихідний сигнальний сервер може мати інформацію про частину маршруту, проте коли пакети досягають певного медіа-вузла, функція сигналізації передається іншому сигнальному серверу, що приймає на себе виконання завдання щодо визначення маршруту від названої точки та далі.

5 Якщо провести грубу аналогію, то якщо пакет повинен відсилатися зі стільникового телефону, що знаходиться в Нью-Йорку, на переносний комп'ютер, що знаходиться в Сан-Франциско, то перший сигнальний сервер (або перший сервер, що виконує сигнальну функцію) може маршрутизувати пакет зі стільникового телефону на локальний сервер у Нью-Йорку (вхідний шлюзний вузол), а звідти – на сервери у Філадельфії, Клівленді, Індіанapolisі та
10 Чикаго, другий сигнальний сервер може забезпечувати маршрутизацію пакета з сервера в Чикаго на сервери в Канзасі та Денвері, третій сигнальний сервер може забезпечувати маршрутизацію пакета з сервера в Денвері на сервери в Солт-Лейк Сіті, Рено та Сан-Франциско (кінцевий шлюзний вузол) і, нарешті, на переносний комп'ютер, при цьому кожен сигнальний сервер визначає частину маршруту, за яку він відповідає, виходячи із затримок відправлення та
15 інших поточних умов трафіка в мережі SDNP. Перший сигнальний сервер може видавати інструкції другому сигнальному серверу про передбачуване переправлення пакета на сервер у Чикаго, а другий сигнальний сервер може видавати інструкції третьому сигнальному серверу про передбачуване переправлення пакета на сервер у Денвері, однак жоден із сигнальних серверів (або жоден із серверів, що виконують сигнальну функцію) не може мати повної
20 інформації про маршрут пакета.

Зазвичай, як відзначалося вище, пакет по маршруту проходження може змішуватися та розділятися на частини. Наприклад, замість звичайної маршрутизації пакета з сервера у Філадельфії на сигнальний сервер у Клівленді, сигнальний сервер може видати інструкції серверу в Клівленді про розділення пакета на три частини і їхню маршрутизацію на сервери в
25 Цинциннаті, Детройті та Клівленді, відповідно. Сигнальний сервер надалі також надасть інструкції серверу у Філадельфії про присвоєння кожному з цих трьох пакетів певного ярлика, та він може проінформувати сервери в Цинциннаті, Детройті та Клівленді про ярлики, для того щоб вони могли розпізнати ці пакети.

Етап 1380G на рис. 110C ілюструє маршрутизацію пакета даних 1374C SDNP від шлюзного медіа-вузла $M_{a, f}$, що знаходиться на медіа-сервері 1220F, до медіа-вузла SDNP $M_{a, j}$, що знаходиться на медіа-сервері 1220J. При одноканальному обміні інформацією маршрутизація даних визначається вперше в момент, коли шлюз уперше одержав адресу викликуваного клієнта на етапі 1380D. На відміну від маршрутизації пакета IP-даних 1374B по лінії зв'язку "останньої милі", перший перехід пакета SDNP 1374C всередині хмари відбувається з
35 використанням адрес SDNP "SDNP Addr MF" і "SDNP Addr MJ", які не можна розпізнати в мережі Інтернет. При одноканальному обміні інформацією маршрутизація даних, тобто послідовність вузлів, через кожен з яких пакет проходить шляхом проходження до місця призначення, визначається в момент, коли шлюзний вузол (у цьому випадку це вузол $M_{a, f}$) уперше одержує адресу викликуваного клієнта (у цьому випадку на етапі 1380D).

40 Корисне навантаження 1373A пакета даних SDNP 1374C піддається скремблюванню та шифруванню з використанням налаштувань безпеки SDNP зони Z1, а субпакет SDNP, що міститься в пакеті даних 1374C, в якому містяться дані з корисного навантаження 1373A, також форматується специфічно відповідно до безпечного динамічного мережевого протоколу для зони Z1. Захищений динамічний мережевий протокол для будь-якої зони являє собою
45 сукупність спільних секретів, специфічно застосовний для обміну інформацією з передачею даних через певну зону; у цьому випадку початкові значення зони Z1 розраховуються з використанням алгоритму початкових даних для зони Z1, алгоритму шифрування для зони Z1 і т.д. Для гарантування безпеки налаштування безпеки зони Z1 не передаються в зони U2 і навпаки.

50 Етап 1380H ілюструє маршрутизацію пакета даних SDNP 1374D від медіа-вузла $M_{a, j}$, що знаходиться на медіа-сервері 1220J, на медіа-вузол $M_{a, s}$, що знаходиться на медіа-сервері 1220S. Перехід пакета SDNP 1374D також здійснюється з використанням адрес SDNP "SDNP Addr MJ" і "SDNP Addr MS", які не можна розпізнати в мережі Інтернет. Корисне навантаження 1373B пакета SDNP 1374D скремблюється та зашифровуються з використанням налаштувань
55 безпеки зони Z1, а субпакет SDNP, що міститься в пакеті даних SDNP 1374D, в якому перебувають дані всередині корисного навантаження 1313У, також форматується специфічно відповідно до безпечного динамічного мережевого протоколу для зони Z1.

Цей процес пересилання пакета між вузлами в хмарі SDNP може виникати один раз або повторюватися багаторазово, при цьому кожен повтор припускає повторне формування пакета та реалізацію операції зміни маршруту 1373.

Останній перехід у хмарі пакета SDNP 1374E, показаний у вигляді етапу 1380J на рис. 110D, відбувається аналогічним чином з використанням адрес SDNP "SDNP Addr MJ" і "SDNP Addr MS", які не можна розпізнати в мережі Інтернет. Пакет даних SDNP 1374E маршрутизується від медіа-вузла $M_{a,s}$, що знаходиться на медіа-сервері 1220S, на шлюзний медіа-вузол SDNP $M_{a,d}$, що знаходиться на медіа-сервері 1220D. Корисне навантаження 1373C всередині пакета даних SDNP 1374E скремблюється та зашифровується з використанням налаштувань безпеки SDNP зони Z1, а субпакет SDNP, що міститься в пакеті даних SDNP 1374E, в якому перебувають дані всередині корисного навантаження 1313C, також форматується специфічно відповідно до безпечного динамічного мережевого протоколу для зони Z1.

На етапі 1380K відбувається маршрутизація пакета даних 1374G з безпечної хмари по шляху зі шлюзного медіа-вузла $M_{a,d}$, що знаходиться на медіа-сервері 1220D, до клієнтського вузла $C_{1,1}$, що знаходиться в додатку 1335 на стільниковому телефоні 32. Ця остання маршрутизація IP-пакета 1374G по лінії зв'язку "останньої милі" з використанням IP-адрес "IP Addr MD" і "IP Addr CP", які можна розпізнати в мережі Інтернет, за винятком того, що корисне навантаження 1374 всередині IP-пакета 1374G скремблюється та зашифровується з використанням налаштувань безпеки зони SDNP U1, а субпакет SDNP, що міститься в пакеті даних SDNP 1374G, в якому перебувають дані всередині корисного навантаження 1374 також форматується специфічно відповідно до безпечного динамічного мережевого протоколу для зони U1. Після доставки вмісту даних корисного навантаження 1374 у додаток 1335 на стільниковому телефоні 32, динамік 1388B перетворює цифровий код у звук 1384A за допомогою аудіокодеку (не показано).

На етапі 1380L, зображеному на рис. 110E, особа, якій адресовано виклик, відповідає голосовим сигналом, що подається в напрямку, протилежному напрямку початкового обміну даними. Звукові хвилі 1384B перехоплюються мікрофоном 1383B, що перетворює їх у цифровий код за допомогою аудіокодеку (не показано), що проводиться додатком 1335 на стільниковому телефоні 32. Голосові дані комбінуються з субпакетом зони SDNP U1 з використанням налаштувань безпеки SDNP для зони U1, формуючи корисне навантаження 1375, і направляються з "IP Addr CP" на "IP Addr MD" з використанням IP-пакета 1374H. Ця маршрутизація IP-пакета 1374H по лінії зв'язку "останньої милі" відбувається з використанням IP-адреси, яку можна розпізнати в мережі Інтернет, за винятком того, що корисне навантаження 1375, що міститься в пакеті даних 1374H, скремблюється та зашифровується з використанням налаштувань безпеки зони SDNP U1, а субпакет SDNP, що міститься в пакеті SDNP 1374H, в якому є дані всередині корисного навантаження 1375, також форматується специфічно відповідно до безпечного динамічного мережевого протоколу для зони Z1.

Як показано на етапі 1380M, після одержання IP-пакета 1374H шлюзний медіа-вузол $M_{a,d}$, що знаходиться на сервері 1220D, перетворює адресацію на маршрутизацію SDNP і направляє пакет даних SDNP 1374J і його корисне навантаження 1376A на медіа-вузол $M_{a,j}$, що знаходиться на комп'ютері-сервері 1220U, з використанням налаштувань безпеки зони Z1. Наскрізний обмін інформацією SDNP може бути єдиним переходом від вузла до вузла або передбачати передачу через ряд медіа-вузлів, при цьому кожен перехід передбачає повторне формування пакета та операцію зміни маршруту 1373.

На етапі 1380N, показаному на рис. 110F, пакет даних SDNP 1374K і його корисне навантаження 1376B, специфічне для зони Z1, направляються з медіа-вузла $M_{a,j}$, що знаходиться на комп'ютері-сервері 1220J, на шлюзний медіа-вузол $M_{a,f}$, що знаходиться на комп'ютері-сервері 1220F. Адреси SDNP "SDNP Addr MJ" і "SDNP Addr MF", використовувані в пакеті SDNP 1374K, являють собою адреси, специфічні для SDNP, аналогічні адресам у мережі NAT, вони не відповідають дійсній маршрутизації по мережі Інтернет. На етапі 1380P шлюзний медіа-вузол $M_{a,f}$ перетворює вміст вхідного пакета даних з корисного навантаження 1376B, специфічного для зони Z1, у корисне навантаження 1377 зони U2, і з використанням IP-адреси "IP Addr MF" і "IP Addr TB" направляє IP-пакет 1374L на клієнтський вузол $C_{2,1}$, що знаходиться на планшеті 33, як показано на рис. 109. Далі додаток 1335 вилучає дані з корисного навантаження 1377 і після дешифрування та дескремблювання перетворює цифровий код з використанням аудіокодеку (не показано) у звукові хвилі 1384B, які подаються динаміком 1388A.

Повну інформацію про всю послідовність обміну інформації через порт з метою ініціювання виклику та маршрутизації голосового сигналу від клієнта, що подає виклик, тобто планшета 33, до викликуваної особи, тобто стільникового телефону 32, представлено на рис. 111A. Відповідно до зображення, IP-команда та пакет керуючих сигналів 1374A використовуються для отримання контактної інформації з метою визначення маршруту, а IP-пакет даних 1374B використовується для ініціювання маршрутизації по лінії зв'язку "останньої милі" з використанням IP-адрес для досягнення шлюзного вузла SDNP $M_{a,f}$ на IP-адресі "IP Addr MF".

Весь обмін інформацією з лінії зв'язку "останньої милі", між планшетом 33 і хмарою SDNP 1114 використовує налаштування безпеки зони U2.

Далі шлюзний медіа-вузол $M_{a, f}$ перетворює маршрутизацію в адреси маршрутизації, специфічні для SDNP, і використовує пакети SDNP1374C, 1374D і 1374E для послідовної передачі інформації через хмару SDNP 1114 від "SDNP Addr MF" на "SDNP Addr MJ", "SDNP Addr MS" і "SDNP Addr MD"; при кожному з цих переходів використовуються налаштування безпеки зони Z1. Ця послідовність функціонально еквівалентна пакету даних SDNP 1374F, що направляє пакет інформації, який обмінюються, від "SDNP Addr MF" безпосередньо на "SDNP Addr MD". Через відсутність пристрою спостереження за маршрутизацією при передачі через порт, метою якого є контроль факту доставки пакета, подачу команд і керування маршрутизацією пакета в хмарі SDNP 1114 можна реалізувати одним із двох способів. При одному зі способів реалізації адреси джерела та місця призначення кожного з пакетів даних SDNP 1374C, 1374D і 1374E точно та однозначно визначається покроковий маршрут руху пакета по мережі SDNP, при цьому шлях при одноканальному обміні інформацією обирається шлюзним медіа-вузлом заздалегідь із таким розрахунком, щоб забезпечити найменшу затримку в ході переміщення. При альтернативному способі реалізації один пакет, "який проходить від шлюзу до шлюзу", наприклад, пакет даних SDNP 1374F, використовується для визначення вузлових шлюзів SDNP при вході в хмару та виході з неї, а не для задавання точного маршруту передачі. При такому варіанті реалізації щоразу при прибутті пакета на медіа-вузол SDNP цей медіа-вузол встановлює наступний його перехід здебільшого в такий же спосіб, яким відбувається маршрутизація через Інтернет, за винятком того, що медіа-вузол SDNP автоматично вибирає шлях, який характеризується найменшою затримкою відправлення, тоді як при передачі через Інтернет це не так.

Нарешті, коли пакет 1374E досягає шлюзного відео-вузла $M_{a, d}$ на "SDNP Addr MD", шлюзний медіа-вузол $M_{a, d}$ створює IP-пакет даних 1374G, перетворюючи вхідний пакет даних в IP-адреси "IP Addr MD" і "IP Addr CP", і змінює налаштування безпеки з таким розрахунком, щоб вони відповідали налаштуванням для зони U1.

Інший варіант зведеної інформації про цей маршрут показано на рис. 111B, він складається з трьох переходів всередині хмари 1441C, 1441D і 1441E, а також двох маршрутів по лінії зв'язку "останньої милі" 1441B і 1441F. Адреси пакета, показані під картою хмари, є комбінацією двох форм адрес пакета при передачі: маршрутизація з використанням IP-адреси та маршрутизація з використанням адреси SDNP, аналогічно використанню адрес NAT. Зокрема, адреси пакета 1442A і 1442F являють собою IP-адреси в мережі Інтернет, тоді як адреси пакета 1442C і 1442D являють собою IP-адреси SDNP. Адреси пакета, використовувані шлюзними медіа-вузлами, містять як IP-адреси, так і адреси SDNP, а це означає, що шлюзні вузли SDNP відповідають за передачу адреси, а також за перетворення налаштувань безпеки зони U2 в налаштування безпеки зони Z1 і перетворення налаштувань безпеки зони Z1 в налаштування безпеки зони U1.

Аналогічним чином на рис. 112A показано зведену інформацію про дані, що направляються у відповідь при обміні інформацією, вона передбачає передачу пакета даних 1374J зони U1 у лінії зв'язку "останньої милі" з використанням IP-адрес "IP Addr CP" і "SDNP Addr MD", маршрутизацію в хмарі SDNP з використанням адрес SDNP "SDNP Addr MD", "SDNP Addr MJ" і "SDNP Addr MF" для передачі пакетів даних 1374K і 1374L, специфічних для зони Z1, а також передачу пакета даних 1374J у зоні U2 по лінії зв'язку "останньої милі" з використанням IP-адрес "IP Addr CP" і "SDNP Addr MD". Відповідну карту маршрутизації в хмарі показано на рис. 112B, де перехід "останньої милі" 1441H та перехід "останньої милі" 1441L припускають використання лише IP-адрес 1442G і 1442L; переходи в хмарі 1441J і 1441K використовують тільки адреси SDNP, а шлюзні медіа-вузли $M_{a, d}$ і $M_{a, f}$ здійснюють передачу між IP-адресами та адресами SDNP 1442H і 1442K.

На рис. 113A показано схему, яка ілюструє, яким чином формується пакет SDNP. При обміні інформацією в режимі голосової або відео-конференції голосовий сигнал або відеосигнал 1384A перетворюються в аналогові електричні сигнали мікрофоном 1383A, після чого оцифровуються аудіовідеопристроєм CODEC 1385. Отримана в результаті цього смуга цифрових даних 1387, що являє собою послідовність сегментів даних, розташованих за абеткою (9A, 9B і т. д.), надалі піддається операції синтаксичного аналізу 1386 з метою формування дрібнішого пакета даних 1388, в який входять аудіо- або відеоматеріали, після чого сміттева інформація 1389 уводиться шляхом виконання одноканальної операції розділення 1106. Одноканальна операція розділення 1106 припускає синтаксичний аналіз 1386 довгого пакета 1387 з його перетворенням у пакет 1388 менших розмірів і введення "сміттевих" даних 1389, що приводить до формування збільшеного пакета 1390, що складається з двох секцій, одна з яких містить субпакет Hdr 9, а інша – сміттевий субпакет J. У смузі сегментів даних, що містяться в Hdr 9 і Hdr J, присутні

аудіо- і відеоматеріали в пакеті 1388 з деяким навантаженням у вигляді сегментів "сміттєвих" даних. Сегменти даних після Hdr J не містять корисної інформації. Надалі операція SSE (з англ. Splitting-Scrambling-Encryption — розділення-скремблювання-шифрування) 1213 проводить скремблювання даних з попереднього пакета 1388 з метою формування смуги даних 1391, додає преамбулу SDNP 1399A для формування пакета SDNP 1392 і надалі проводить шифрування всього пакета, за винятком преамбули SDNP, формуючи скрембльоване та шифроване корисне навантаження 1393A, що у свою чергу завантажується в пакет SDNP 1374B з адресою джерела "IP Addr TB" і адресою місця призначення "IP Addr MF", готовий до маршрутизації. субпакети Hdr 9 і Hdr J забезпечують можливість ідентифікації кожної із частин компонента в межах корисного навантаження. Завдання та формат субпакетів і преамбули SDNP описані далі по тексту даного опису.

Аналогічним чином сегмент даних 9G і наступні сегменти смуги даних 1387 формуються в додаткові пакети SDNP.

На рис. 113B показано низку інших методів, якими можна користуватися для формування корисного навантаження з початкових послідовних даних. Наприклад, смугу даних 1387 з аудіокодеку 1385 можна піддати синтаксичному аналізу та розділити іншим способом. Відповідно до зображення, сегменти даних 9A, 9B, 9D і 9F збирають у секцію Hdr 91, замінюючи відсутні сегменти даних сміттєвою інформацією, тоді як сегменти даних 9C і 9E збирають у секцію Hdr 92, у підсумку формуючи пакет даних 1394. Далі сегменти даних у кожному розділі субпакета скремблюють, для того щоб індивідуальні сегменти даних у полі даних 1399C після Hdr 91 не змішувалися з сегментами даних у полі даних 1399E після Hdr 92. Сформований у результаті пакет 1395 складається з преамбули SDNP 1399A, першого субпакета даних 1399B, оснащеного ярликом Hdr 91, першого поля даних 1399C, другого субпакета даних 1399D (Hdr 92) і другого поля даних 1399E. Допускається користування іншими методами для передачі сегментів 9A-9F смуги даних 1387 через різні поля даних. Представлена схема подана тільки для ілюстрування.

Пакет 1395, що містить кілька полів даних, розділених множинними субпакетами, надалі можна зашифрувати одним із декількох способів. При шифруванні всього пакета має місце шифрування всіх даних, що містяться в пакеті SDNP 1395, за винятком даних, що містяться в преамбулі SDNP 1399A, тобто увесь вміст усього першого субпакета 1399B, першого поля даних 1399C, другого субпакета даних 1399D і другого поля даних 1399E зашифровується з формуванням пакета SDNP 1396, що складається з незашифрованої преамбули SDNP 1399A і зашифрованого тексту 1393A. Замість цього, при шифруванні повідомлення пакет SDNP 1397 складається з двох окремо зашифрованих смуг зашифрованого тексту, а саме смуги зашифрованого тексту 1393B, що складається з зашифрованих даних субпакета даних 1399B і поля даних 1399C, а також смуги зашифрованого тексту 1393C, що складається з зашифрованих даних субпакета даних 1399D і поля даних 1399E. При іншому способі реалізації даного винаходу, що називається шифруванням тільки даних, здійснюється шифрування тільки полів даних 1399C і 1399E з формуванням смуг зашифрованого тексту 1393D і 1393E, однак субпакети даних 1399B і 1399D не зачіпають. Сформований у результаті пакет SDNP 1398 складається з тексту, що читається, для преамбули SDNP 1399A, першого субпакета даних 1399B, а також другого субпакета даних 1399D і рядків шифрованого тексту 1393D і 1393E, які являють собою окремо зашифровані версії полів даних 1399C і 1399E, відповідно.

При одноканальному обміні інформацією для ретрансляції необхідної інформації про маршрутизацію та пріоритетність на наступний медіа-вузол корисне навантаження SDNP 1400, показане на рис. 114, повинне містити необхідну інформацію. Ці дані містяться або в преамбулі SDNP 1401, або в субпакеті поля даних 1402. Преамбула SDNP 1401 містить інформацію, що стосується всього пакета, включаючи опис ряду полів даних "Fld #", при цьому кількість полів може досягати восьми, а довжина кожного поля даних дорівнює "L Fld X", де при даному варіанті реалізації X може відповідати наявності від 1 до 8 полів, зони SDNP, в якій був сформований пакет SDNP, наприклад, зона Z1, двох наборів початкових даних і двох ключів, генерування яких здійснено з використанням двох спільних секретів.

субпакет поля даних 1402 створюється в незмінному форматі для кожного з X полів даних. субпакет поля даних 1402 містить адресу, типову для місця призначення, а також адресу призначення певного поля даних, тобто місце призначення саме цього переходу в хмарі. Адреса місця призначення кожного з полів даних у даному пакеті завжди однакова, оскільки пакет залишається незачепленим до моменту його надходження на наступний медіа-вузол. Однак якщо пакет розділяється на кілька пакетів, то адреса місця призначення поля для кожного з пакетів відрізняється від адреси місця призначення поля в кожному з інших пакетів, на які зроблена розбивка, якщо пакети направляються на різні медіа-вузли.

При пересиланні по декількох маршрутах і по решітчастій мережі адреси місця призначення поля використовуються при поділі та змішуванні різних полів, використовуваних у динамічній маршрутизації.

Тип адреси наступного переходу в міру руху пакета по мережі може змінюватися. Наприклад, він може складатися з IP-адреси між клієнтом і шлюзом, а також адреси SDNP або "поштового індексу" SDNP після його входу в хмару SDNP. Адреса місця призначення може складатися зі специфічного для SDNP коду маршрутизації, тобто адреси SDNP, "поштового індексу" SDNP або адреси IPv4 або IPv6, адреси NAT, номери телефону традиційної телефонної мережі тощо.

Поле пакета, якому присвоєно ярлик "Field Zone" ("Зона поля"), містить опис зони, в якій було сформовано відповідне поле, тобто показує, чи було виконане раніше шифрування або скремблювання, виконане з використанням налаштувань зони U1, Z1, U2 і ін. У деяких випадках дескремблювання або дешифрування пакета даних потребує додаткової інформації, наприклад, ключа, початкових даних, часу або стану; у такому разі поле пакета, позначене міткою "Field Other" ("Інше поле") може використовуватися для перенесення інформації, специфічної для поля. Поле пакета з ярликом "Data Type" ("Тип даних") при його використанні полегшує маршрутизацію, специфічну для контексту, розпізнаючи дані, раніше зроблений відеозапис, текст і комп'ютерні файли, не потребуючи обміну в режимі реального часу даними з пакетів даних, що містять інформацію, яка змінюється згодом, наприклад, голосовий запис і відеозапис у режимі реального часу, тобто відрізняючи маршрутизацію в режимі реального часу від даних, не отриманих у цьому режимі. До типів даних належать голосові, текстові записи, відеозаписи в режимі реального часу, дані, програмне забезпечення та ін.

Поля пакетів, оснащені ярликами "Urgency" ("Терміновість") і "Delivery" ("Доставка"), використовуються одночасно для визначення кращого шляху маршрутизації даних у конкретному полі даних. Терміновість припускає такі категорії як "відсутність обов'язковості швидкого виконання", "звичайна швидкість виконання", "пріоритетність виконання" і "необхідність негайного виконання". Доставка містить у собі різні маркери якості обслуговування для категорій "звичайна", "з запасом", "спеціальна" і "VIP". В одному з варіантів реалізації даного винаходу подвійний розмір різних полів даних, показаний у таблиці 1403, обрано з метою мінімізації ширини смуги частот, необхідної для обміну інформацією. Наприклад, показані пакети даних можуть мати розмір від 0 до 200 байт, тоді як наявність восьми пакетів розміром 200 байт в одному полі даних означає, що один пакет SDNP може переносити 1600 байт даних.

Двоканальний обмін інформацією. При варіанті реалізації двоканальної передачі даних SDNP, показаний на рис. 115, вміст рухається по каналах передачі медіа-файлів від клієнтського вузла C_{2,1}, що знаходиться на планшеті 33, до шлюзного медіа-вузлу M_{a, f} з маршрутизацією "останньої милі" в зоні U2 з наступною маршрутизацією по решітчастій мережі зони Z1, що знаходиться на комп'ютерах-серверах 1118, з подальшим пересуванням від шлюзного медіа-вузла M_{a, d} по "останній милі" в зоні U1 до клієнта C_{1,1}, що знаходиться на стільниковому телефоні 32. Керування маршрутизацією виконується IP-пакетом 1374B "останньої милі", пакетом SDNP 1374F по решітчастій мережі SDNP, а також IP-пакетом 1374G "останньої милі".

Паралельно з перенесенням медіа-файлів і вмісту, клієнт C_{2,1}, що обмінюється інформацією з сигнальним вузлом S_s, що знаходиться на сигнальному сервері 1365, пересилає початкові значення 929 і код дешифрування 1030 клієнтові C_{1,1} через сигнальний сервер S_d, при цьому початкові значення 929 і код дешифрування 1030 ґрунтуються на часі або стані, коли клієнт C_{2,1} відправив їх. При обміні налаштуваннями безпеки, такими як ключі та початкові значення (які відомі також як права безпеки), безпосередньо між клієнтами по маршруту передачі сигналів 1405, а не через зону Z1, додатково реалізується наскрізна безпека з усуненням ризику в зоні Z1 для оператора мережі, що отримує доступ до налаштувань безпеки, знижуючи рівень безпеки в зоні U1 або зоні U2. Цей варіант реалізації передбачає лише інший ступінь безпеки при обміні інформацією в мережі SDNP. Наприклад, початковими даними 929 можна користуватися для скремблювання та дескремблювання пакетів даних у клієнтських додатках. Аналогічно, відповідно до зображення, ключ дешифрування 1030 дає можливість відкрити шифроване повідомлення тільки клієнтові C_{1,1}. Оскільки ключ 1030 і числові початкові значення 929 в жодному випадку не передаються через зону Z1, то оператор мережі не може знизити рівень безпеки в мережі. Коли пакети даних надходять у шлюзний вузол M_{a, f} від клієнта C_{2,1}, пакети даних що надходять, вже зашифровані та скрембльовані. Пакети, отримані клієнтом C_{1,1} зі шлюзного вузла M_{a, d}, перебувають у такому ж скрембльованому та/або шифрованому вигляді, як і ті, які відправляє клієнт C_{2,1}, і місцем призначення яких є шлюзний вузол M_{a, f}. Динамічне скремблювання та шифрування в мережі відбувається в кожному з вузлів (хоча це

чітко не показано на рис. 115), що є другим рівнем безпеки, полегшеним завдяки наявності хмари SDNP. Інакше кажучи, цей зовнішній і наскрізний рівень безпеки, що являє собою обмін правами безпеки безпосередньо між клієнтами, існує на додаток до динамічного скремблювання та шифрування, здійснюваних безпосередньо в хмарі SDNP.

Як показано на рис. 115, сигнальні вузли S_s і S_d видають інструкції медіа-вузлам $M_{a,f}$ і $M_{a,d}$ на предмет маршрутизації даних з "IP Addr TB" на "IP Addr MF" у зоні U2 з використанням IP-пакета 1374B, з "SDNP Addr MF" на "SDNP Addr MD" у зоні Z1 з використанням пакета 1374F, а також з "IP Addr MD" на "IP Addr CP" у зоні U1 з використанням IP-пакета 1374G. При такому варіанті реалізації, оскільки сигнальні вузли S_s і S_d обмінюються інформацією тільки безпосередньо через клієнтські вузли $C_{2,1}$ і $C_{1,1}$, а також опосередковано через пакети даних по каналу передачі медіа-файлів з вузлами $M_{a,f}$ і $M_{a,d}$, єдиною інструкцією з маршрутизації для решітчастої мережі є передача від одного шлюзу до іншого з використанням пакета SDNP 1374F. Сигнальні сервера S_s і S_d не можуть обмінюватися інформацією з проміжним медіа-вузлами в межах решітчастої мережі. Тому у варіанті реалізації, показаному на рис. 115, медіа-вузли управляють динамічною безпекою в хмарі у вигляді одноканальної системи обміну інформацією, тоді як сигнальні вузли використовуються для сприяння забезпеченню наскрізної безпеки за межами хмари SDNP, тобто поза зоною Z1.

В іншому варіанті реалізації пересилання даних SDNP двома каналами, показаному на рис. 116, сигнальні вузли S_s і S_d , що знаходяться на серверах 1365, сприяють забезпеченню наскрізної безпеки клієнтів і одночасно керують динамічною маршрутизацією та безпекою у хмарі SDNP. Сигнальні вузли S_s і S_d не тільки забезпечують пересилання числових початкових значень 929 і ключа дешифрування 1030 між клієнтськими вузлами $C_{2,1}$ і $C_{1,1}$ на всьому маршруті з використанням маршруту передачі сигналів 1045, але й пропускають через себе початкові значення 929, специфічні для зони, і ключ дешифрування 1030, а також інструкції з маршрутизації із зазначенням послідовного переміщення між вузлами, використовуючи динамічний пакет SDNP 1374Z, переданий каналом передачі сигналів 1406 до кожного окремого медіа-вузлу решітчастої мережі, через які проходять пакети інформації, обмін якої здійснюється, і вміст. За рахунок цього сигнальні вузли S_s і S_d , які керують маршрутизацією та безпекою, а також медіа-вузли, що знаходяться у мережі, переносять вміст і виконують інструкції, видані сигнальними вузлами S_s і S_d . При такому виконанні або медіа-вузли, або сигнальні вузли S_s і S_d відповідальні за відстеження того, які медіа-сервери працюють у режимі он-лайн, а які – ні, а також які їх динамічні IP-адреси в цей момент.

Триканальний обмін інформацією. Вищого рівня безпеки та поліпшених параметрів мережі можна досягти, розділяючи відповідальність за відстеження вузлів у мережі та реальну передачу даних. При такому підході, надлишкова (резервована) мережа серверів, які називаються "серверами імен", безперервно стежить за мережею та її медіа-вузлами, забезпечуючи сигнальним серверам можливість виконувати свої завдання по маршрутизації та обміну даними щодо безпеки, а також даючи медіа-серверам можливість сконцентруватися на виконанні інструкцій з маршрутизації, що надходять від сигнальних вузлів. У такий спосіб забезпечується рішення, іменоване в цьому документі "триканальною" системою, що показана на рис. 117, де сервер імен 1408, на якому знаходиться вузол NS сервера імен, веде перелік активних вузлів SDNP у мережі, що являє собою перелік мережеских вузлів 1410. Після отримання запиту з сигнального вузла S, що знаходиться на сигнальному сервері 1365, вузол NS сервера імен, що знаходиться на сервері імен 1408, пропускає через себе опис мережі, тоді як сигнальний вузол S відслідковує та реєструє стан і затримку відправлення між всіма медіа-вузлами в хмарі SDNP 1114, як показано в таблиці стану мережі 1409, у тому числі в зонах U2, Z1, U1 та інших. У процесі подачі виклику сигнальний вузол S передає інструкції з маршрутизації на кожен вузол, що бере участь у запланованому переміщенні пакета даних по мережі, включаючи інструкції для "останньої милі" зони U2 на клієнтський вузол $C_{2,1}$, який знаходиться на планшеті 33, а також інструкції з маршрутизації для зони Z1 для всіх проміжних медіа-вузлів у безпечній хмарі SDNP 1114, використовуваних для передачі вмісту медіа-файлів у пакетах даних SDNP.

Для підтримки оновленого опису мережі щоразу, коли пристрій входить у мережу, дані про його стан і його IP-адреси, а також адреси SDNP, а в деяких випадках про обидві адреси, передаються на сервер імен 1408, як показано на рис. 118. Дані про стан мережі та/або адреси надалі зберігаються в таблиці мережеских адрес 1415, що зберігається в додатку 1335, який працює на планшеті 33 або стільниковому телефоні 32, додатку 1411, що працює на ноутбучі 35, або на робочому столі (не показано); вбудовані додатки 1412 і 1413 працюють на самохідному пристрої 1255 або на пристрої Інтернету речей 34, що графічно зображено у вигляді холодильника. Таблиця мережеских адрес 1415 також проводить відстеження стану всіх

медіа-серверів у хмарі, включаючи, приміром, медіа-вузол $M_{a, f}$, що знаходиться на комп'ютері 1220F, і медіа-вузол $M_{a, d}$, що знаходиться на комп'ютері 1220D. Таблиця мережесих адрес 1415 забезпечує реєстрацію адрес маршрутизації всіх пристроїв, підключених до мережі. Практично у всіх випадках IP-адреса або адреса SDNP мережевого пристрою реєструється та відстежується в таблиці мережесих адрес 1415. В інших випадках, наприклад, для медіа-серверів і додатково для персональних мобільних пристроїв, на яких працюють додатки обміну інформацією SDNP, у таблиці мережесих адрес можуть реєструватися як IP-адреси, так і адреси SDNP, необхідні для передачі адрес у шлюзних медіа-вузлах.

Тоді як вузол NS сервера імен веде вичерпний опис мережі, сигнальний вузол S, що знаходиться на сигнальному сервері 1365, показаному на рис. 119, веде таблицю затримок розповсюдження 1416 між всіма комбінаціями медіа-вузлів, зазначеними в таблиці. Таблиця затримок розповсюдження 1416 оновлюється розрахунками затримки, отриманими для звичайного руху пакетів даних через медіа-вузли мережі, що умовно показано у вигляді секундів 1415A, 1415B і 1415C, які відстежують затримки відправлення між медіа-серверами 1220D і 1220F, 1220F і 1220H, а також 1220D і 1220H, відповідно. У разі, якщо поточний трафік малий або рідкий, мережа SDNP використовує також тестові пакети для перевірки стану з'єднання. Один із методів, які передбачають використання тестових пакетів показано на рис. 120, де завдання медіа-серверу ставить сигнальний сервер, вони полягають у тому, щоб відправити серію груп пакетів; при цьому розмір відправлених пакетів даних збільшується або збільшується частота їхнього відправлення, при цьому проводиться відстеження затримки. Отриманий у результаті графік завантаження, відображений у вигляді кривої 1417, показує, що максимальне навантаження на певний маршрут обміну інформацією або лінію зв'язку повинен обмежуватися за розміром, або його інтенсивність не повинна перевищувати максимального навантаження, зображеного у вигляді лінії 1418.

В умовах, коли зазначена вище інформація щодо мережі, адрес її вузлів, а також затримок відправлення в ній легкодоступна на серверах імен і сигнальних серверах, високої якості обслуговування при обміні інформацією найкраще досягти при використанні триканального обміну інформацією згідно зі схемою, показаною на рис. 121. Відповідно до зображення, сигнальний вузол S, що знаходиться на сигнальному сервері 1365, повністю керує маршрутизацією даних, що переміщуються через медіа-сервери 1118 і до клієнтів 1335 шляхом розподілу пакетів SDNP 1420, що складаються з даних про маршрутизацію "вузол-вузол" 1374Z, характерних для цієї зони початкових числових значення 929 і ключів дешифрування 1030. При встановленні виклику клієнтський вузол $C_{2,1}$ (у цьому випадку цей додаток SDNP 1335 на планшеті 33) контактує з вузлом NS сервера імен, що знаходиться на сервері імен 1406, для власної реєстрації в мережі та пошуку в ній найближчого сигнального сервера; при цьому для ініціювання виклику він контактує з сигнальним вузлом S на сигнальному сервері 1365. Потім сигнальний вузол S керує маршрутизацією, а медіа-сервери здійснюють відповідну маршрутизацію даних, змінюючи налаштування безпеки для кожної із зон U2, Z1 і U1.

З огляду на важливість імені сервера в підтримці оновленого списку мережесих вузлів 1410, показаного на рис. 122, вузол NS сервера імен, розташований на сервері імен 1408, працює разом з одним або декількома надлишковими (резервованими) серверами, що ілюструється у вигляді резервного вузла сервера імен NS2, який функціонує на резервному сервері імен 1421. Той самий метод резервування реалізований для сигнальних серверів з метою гарантування постійної можливості адресації виклику або маршрутизації пакета. Як показано на рис. 123, сигнальний вузол S, розташований на сигнальному сервері 1365, оснащений резервним сигнальним вузлом S2, який знаходиться на резервному сигнальному сервері 1422, що автоматично перемикається у разі несправності сигнального сервера 1365 або атаки на нього.

Обмін інформацією з використанням триканальної маршрутизації пакета SDNP відповідно до даного винаходу показано на рис. 124A, де на етапі 1430A пристрій або пристрій, який викликає, входить у мережу. Для реалізації цього клієнтський додаток 1335 на планшеті 33 в автоматичному режимі контактує та реєструється під іменем вузла NS сервера імен, який знаходиться на сервері імен 1408. Ця подія пов'язана з входом клієнта в мережу, а не обов'язково з адресацією виклику. У процесі реєстрації вузол NS сервера імен передає список серверів імен, тобто перелік 1431 серверів імен SDNP, і додатково список сигнальних серверів клієнтському додатку 1335. За наявності такої інформації пристрій знаходиться у стані готовності і може здійснювати адресацію виклику SDNP.

На першому етапі в ході дійсної адресації виклику планшет 33 направляє IP-пакет 1450A на вузол NS сервера імен, запитуючи інформацію про маршрутизації та контактну інформацію для визначення місця призначення або особи, якій необхідно адресувати виклик. Запит про контактну інформацію, тобто запит про маршрут 1431, може надійти у вигляді IP-адреси, адреси

SDNP, номер телефону, URL або іншого ідентифікатора для обміну інформацією. На етапі 1480C вузол NS сервера імен, що знаходиться на сервері імен 1408, направляє клієнтському додатку SDNP 1335 адресу призначення одержувача. Відповідь доставляється IP-пакетом 1450B з використанням протоколу TCP як транспортний рівень. При іншому варіанті реалізації

5 клієнт подає запит на одержання інформації про маршрутизацію від сигнального сервера, а сигнальний сервер подає запит на одержання інформації серверу імен.

На етапі 1430D, що показано на рис. 124У, клієнт зрештою отримує можливість ініціювати запит з використанням IP-пакета 1450C з "IP Addr TB" на "IP Addr S", тобто IP-адреси сигнального сервера 1365, на якому знаходиться сигнальний вузол S. Оскільки IP-пакет 1450C несе інформацію про адресу одержувача, а не дані, отримані в режимі реального часу, то бажано, щоб IP-пакет 1450C використовував протокол TCP як транспортний рівень. Користуючись інформацією про затримки у відправленні даних між вузлами мережі, показаною в таблиці 1416, сигнальний вузол S розробляє мережевий план маршрутизації для мережі SDNP 1114, а також "останньої милі" для шлюзних серверів SDNP, а на етапі 1430E ця

10 інформація про маршрутизацію передається в хмару SDNP 1114. Сигнальний сервер направляє пакет даних з командними сигналами та сигналами керування на кожен медіа-сервер з метою видачі їм інструкцій про те, як їм працювати із вхідними пакетами даних. Пакет даних з командними сигналами та сигналами керування схожий на звичайний пакет даних, за винятком того, що замість перенесення аудіо-інформації його корисне навантаження складається з серії інструкцій, що інформують медіа-вузол про те, яким чином здійснювати маршрутизацію пакета

20 зі спеціальним ідентифікаційним ярликом, адресою SDNP або "поштовою адресою" SDNP до нового місця призначення. Замість цього відповідно до представленого вище опису, при реалізації зі створенням розподільних мереж жоден із сигнальних серверів не розробляє повного плану маршрутизації, замість цього серія сигнальних серверів розробляє послідовні частини плану маршрутизації в міру переміщення пакета по мережі SDNP.

25

Далі на етапі 1430F сигнальний вузол S направляє додатку 1335 на планшеті 33 адресу шлюзного медіа-вузла, ключі дешифрування 1030 зони U2, початкові значення 929 і інші параметри безпеки, необхідні для безпечної відправки першого пакету по каналах зв'язку "першої милі".

Після отримання планшетом 33 налаштувань безпеки зони U2 на етапі 1430F, він ініціює виклик з пакетом SDNP 1450D, як показано на рис. 124C. Звук, що являє собою голосові хвилі 1384A, які вловлюються мікрофоном 1383A, перетворюється в цифрову інформацію аудіокодеком (не показано) і передається в додаток 1335 на планшеті 33. Комбінуючи аудіо-дані з маршрутизацією адреси та іншою зібраною інформацією при формуванні субпакета SDNP, додаток 1335 формує пакет SDNP 1450D для маршрутизації по "останній милі" від "IP Addr TB" на "IP Addr MF" і починає передачу пакета на шлюзний медіа-вузол $M_{a, f}$. субпакет SDNP, вбудований у корисне навантаження пакета даних 1432, може містити інформацію про терміновість, перевагу доставки, протоколи безпеки, а також специфікації типів даних. Субпакет SDNP також містить преамбулу та MAC-адресу, IP-адреси джерела та місця призначення, а також поле протоколу, в основному інформацію 2-го, 3-го та 4-го рівнів з корисним навантаженням, що включає субпакет SDNP, а також усі пакети даних з їхніми власними субпакетами. Оскільки передача пакета SDNP 1450D по "останній милі" відбувається з використанням IP-адрес, передача пакетів схожа на звичайний Інтернет-трафік, за винятком того, що дійсний вміст даних скрембльовано та зашифровано з використанням налаштувань безпеки для зони U2, а субпакет SDNP, що міститься в корисному навантаженні SDNP 1432, що також містить дані, форматується специфічно відповідно до безпечного динамічного мережевого протоколу для зони U2.

40

45

Етап 1430H, який показано на рис. 124C, ілюструє маршрутизацію пакета даних SDNP 1450E від шлюзного медіа-вузла $M_{a, f}$, що знаходиться на медіа-сервері 1220F, до медіа-вузла $M_{a, j}$, що знаходиться на медіа-сервері 1220J у хмарі SDNP. На відміну від маршрутизації IP-пакета даних 1450D по "останній милі", цей перший перехід пакета SDNP 1450D всередині хмари відбувається з використанням адрес SDNP "SDNP Addr MF" і "SDNP Addr MJ", які не можна розпізнати в мережі Інтернет. До того ж, корисне навантаження 1433 скремблюється та зашифровується з використанням налаштувань безпеки SDNP зони Z1, а субпакет SDNP, що міститься в пакеті SDNP Z1, до складу якого входять дані, також форматується специфічно відповідно до спільних секретів для зони Z1. Для гарантування безпеки налаштування безпеки зони Z1 не передаються в зону U2 і навпаки.

50

55

На етапі 1430J, показаному на рис. 124D, пакет даних 1450F маршрутизується з безпечної хмари SDNP від шлюзного медіа-вузла $M_{a, d}$, що знаходиться на медіа-сервері 1220D, до клієнтського вузла $C_{1,1}$, що знаходиться в додатку 1335 на стільниковому телефоні 32. Така

60

маршрутизація по "останній милі" IP-пакета 1450F відбувається з використанням IP-адрес "IP Addr MD" і "IP Addr CP", розпізнаваних у мережі Інтернет, однак корисне навантаження 1434 скремблюється та зашифровується з використанням спільних секретів SDNP зони U1, а субпакет SDNP, що міститься в корисному навантаженні 1434, також форматується специфічно відповідно до спільних секретів. Після доставки вмісту даних 1434 у додаток 1335 на стільниковому телефоні 32, гучномовець 1388B перетворює цифровий код у звукові хвилі 1384A з використанням аудіокодеку (не показано).

Коли вхідний пакет SDNP 1450F одержує додаток 1335 на стільниковому телефоні 32, він може бачити лише адресу останнього медіа-вузла $M_{a,d}$, з якого був відправлений пакет при виході з хмари SDNP. За винятком випадків, коли корисне навантаження SDNP несе інформацію щодо особи, яка здійснила виклик, а також випадків, коли сигнальний вузол S забезпечує доставку цієї інформації особі, яка подала виклик або отримала дані, немає можливості відстежити її походження або її джерело. Ця особливість, тобто "анонімний" обмін інформацією та неможливість відстеження доставки даних, являє собою унікальний аспект обміну інформацією SDNP і властиву їй особливість одноетапної динамічної маршрутизації відповідно до даного винаходу. Мережа SDNP забезпечує доставку інформації про абонента і джерело тільки в тому випадку, якщо абонент, який здійснює виклик, сам прагне поділитися цією інформацією; в іншому разі інформація не надходить – анонімність є умовою за замовчуванням при доставці пакета SDNP. Фактично клієнтський додаток SDNP, який здійснює відправлення, повинен навмисно відправити повідомлення, що інформує особу, якій адресований виклик або якій передається повідомлення, про те, що інформація надійшла від конкретної особи, яка здійснює виклик. Оскільки сигнальний сервер має інформацію про особу, яка здійснює виклик, і маршрутизації пакета, він може визначити маршрут руху пакета даних, що відправляється у відповідь, без жодної перевірки ідентичності особи, яка здійснює виклик.

Замість цього сигнальний сервер може перевіряти ідентичність псевдоніма або аватара, або обмежувати доступ до ідентифікації особи, що здійснює виклик, лише декількома близькими друзями та авторизованими контактами. Анонімність має особливо велике значення в таких додатках як ігри, коли гравець не хоче розкривати свою особистість, особливо з незнайомим суперником. Іншою умовою, що потребує анонімного обміну інформацією, є засоби міжмашинного обміну даними (M2M), мережі Інтернет речей (IoT), міжавтомобільної (V2V), а також обмін інформацією між автомобілем і транспортною інфраструктурою (V2X), коли клієнт не хоче, щоб машини, пристрої або вироби видавали контактну або персональну інформацію пристроям або агентам, які можуть бути ворожими, а також пристроям, якими можуть користуватися кіберзлодії. Для користувача, що висуває особливі вимоги до приватності, голосовий сигнал також можна приховати за допомогою електронних засобів, для того щоб навіть локальний обмін інформацією можна було зробити анонімним.

Як показано на етапі 1430K, зображеному на рис. 124D, у відповідь на пакет, що надійшов, додаток 1335, встановлений на стільниковому телефоні 32, направляє IP-пакет 1450G на сигнальний вузол S, що знаходиться на сигнальному сервері 1365. Пакет, що відсилається, вимагає відповіді на запит з інформацією про маршрутизацію. В одному з варіантів реалізації сигнальний вузол S після цього може направити особі, якій адресовано виклик, інформацію про справжню ідентичність особи, що робить виклик, тоді як прикладна програма SDNP особи, якій адресовано виклик, може видавати відповідь у вигляді повторення у зворотному порядку всього процесу підключення, яке використовувалося для підключення до них, тобто контакту з сервером імен, пошуку їхньої адреси SDNP або IP-адреси, контакту з сигнальним сервером, маршрутизації відповіді і т.д. В іншому варіанті впровадження сигнальний сервер має інформацію про те, звідки прийшов пакет, і встановлює маршрут руху пакета, який направляється у відповідь, і повинен бути спрямований без розкриття контактної інформації особи, яка здійснює виклик.

Незалежно від застосовуваного методу видачі відповіді, на етапі 1430L, зображеному на рис. 124E, IP-пакет, що направляється у відповідь, містить у своєму складі аудіо-дані, що складаються з голосових хвиль 1384B, уловлених мікрофоном 1383B і перетворених в аналогові сигнали, надалі перетворені в цифровий формат аудіокодеком (не показано). Звуковий вміст після обробки, скремблювання, шифрування та пакування стає безпечним корисним навантаженням 1345 IP-пакета 1450H, що маршрутизується від "IP Addr CP" на шлюзний медіа-вузол SDNP "IP Addr MF". Ці IP-адреси розпізнавані в мережі Інтернет, за винятком того, що корисне навантаження 1345 являє собою вміст, скремблований та зашифрований з використанням налаштування безпеки SDNP зони U1, а субпакет SDNP, що міститься в корисному навантаженні 1435, форматується специфічно відповідно до спільних секретів для зони U1.

На етапі 1430M пакет, що відправляється у відповідь, виходить із безпечної хмари SDNP без здійснення будь-яких переходів між вузлами в хмарі SDNP. У цьому випадку шлюзний медіа-вузол $M_{a, f}$, розташований на медіа-сервері 1220F, перетворює вміст пакета 1450H з форми корисного навантаження 1435, специфічної для зони Z1, у корисне навантаження 1436 зони U2, і з використанням IP-адрес "IP Addr MF" і "IP Addr TB" направляє IP-пакет 1450J на клієнтський вузол $C_{2,1}$, що знаходиться на планшеті 33. Така маршрутизація IP-пакета по "останній милі" відбувається з використанням IP-адрес "IP Addr MD" і "IP Addr CP", розпізнаваних у мережі Інтернет, однак корисне навантаження 1436 скремблюється та зашифровується з використанням спільних секретів SDNP зони U2, а субпакет SDNP, що міститься в корисному навантаженні 1436, також форматується специфічно відповідно до безпечного динамічного мережевого протоколу для зони U2. Після доставки на стільниковий телефон 32 додаток 1335, дозвіл на роботу якого наданий SDNP, вилучає дані корисного навантаження та після дешифрування та дескремблювання перетворює цифровий код з використанням аудіокодеку (не показано) у звукові хвилі 1384B, які подаються мікрофоном 1388A. У ході виконання послідовності дій, показаної у вигляді етапів 1430K-1430M, в обміні інформацією бере участь тільки один шлюзний медіа-вузол, в результаті чого після "першої милі" одразу йде "остання миля" (лінії зв'язку між пристроєм кінцевого користувача з вузлом доступу Інтернет-провайдера).

Зведену інформацію про послідовність викликів із триканальним обміном інформацією відповідно до даного винаходу показано на рис. 125A, де додаток 1335, який працює на планшеті 33, з використанням IP-пакетів 1450A і 1450B, заснованих на протоколі передачі TCP, і вузол NS сервера імен встановлюють діалог, тоді як при отриманні контактної інформації або IP-адреси особи, з якою здійснюється контакт, планшет 33 видає інструкції сигнальному вузлу S по адресації виклику та встановлює сесію з одержувачем із використанням IP-пакета 1450C, заснованого на протоколі транспортування TCP. Далі звукові хвилі 1384A вловлюються, упаковуються та маршрутизуються медіа-вузлами в місця свого призначення з використанням комбінації IP-пакетів 1450D і 1450F для "першої милі" і "останньої милі", відповідно, а також пакета SDNP 1450E для передачі через хмару SDNP. Отриману в результаті маршрутизацію з планшета 33 на шлюзний медіа-вузол $M_{a, f}$ і далі на шлюзний медіа-вузол $M_{a, d}$ і з нього на стільниковий телефон 32 показано на рис. 125B. У ході всього переміщення, за винятком переходу між вузлами 1453B, використовуються IP-адреси, а не адреси SDNP. Цю послідовність показано на блок-схемі в нижній частині рис. 125B.

Послідовність відповіді показано на рис. 126A, у ході її додаток 1335 на стільниковому телефоні 32, використовуючи IP-пакет даних 1452G, подає питання на сигнальний вузол S з метою відправлення пакета з відповіддю на планшет 32, а шлюзний медіа-вузол маршрутизує голосову відповідь із використанням IP-пакетів 1452H і 1452J. Передача отриманого в результаті пакета, яку показано на рис. 126A, що складається з переходів 1453D і 1453E, практично занадто безпечна, оскільки ця передача відбувається повністю по мережі Інтернет, за винятком маршрутизації через шлюзний медіа-вузол $M_{a, f}$, що підвищує рівень безпеки тільки за рахунок переписування IP-адрес джерела та місця призначення та перетворення налаштувань безпеки пакета даних з тих, які діють для зони U1, в налаштування для зони U2. У цьому прикладі відбувається перехід з одного вузла в інший у хмарі SDNP, недоліком чого є полегшення відстеження та кореляції пакетів даних, що надходять в окремий вузол і виходять із нього, у цьому випадку таким є медіа-сервер 1220F.

У такому разі в маршрут передачі даних корисно додати фіктивний вузол, щоб полегшити зазначення неправильного курсу, як показано на рис. 126C. У такому разі маршрут змінюють із таким розрахунком, щоб він містив адресу другого сервера "IP Addr MF2", що знаходиться або на тому ж сервері, або в групі серверів, для яких встановлена адреса "IP Addr MF", і для того, щоб перетворити вхідний IP-пакет 1452H, що впливає з "IP Addr CP" на "IP Addr MF" у вихідний IP-пакет 1462L, що впливає з "IP Addr MF2" на "IP Addr TB", вставкою проміжного IP-пакета 1452K, що "передає" пакет 1452K з "IP Addr MF" на "IP Addr MF2" або з "SDNP Addr MF" на "SDNP Addr MF2". У процесі передачі змінюється також присвоювання порту. У такому разі не має значення, чи є ця адреса IP-адресою в мережі Інтернет, адресою NAT або адресою SDNP, оскільки пакет даних 1452K ніколи не залишає сервер або групу серверів, тобто він являє собою внутрішню передачу та перехід.

Корисне навантаження "Поля". Обробку корисного навантаження пакета даних, що надходить до клієнта SDNP через шлюзний медіа-вузол, показано на рис. 127, де вхідний IP-пакет 1374B спочатку розпаковується з метою вилучення зашифрованого корисного навантаження, що являє собою зашифрований текст 1393, потім відбувається його дешифрування з використанням відповідного ключа з зони, в якій виконувалося шифрування, з

використанням за необхідності інформації про час і стан. Отримане в результаті цього корисне навантаження складається з тексту 1392, що читається, який у випадку скремблювання повинен бути також дескремблюваний, знов-таки з використанням налаштувань безпеки відповідної зони та стану. Далі з преамбулою SDNP виконується ряд операцій з розкриттям вмісту пакета даних 1391, що складається з різних полів; у цьому випадку він складається з поля 9 із відповідним субпакетом Hdr 9 і сміттевого поля з відповідним субпакетом Hdr J.

При альтернативному варіанті реалізації, що показаний на рис. 127, вхідний пакет 1460 дешифрується та дескремблюється, преамбула видаляється та проводиться його синтаксичний аналіз з утворенням двох дійсних полів даних – поля 6 з відповідним субпакетом Hdr 6 і поля 8 з відповідним субпакетом Hdr 8. Далі ці пакети можуть зливатися з іншими полями, формуючи нові IP-пакети та пакети SDNP, відповідно.

Використання структури даних вкладених полів і пакування декількох полів даних з їх власними субпакетами з формуванням одного корисного навантаження пакета багато в чому схоже на встановлення декількох ящиків всередину ящика більшого розміру. Процес переупаковки SDNP даних з пакування, тобто відкриття ящика з вилученням дрібніших ящиків і їхнім встановленням у нові більші ящики, містить у собі багато варіантів вибору при маршрутизації сегмента даних. Щоб уникнути втрати пакета, бажано, щоб пакети даних однакового походження не входили до складу тих самих полів, куди потрапляють сегменти даних з інших даних, обмін інформацією та комюніке, а залишалися винятково відділеними відповідно до ідентифікації, передбаченої субпакетом, і розташуванням, встановленим пристроєм, що здійснив відправлення. Наприклад, на рис. 128 вхідне корисне навантаження 1461 і 1393 з пакетів даних SDNP або IP-пакетів даних (не показано) дешифрується з використанням операції дешифрування 1032, по можливості з використанням різних ключів дешифрування з різних станів і зон, з утворенням двох корисних навантажень у вигляді текстів 1392 і 1462, що читаються. Операція змішування передбачає комбінування корисних навантажень 1392 і 1462 і, після синтаксичного аналізу, створює вміст для трьох полів – поля 6, що складається з пакета 1464, поля 8, що складається з пакета 1463, і поля 9, що складається з пакета 1459, які в сукупності утворюють вміст даних 1470. Три пакети (1459, 1463 і 1464) можна зберігати окремо або введення до складу довгого пакета. Завдяки субпакетам SDNP кожне поле даних легко ідентифікувати, хоча вони були вилучені з пакета SDNP або IP-пакета, що використовувався для їхньої доставки. У сукупності вміст даних 1470 являє собою дані, що знаходяться на медіа-вузлі саме в той момент. Процес проходить у динамічному режимі, при цьому вміст безупинно змінюється по мірі руху пакетів по мережі SDNP. Після закінчення заданого проміжку часу, коли не залишається причин для того, щоб очікувати надходження яких-небудь даних, вміст даних 1470 розбивається на нові комбінації операцією розділення 1057, тоді як корисне навантаження 1742 містить деякі сегменти даних з кожного з трьох полів, тобто сегменти даних 9C і 9D з поля 9, сегмент даних 8B з поля 8, а також сегменти 6C і 6D з поля 6. Номери цих полів переміщуються в корисне навантаження 1472. Текст, що читається, при бажанні скремблюють, після чого його зашифровують, проводячи операцію шифрування 1026 у поточному стані для поточної зони з формуванням корисного навантаження 1474, готового до складання в пакет SDNP або IP-пакет із маршрутизацією заданим шляхом.

В результаті операції розбивки 1057 також формується друге корисне навантаження 1471, що містить сегменти даних для трьох полів, тобто поле 9 містить сегменти даних 9B, 9A, 9F і 9E, поле 8 містить тільки сегмент даних 8F, а поле 6 містить сегмент даних 6F.

Відповідно до зображення, всі поля в корисних навантаженнях 1471 і 1472 також містять один або більше сегментів "сміттевих" даних. За винятком випадків, коли проведено дескремблювання, скрембльоване корисне навантаження 1471 надалі зашифровується проведенням операції шифрування 1026 у поточному стані та для поточної зони з формуванням корисного навантаження 1473, готового до складання з формуванням пакета SDNP або IP-пакета. Аналогічно, корисне навантаження 1472 зашифровується проведенням операції шифрування 1026 у поточному стані та для поточної зони з формуванням корисного навантаження 1474, готового до складання з формуванням пакета SDNP або IP-пакета. Корисне навантаження 1473 маршрутизується на інший медіа-вузол, ніж корисне навантаження 1474. На цьому малюнку IP-адреса або адреса SDNP, а також, залишкова частина пакета даних вилучені для ясності.

Динамічну природу повторного формування пакетів проілюстровано на рис. 129A, де в момент часу t_4 і при відповідному стані 994 корисні навантаження 1483A і 1483B, що являють собою сегменти даних з полів Fld 91 і Fld 92, відповідно, змішуються з використанням операції змішування 1061 з формуванням гібридного корисного навантаження 1484A. У момент часу t_5 і при відповідному стані 995 операція змішування 1061 комбінує гібридне корисне навантаження

1484A з корисним навантаженням 1484У, що містить дані для Fld 93, з формуванням гібридного довгого корисного навантаження 1485, що складається з сегментів даних 9B, 9A, 9F і 9E, розташованих у скрембльованому порядку в полі 91 з субпакетом Hdr 91, сегмента даних 9C у полі 92 з Hdr 92 і сегмента даних 9D в полі 93 з субпакетом Hdr 93. У момент часу t_f і в стані 999 додаток 1335, встановлений на стільниковому телефоні 32, обробляє гібридне корисне навантаження 1485A і повторно відновлює вихідну послідовність даних 1489A, що складається з сегментів 9A – 9F, розташованих послідовно.

У деяких випадках, описаних вище в цьому описі, може виникнути необхідність тимчасового зберігання деяких сегментів даних або полів у ході очікування прибуття наступних. Ця операція зберігання може здійснюватися всередині будь-якого заданого вузла в мережі SDNP, у тому числі у внутрішніх медіа-вузлах і шлюзових медіа-вузлах. Замість цього зберігання може здійснюватися в клієнтському додатку, встановленому на стільниковий телефон, планшет, ноутбук та ін. Такий приклад показано на рис. 129, де в момент часу t_4 корисні навантаження 1483A і 1483B, що являють собою сегменти даних з полів 91 і 92, змішуються проведенням операції змішування 1061 з утворенням гібридного корисного навантаження 1484A. Це нове корисне навантаження зберігається в неактивному стані в мережевому кеші 1550 або у вигляді його компонентів, що являють собою поля 1485B і 1485C, або у вигляді довгого гібридного корисного навантаження 1484A. Нарешті, у момент t_5 , коли надходить корисне навантаження 1485D, вміст мережевого кеша 1550 звільняється з його подачею на операцію змішування 1061, формуючи в момент часу t_6 і при відповідному стані 996 гібридне корисне навантаження 1486A, що складається з сегментів даних 9A – 9F, розподілених між полями Fld 91, Fld 92 і Fld 93. У момент часу t_f і при стані 999 додаток 1335, встановлений на стільниковий телефон 32, обробляє гібридне корисне навантаження 1486, сформоване з ряду полів, подібне до вихідної послідовності даних 1489A, що складається з сегментів 9A – 9F, розташованих послідовно.

При іншому варіанті реалізації даного винаходу остаточне повторне складання та кешування полів здійснюється всередині додатка 1335 на стільниковому телефоні 32, тобто всередині клієнтського додатка, а не в хмарі SDNP. Як показано на рис. 129C, у момент часу t_4 корисні навантаження 1483A і 1483B, що складаються з сегментів даних з полів 91 і 92, змішуються проведенням операції змішування 1061 з утворенням гібридного корисного навантаження 1484A, яке негайно направляється додатку 1335 на стільниковому телефоні 32 і зберігається в безпечному кеші клієнтського додатка 1551 у вигляді корисних навантажень 1484C і 1484D. Коли корисне навантаження 1485E прибуває в момент часу t_4 і далі направляється в додаток 1335 на стільниковому телефоні 32 у момент часу t_5 і у відповідному стані 995, додаток 1335 у момент t_f отримує здатність відновити вихідну послідовність даних 1489A, що складається з сегментів 9A – 9F, розташованих послідовно.

Зведену блок-схему, в якій наведено повну інформацію про реконструкцію клієнтом пакета SDNP, представлено на рис. 129D, де одноканальний пакет даних 1490, що складається з одного або декількох блоків шифрованого тексту, дешифрується проведенням операції дешифрування 1032 з утворенням тексту, що читається з безлічі полів 1491, і дескремблюється проведенням операції 928 з утворенням смуг тексту, що читається, з безлічі полів 1492A, 1492B і 1492C, які надалі впроваджуються проведенням операції змішування 1061, що включає в себе операцію синтаксичного аналізу 1037 і видалення "сміттєвих" даних (не показано) з утворенням вихідного пакета даних 1493. Наприкінці пакет даних 1493 перетвориться аудіокодеком 1385 у звукові або голосові хвилі 1384A.

Командні сигнали та сигнали керування. Як кінцевий елемент комунікаційної системи SDNP, відповідно до даного винаходу, подача командних сигналів і сигналів керування медіа-вузлами сигнальними вузлами є ключовим компонентом у забезпеченні високої якості обслуговування та малої затримки у відправленні пакетів у режимі реального часу без зниження рівня безпеки та відповідності аудіо-матеріалів. Один із прикладів базового дерева рішень, використовуваного для визначення маршруту та пріоритетності роботи з клієнтами, обміну інформацією, а також пакетів даних, зображено на рис. 130. Відповідно до зображення, коли клієнтський вузол $C_{2,1}$, яким є планшет 33, видає вимогу на адресацію виклику, що подається на сигнальний вузол S, на сигнальний сервер 1365, він вказує в пакеті командних сигналів і керуючих команд 1495A не тільки того, з ким хоче зв'язатися особа, що подає виклик, приміром, голосовий виклик, відеовиклик тощо, але і його терміновість, кращий спосіб доставки, тобто звичайний оптимальний варіант, гарантовану доставку, доставку VIP-особам та ін. Сигнальний вузол S інтерпретує запит на доставку 1499A, користуючись "вибором методу доставки" (етап 1500) виходячи із запиту, бізнес-стану клієнта, історії платежів або будь-якої кількості бізнес-міркувань. Можливе отримання декількох результатів. Якщо клієнт є VIP-особою або кращим клієнтом, виходячи з обсягу або можливості надходження матеріалів, то сесія обміну

інформацією оснащується ярликом VIP. VIP-доставка може передбачати також спеціальну технологію поліпшення робочих параметрів, відому як "пересилання декількома маршрутами" і описану нижче в цьому описі.

Якщо найважливішим фактором, що характеризує файл, є його гарантована доставка, то можна застосовувати гарантовану доставку пакета, тобто відправлення численних резервних копій пакетів і мінімізацію кількості переходів між вузлами з метою мінімізації ризику втрати пакета навіть у тому випадку, якщо це призведе до погіршення параметрів, що характеризують роботу в режимі реального часу. Спеціальна доставка може передбачати процедури автентифікації, специфічні для клієнта. В іншому випадку реалізують звичайну маршрутизацію SDNP. На рис. 130 результат ухваленого рішення про обраний спосіб доставки (етап 1500) разом з адресою та номером телефону 1499В особи, якій повинен бути адресований виклик, використовується для керування маршрутизацією з впливом на операцію "визначення та ранжирування варіантів маршрутизації" (етап 1501). Після ранжирування маршрутів ухвалюють рішення щодо подачі запиту про терміновість 1499С та особливі фінансові міркування, приміром, плату за швидкість, реалізуючи процедуру ухвалення рішення "вибір терміновості доставки пакета" (етап 1502); при цьому результатом вибору може бути звичайна швидкість, пріоритетність доставки, висока терміновість доставки, а також вибір на користь дешевшої "тихохідної" операції для відправлення даних без зниження якості аудіо-записів.

Комбінування варіантів маршрутизації (етап 1501) і вибору терміновості (етап 1502) дає можливість найбільш правильного вибору сигнальним вузлом S маршруту передачі кожного пакета, кадру або сегмента даних (етап 1503). Якщо обраний маршрут пролягає через велику кількість зон, то він буде передбачати різні налаштування безпеки (етап 1504) для кожної зони. Ці дані, що складаються з початкових даних, ключів дешифрування 1030 та іншої, пов'язаної з безпекою, інформації, далі комбінуються з маршрутизацією між усіма вузлами, поділом і змішуванням для пересилання по решітчастій мережі, використовуються для генерування преамбул для кожного пакета даних, у тому числі IP-пакетів для "першої милі" та "останньої милі", які складаються з преамбули 1505А SDNP зони U2, преамбули 1505С SDNP зони U1, а також численних преамбул SDNP зони Z1 для пересилання по решітчастій мережі в SDNP, які в сукупності представлені преамбулою 1505В. Преамбули 1505А, 1505В, 1505С та інші надалі комбінуються з IP-адресами та адресами SDNP з утворенням різних IP-пакетів і пакетів SDNP. До цих інструкцій з маршрутизації належить IP-пакет 1506А, направлений на планшет 33, в якому викладена інформація про маршрутизацію виклику або комюніке з клієнтського вузла C_{2,1} до шлюзового медіа-вузла SDNP, множинних пакетів SDNP 1506В, направлених на медіа-сервери 1118 і використовуваних для маршрутизації виклику або комюніке між медіа-вузлами M_{i,j} у хмарі SDNP, а також IP-пакета 1506С, направленого на стільниковий телефон 32, що містить інформацію про маршрутизацію запиту або комюніке зі шлюзового вузла SDNP на клієнтський вузол C_{1,1}, що являє собою стільниковий телефон 32. За рахунок такого рішення медіа-вузли повинні лише направляти вхідні корисні навантаження відповідно до інструкцій, які вони отримують із сигнальних серверів, тобто механізм повністю протилежний механізму маршрутизації, використовуваному при обміні інформацією ОТТ із використанням мережі Інтернет.

Наприклад, як уже зазначалося, Інтернет-маршрутизаторами володіють різні Інтернет-провайтери та телефонні компанії, які не завжди зацікавлені в клієнтах, які хочуть забезпечити маршрутизацію своїх пакетів з найменшою затримкою відправлення та пересилання. Фактично, на відміну від обміну інформацією SDNP відповідно до даного винаходу, Інтернет-маршрутизатори не можуть навіть відрізнити пакети даних, які містять аудіо- або відеоматеріали в режимі реального часу від "сміттєвої" пошти. При обміні інформацією в режимі реального часу затримка є критичним чинником. Затримки тривалістю кілька сотень мілісекунд помітно впливають на якість обслуговування, а затримки величиною понад 500 мілісекунд стають неприйнятними для підтримки синхронного голосового обміну інформацією. З цієї та багатьох інших причин параметри мережі SDNP, що характеризують її функціонування в режимі реального часу та описані в цьому документі, безупинно відстежують затримки відправлення та обирають кращий маршрут для кожного пакета даних у режимі реального часу в момент його відправлення.

Як показано на рис. 131, запитуване пересилання з "IP Addr ТВ", тобто планшета 33, на "IP Addr CP", тобто стільниковий телефон 32 може бути здійснене різними можливими маршрутами. Кожна затримка при переході від одного вузла до іншого, відстежена та внесена в таблицю 1416, істотно відрізняється від інших. Більше того, результатом маршрутизації виклику через менше число медіа-серверів не обов'язково є найменший час затримки при обміні інформацією. Наприклад, пересилання (маршрутизація) виклику з клієнтського вузла C_{2,1} на

медіа-вузол $M_{a, f}$ і далі на клієнтський вузол $C_{2,1}$ характеризується сумарним часом затримки $55+60=115$ мс, тоді як пересилання виклику з медіа-вузла $M_{a, f}$ через медіа-вузол $M_{a, d}$ замість прямого пересилання на клієнтський вузол $C_{1,1}$, показане у вигляді затіненого шляху та показане на рис. 132A, характеризується затримкою $55+15+15=85$ мс, що на 20 % швидше, незважаючи на те, що воно здійснюється через додатковий вузол. При динамічній маршрутизації SDNP сигнальний сервер S завжди передбачає найкращу комбінацію шляхів, що означає не тільки забезпечення мінімальної затримки передачі, але й розбивку даних на частини та відсилання вмісту з пересиланням по решітчастому маршруту з метою підвищення безпеки. Відповідно до зображення, інший шлях, який характеризується короткою затримкою, показаний у вигляді затіненої доріжки, що пролягає через медіа-вузол $M_{a, h}$, наведений на рис. 132B, характеризується сумарною затримкою пересилання $25+20+15+15+15=105$ с, що також є більш сприятливою, незважаючи на велику кількість вчинених із цією метою переходів.

Інша важлива функція командних сигналів і сигналів керування полягає в керуванні перетворенням пакета. Ця функція відіграє ключову роль у змішуванні, поділі та зміні маршрутизації пакетів SDNP у хмарі. На рис. 132C проілюстровано один із способів того, яким чином сигнальний вузол S може обмінюватися інформацією з медіа-сервером (у цьому випадку це хостинговий медіа-вузол $M_{a, q}$) для керування входом і виходом пакетів з конкретного вузла. Маючи повну інформацію про всі відповідні налаштування безпеки 1504 для вхідного пакета SDNP і кадру його корисного навантаження, користуючись пакетом командних сигналів і сигналів керування 1496C, сигнальний вузол S видає інструкції медіа-вузлу $M_{a, q}$ про те, яким чином обробляти вхідний пакет SDNP 1497A для формування вихідного пакета даних 1497B. Відповідно до зображення, після вилучення корисного навантаження 1511A, що складається з множинних кадрів, медіа-вузол $M_{a, q}$, виконуючи DUM-операцію 1210, здійснює дешифрування та дескремблювання всіх кадрів з корисного навантаження 1511A, а також усіх кадрів з корисних навантажень інших вхідних пакетів (не показано), виходячи з інформації про стан 920, початкові значення 929 і ключі дешифрування 1030, використовувані при формуванні кожного з них, з подальшим змішуванням усіх вхідних полів і формуванням довгого пакета, які в цьому випадку представлені всіма незалежними кадрами в сукупності у вигляді кадрів даних 1512 і окремо у вигляді кадрів даних 1, 6, 9, 12, 23 і 31, відповідно.

Далі ці дані передаються в сортувальник "поштових індексів" SDNP з метою сортування кадрів у групи, кожна з яких характеризується загальним місцем призначення на наступному переході в хмарі SDNP; усе це відбувається відповідно до інформації про маршрутизацію, що міститься в пакеті SDNP 1506B, пересланому раніше сигнальним вузлом S для кожного кадру пакета SDNP у відповідь на інформацію, що міститься у виклику, визначену пакетом командних сигналів і сигналів керування 1495A. Далі операція SSE 1213 здійснює розділення кадрів на групи, що характеризуються однаковими місцями призначення, з використанням поточної інформації про стан 920, оновлені початкові значення 929, а також нові ключі дешифрування 1030. Місцем призначення одного з таких корисних навантажень, а саме навантаження 1511B, що містить кадри даних 1, 9 і 23, є медіа-вузол $M_{a, j}$, тоді як попереднє корисне навантаження 1511A складається з кадрів даних 1, 6 і 9. Тому, відповідно до інструкцій, виданих сигнальним вузлом S , медіа-вузол $M_{a, q}$ видалив дані кадру 6 і замінив їх даними кадру 23 з метою формування корисного навантаження 1511B, що збирається у відправний пакет SDNP 1487B, і відсилається на медіа-вузол $M_{a, j}$...

Користуючись семирівневою моделлю взаємодії відкритих систем (OSI), з'єднання SDNP, показане на рис. 133A, являє собою безпечний тунель між шлюзами 1522, що підтримує безпечний наскрізний обмін інформацією 1529 між відповідними додатками SDNP 1335, встановленими тільки на двох клієнтах (у цьому випадку це планшет 33 і стільниковий телефон 32). У різних варіантах реалізації даного винаходу фізичні рівні та рівні передачі даних 1525, як правило, не передбачають спеціальних рішень для реалізації операції SDNP. Разом з тим, мережевий рівень 3 функціонує зовсім по-іншому, ніж мережа Інтернет, оскільки SDNP керує маршрутизацією кожного з переходів у хмарі SDNP щодо безпеки, мінімізації затримки пересилання, а також щоб забезпечити найкращу якість обслуговування. Рівень пересилання 4, використовуючи протокол TCP для керування та поліпшену версію протоколу UDP для пересилання даних у режимі реального часу, забезпечує контекстуальне пересилання, змінюючи свої методи та пріоритети, виходячи з певної інформації про те, що являють собою пакет SDNP, корисне навантаження або кадр і яка його пріоритетність. Сесійний рівень 5 також є унікальним при проведенні операції SDNP, при цьому інформація про команди керування, передані або через пакети командних сигналів і сигналів керування, переданих по каналу пересилання медіа-файлів або по каналу пересилання сигналів, визначає порядок керування кожною сесією, включаючи маршрутизацію, якість, умови доставки та пріоритетність.

При обміні інформацією SDNP презентаційний рівень 6 виконує шифрування та скремблювання на кожному етапі пересилання по мережі незалежно від шифрування, яке проводить сам клієнт.

На Прикладному рівні 7 обмін інформацією SDNP також унікальний, оскільки будь-який додаток, сумісний з SDNP, повинен бути здатний до змішування та відновлення фрагментованих даних, а також знати, що робити, якщо частина розбитого на фрагменти корисного навантаження не прибуває в місце призначення, тобто повторно виконувати контекстуальне пересилання.

Усі описані вище безпека та робочі параметри розглядуваної SDNP мережі досягаються без використання шифрування клієнтом і управління закритим ключем. Якщо клієнтський додаток також зашифрований, наприклад, власною службою безпеки компанії, тунелювання за типом VPN комбінується з розділенням даних на фрагменти з формуванням нового типу безпечного обміну інформацією – фрагментованих даних, які піддаються тунелюванню, що являє собою гібрид презентаційного 6 і прикладного рівнів 7, показаний на рис. 133B.

Одним із унікальних аспектів обміну інформацією SDNP відповідно до даного винаходу є приклад "пересилання декількох маршрутів", представлений на рис. 134. Оскільки мережа SDNP побудована за принципом решітчастого перенесення фрагментованих даних, при дворазовому або триразовому пересиланні фрагментованих полів даних по решітчастій мережі перевантаження не виникає. Концептуально, для досягнення найменшої можливої затримки при пересиланні без шкоди безпеці, корисне навантаження ділиться на субпакети та перетворюється на два додаткові кадри. Замість відсилання одного кадру одним маршрутом та другого кадру іншим маршрутом, при пересиланні декількох маршрутами численні копії кожного кадру пересилаються різними маршрутами, причому першим використовується той, який надійшов раніше. Копії, що надійшли пізніше, просто відкидаються. Наприклад, відповідно до зображення, кадр 91 відсилається двома маршрутами, а саме маршрутами 1540 і 1541, тоді як кадр 92 також пересилається декількома маршрутами, а саме маршрутами 1541 і 1543. Використовується та комбінація шляхів, які забезпечують доставку корисного навантаження одного з кадрів 91 і одного з кадрів 92 першими.

Формула винаходу. Наведений вище опис ілюструє численні переваги, пов'язані з робочими параметрами, затримками при пересиланні, якістю, безпекою та конфіденційністю, які досягаються при обміні даними відповідно до даного винаходу. У таблиці, наведеній на рис. 135, представлено порівняння описаної динамічної мережі та протоколу (SDNP) з Отт-носіями, віртуальними приватними мережами (VPN), а також мережами із з'єднанням рівноправних вузлів (RTP). Як показує таблиця, усі конкуруючі та раніше розроблені методи обміну інформацією базуються на одночасній передачі одним маршрутом, при цьому при захисті вмісту інформації, якою обмінюються, покладаються лише на шифрування. При шифруванні в сторонніх мережах VPN усі існуючі методи обміну інформацією роблять відкритою інформацію про адреси джерела та призначення сторін, які обмінюються інформацією, створюючи можливість для фішингу, перехоплення, а також профілювання, тобто схильності до кіберзлочинів. При усіх цих методах безпека є постійною, залишаючись незмінною у міру переміщення пакета по мережі. Оскільки жоден з раніше розроблених методів не забезпечує керування маршрутизацією при обміні інформацією, вони не дають можливості визначити, чи мало місце втручання в обмін інформацією; крім того, вони не можуть керувати затримкою передачі та параметрами роботи мережі в режимі реального часу. Крім того, мережі OTT і RTP не гарантують, що широкосмуговий маршрутизатор матиме хоча б можливість підтримати виклик, що призводить до постійної зміни якості голосового сигналу та безперервним зривам викликів. Нарешті, у всіх випадках, окрім описаного методу обміну інформацією SDNP і решітчастої мережі, хакер, зламавши код шифрування, може користуватися відомою йому інформацією для нанесення істотної шкоди до моменту виявлення порушення безпеки та з цієї причини зможе прочитати або почути весь вміст особистого або персонального обміну інформацією.

В описаній мережі SDNP навіть у тому випадку, якщо кіберзлочинець зламає код шифрування, дані в кожному з пакетів перекручені, неповні, змішані з іншими повідомленнями та невпорядковано скремблювані, тобто вміст будь-якого пакета SDNP не має жодної користі ні для кого, окрім особи, якій він адресований. До того ж, навіть якщо шифрування мережі зламують, завдання, на вирішення яких ідуть роки навіть при проведенні квантових обчислень, виникає на одну десятку частку секунди пізніше, ніж змінюється динамічне шифрування кожного пакета, що проходить усю хмару SDNP. Це означає, що потенційний хакер повинен починати все спочатку кожні 100 мс. При використанні таких динамічних методів на дешифрування п'ятихвилинної бесіди, навіть якщо вся вона міститься в одному рядку даних, пішли б сторіччя.

Крім цього, при додатковій розбивці даних на фрагменти, динамічному скремблюванні, а також при динамічних змішуванні та зміні маршрутизації усі переваги, забезпечувані в результаті злому коду шифрування, були б зовсім ілюзорними.

Комбінація численних рівнів безпеки, реалізована захищеною динамічною мережею та протоколом, описаними у цьому документі, включаючи динамічне скремблювання, переправлення фрагментованих даних, анонімність пакетів даних, а також динамічне шифрування, істотно перевершує рівень безпеки, забезпечуваний одним лише статичним шифруванням. При обміні інформацією SDNP, описаному в цьому документі, пакети даних при одиничній бесіді, діалозі, а також інших формах обміну інформацією не переміщуються одним маршрутом, а розбиваються на незрозумілі уривки даних, що не мають змісту, які скремблюються невпорядковано та пересилаються численними маршрутами, ці дані безупинно змінюються щодо вмісту та змішуються відповідно до базових прав безпеки. Отриманий у результаті метод обміну інформацією являє собою першу "гіперзахищену" комунікаційну систему.

ФОРМУЛА ВИНАХОДУ

1. Спосіб передавання пакетів даних у захищеному режимі через хмару, пакети даних містять цифрові дані, цифрові дані містять серію сегментів даних, хмара містить мережу медіа-вузлів, медіа-вузли розміщуються на серверах, кожний з медіа-вузлів приймає пакети даних від інших медіа-вузлів у мережі та передачу пакетів даних до інших медіа-вузлів у мережі, спосіб включає: зберігання спільних секретів у першому медіа-вузлі або на сервері, пов'язаному з першим медіа-вузлом, спільні секрети містять список алгоритмів приховування; зберігання спільних секретів у другому медіа-вузлі або на сервері, пов'язаному з другим медіа-вузлом;

ініціювання першого медіа-вузла виконувати першу операцію приховування на пакеті даних згідно з одним або більше алгоритмами приховування в списку алгоритмів приховування для приховування щонайменше частини цифрових даних у пакеті даних, один або більше алгоритмів приховування використовуються першим медіа-вузлом при виконанні першої маскувальної операції, вибраної з переліку алгоритмів приховування згідно з динамічним станом, динамічний стан включає параметр, що змінюється;

ініціювання першого медіа-вузла передавати пакет даних, змішаний пакет даних включає в себе пакет даних або компонент субпакета пакета даних до другого медіа-вузла;

передачу цифрового значення, що представляє динамічний стан, який використовується при виборі одного або більше алгоритмів приховування, що використовуються першим медіа-вузлом, при виконанні першої операції приховування з пакетом даних до другого медіа-вузла або сервера, пов'язаного з другим медіа-вузлом;

ініціювання другим медіа-вузлом або сервером, пов'язаним з другим медіа-вузлом використання цифрового значення, що представляє динамічний стан, для ідентифікації одного або більше алгоритмів приховування, що використовуються першим медіа-вузлом при виконанні першої операції приховування пакета даних;

ініціювання другим медіа-вузлом виконання зворотної першої операції приховування, щоб відтворити пакет даних у формі, в якій пакет даних існував перед тим, як перший медіа-вузол виконав першу операцію приховування з пакетом даних, використовуючи один або більше алгоритмів приховування, що використовуються першим медіа-вузлом при виконанні першої операції приховування пакета даних.

2. Спосіб за п. 1, у якому спільні секрети містять щонайменше одне з наступного:

генератор початкового стану для генерування початкового стану, початковий стан містить цифрове значення, що представляє динамічний стан;

генератор прихованого числа для генерування прихованого числа з динамічного стану або початкового стану;

інформаційну зону; й

алгоритм перетасовування процесів.

3. Спосіб за п. 1, у якому динамічний стан включає в себе час.

4. Спосіб за п. 1, у якому динамічний стан включає в себе один або більше з наступного:

номер медіа-вузла;

ідентифікацію мережі;

GPS-локалізацію;

число, що генерується шляхом збільшення випадкового числа кожного разу, коли пакет перетинає медіа-вузол у мережі; й

алгоритм для вибору алгоритму приховування, оснований на параметричному значенні, отриманому на основі даних, що містяться у пакеті даних.

5. Спосіб за п. 1, який включає в себе використання цифрового значення, що представляє динамічний стан, як вхідну змінну при виконанні щонайменше одного з алгоритмів приховування.

6. Спосіб за п. 1, у якому перша операція приховування включає в себе щонайменше один метод, вибраний з групи, що складається з:

скремблювання пакета даних шляхом зміни порядку щонайменше деяких сегментів даних у пакеті даних відповідно до алгоритму скремблювання;

шифрування пакета даних шляхом шифрування щонайменше деяких даних у пакеті даних відповідно до алгоритму шифрування;

розділення пакета даних на щонайменше два субпакети відповідно до алгоритму розділення;

перемішування пакета даних шляхом об'єднання пакета даних зі щонайменше одним іншим пакетом даних відповідно до алгоритму перемішування для утворення перемішаного пакета даних; і

додавання небажаних даних до та/або видалення небажаних даних з пакета даних відповідно до щонайменше одного алгоритму небажаних даних.

7. Спосіб за п. 1, у якому адреса другого медіа-вузла використовується першим медіа-вузлом для передавання пакета даних, змішаного пакета даних, що включає пакет даних, або компонента субпакета пакета даних до другого медіа-вузла, вибирається сервером, який не приймає перший медіа-вузол.

8. Спосіб за п. 1, який включає в себе ініціювання першого медіа-вузла передавати пакети даних, змішаного пакета даних, що включає пакет даних, або компонента субпакета пакета даних через щонайменше один проміжний медіа-вузол на шляху до другого медіа-вузла, в якому щонайменше один проміжний медіа-вузол не змінює цифрові дані у пакеті даних, перемішує пакет даних або компонент субпакета, за винятком того, щоб оновлювати адресу призначення для наступної транзитної ділянки пакета даних, змішаного пакета даних або компонента субпакета.

9. Спосіб за п. 8, у якому адреса щонайменше одного проміжного медіа-вузла, що використовується першим мультимедійним вузлом для передачі пакета даних, змішаного пакета даних або компонент субпакета до щонайменше одного проміжного медіа-вузла, вибирається іншим сервером, який не приймає перший медійний вузол.

10. Спосіб за п. 1, який включає в себе ініціювання першого медіа-вузла генерувати початкове число та передавати початкове число до другого медіа-вузла, початкове число містить цифрове значення, що представляє динамічний стан, який використовується при виборі одного або більше алгоритмів приховування від загальних секретів для виконання першої операції приховування.

11. Спосіб за п. 1, який включає в себе ініціювання другого медіа-вузла для виконання другої операції приховування пакета даних, друга операція приховування містить щонайменше один спосіб, вибраний з групи, яка складається з:

скремблювання пакета даних шляхом зміни порядку щонайменше деяких сегментів даних у пакеті даних відповідно до алгоритму скремблювання;

шифрування пакета даних шляхом шифрування щонайменше деяких даних у пакеті даних відповідно до алгоритму шифрування;

розділення пакета даних на щонайменше два субпакети відповідно до алгоритму розділення;

перемішування пакета даних шляхом об'єднання пакета даних зі щонайменше одним іншим пакетом даних відповідно до алгоритму перемішування для утворення перемішаного пакета даних; і

додавання небажаних даних до та/або видалення небажаних даних з пакета даних відповідно до щонайменше одного алгоритму небажаних даних,

у якому друга операція приховування вибрана відповідно до динамічного стану та відрізняється від першої операції приховування.

12. Спосіб за п. 11, у якому динамічний стан включає в себе час.

13. Спосіб за п. 11, який включає в себе використання цифрового значення, що представляє динамічний стан як вхідну змінну при виконанні щонайменше одного з алгоритмів скремблювання, шифрування, розділення, змішування та небажаних даних.

14. Спосіб за п. 1, у якому сервер, пов'язаний з першим медіа-вузлом, містить перший DMZ-сервер, а сервер, пов'язаний з другим медіа-вузлом, містить другий DMZ-сервер, в якому спільні секрети зберігаються на першому та другому DMZ-серверах, перший та другий DMZ-сервери

ізолюються від мережі таким чином, що жоден із медіа-вузлів у мережі, включаючи перший та другий медіа-вузли, не мають доступу до загальних секретів.

15. Спосіб за п. 14, який включає в себе ініціювання першого DMZ-сервера вибрати один або більше алгоритмів приховування зі спільних секретів відповідно до динамічного стану й інструктувати перший медіа-вузол виконувати першу операцію приховування з пакетом даних шляхом використання одного або більше алгоритмів приховування.

16. Спосіб за п. 15, який включає в себе:

ініціювання першого DMZ-сервера генерувати початковий стан, початковий стан містить цифрове значення, що представляє динамічний стан, який використовується першим DMZ-сервером для вибору одного або більше алгоритмів приховування зі спільних секретів; й ініціювання початкового стану для доставки на другий DMZ-сервер.

17. Спосіб за п. 16, у якому ініціювання початкового стану для доставки на другий DMZ-сервер включає ініціювання першого DMZ-сервера передавати початковий стан на перший медіа-вузол, ініціювання першого медіа-вузла передавати початковий стан до другого медіа-вузла й ініціювання другого медіа-вузла передавати початковий стан на другий DMZ-сервер.

18. Спосіб за п. 16, у якому ініціювання початкового стану для доставки на другий DMZ-сервер включає ініціювання першого DMZ-сервера передавати початковий стан на сигнальний сервер й ініціювання сигнального сервера передавати початковий стан на другий DMZ-сервер.

19. Спосіб за п. 16, який включає в себе ініціювання другого DMZ-сервера використовувати початковий стан для ідентифікації одного або більше алгоритмів приховування, використовуючи перший медіа-вузол, при виконанні першої операції приховування з пакетом даних й інструктування другого медіа-вузла виконувати інверсію першої операції приховування на пакеті даних.

20. Спосіб за п. 19, у якому ініціювання другого DMZ-сервера використовувати початковий стан для ідентифікації одного або більше алгоритмів приховування, використовуючи перший медіа-вузол, при виконанні першої операції приховування з пакетом даних, який включає ініціювання другого DMZ-сервера використовувати початковий стан для генерування прихованого номера та використання прихованого номера для ідентифікації одного або більше алгоритмів приховування, що використовуються першим медіа-вузлом при виконанні першої операції приховування на пакеті даних, прихований номер й алгоритм використовують для генерування прихованого числа, що є частиною спільних секретів, і не доступне для будь-якого медіа-вузла у мережі.

21. Спосіб за п. 14, який включає в себе ініціювання другого медіа-вузла виконувати другу операцію приховування на пакеті даних, друга операція приховування містить щонайменше один спосіб, вибраний з групи, що складається з:

скремблювання пакета даних шляхом зміни порядку щонайменше деяких сегментів даних у пакеті даних відповідно до алгоритму скремблювання;

шифрування пакета даних шляхом шифрування щонайменше деяких даних у пакеті даних відповідно до алгоритму шифрування;

розділення пакета даних на щонайменше два субпакети відповідно до алгоритму розділення;

перемішування пакета даних шляхом об'єднання пакета даних зі щонайменше одним іншим пакетом даних відповідно до алгоритму перемішування для утворення перемішаного пакета даних; і

додавання небажаних даних до та/або видалення небажаних даних з пакета даних відповідно до щонайменше одного алгоритму небажаних даних,

у якому друга операція приховування вибирається відповідно до динамічного стану та відрізняється від першої операції приховування.

22. Спосіб за п. 21, у якому ініціювання другого медіа-вузла виконувати другу операцію приховування на пакеті даних включає ініціювання другого DMZ-сервера вибрати один або більше з алгоритмів скремблювання, шифрування, розділення, перемішування та додавання небажаних даних зі спільних секретів відповідно до динамічного стану й інструктування другого медіа-вузла виконувати другу операцію приховування на пакеті даних шляхом використання одного або більше другого алгоритмів приховування.

23. Спосіб за п. 22, у якому динамічний стан, що використовується другим DMZ-сервером, при виконанні другої операції приховування на пакеті даних включає в себе час.

24. Спосіб за п. 1, у якому перший та другий медіа-вузли знаходяться у першій зоні хмари та в якому хмара містить другу зону, друга зона містить множину медіа-вузлів, спосіб включає:

зберігання другого набору спільних секретів у медіа-вузлах в другій зоні або серверах, пов'язаних з медіа-вузлами в другій зоні, другий набір спільних секретів включає другий перелік

алгоритмів приховування, другий перелік алгоритмів приховування відрізняється від переліку алгоритмів приховування в спільних секретах; і

використання другого набору спільних секретів для вибору алгоритмів приховування, що призначені медіа-вузлами в другій зоні для виконання операцій приховування на пакетах даних, як пакети даних, що проходять через медіа-вузли в другій зоні.

25. Спосіб за п. 24, у якому хмара містить міст медіа-вузла, який з'єднує першу та другу зони, міст медіа-вузла реалізує інверсію операцій приховування на пакетах даних, що надходять з медіа-вузлів у першій зоні відповідно до спільних секретів і реалізує операції приховування на пакетах даних, що призначені для медіа-вузлів у другій зоні відповідно до другого набору спільних секретів.

26. Спосіб за п. 1, у якому хмара містить шлюзовий вузол, шлюзовий вузол з'єднаний з пристроєм клієнта за допомогою з'єднання "остання миля", спосіб включає зберігання спільних секретів і другого набору спільних секретів у шлюзовому вузлі або на сервері, пов'язаному зі шлюзовим вузлом і зберігаючим другий набір спільних секретів на пристрої клієнта, другий набір спільних секретів містить другий перелік алгоритмів приховування, другий перелік алгоритмів приховування відрізняється від переліку алгоритмів приховування в спільних секретах і містить множину алгоритмів, вибраних з групи, що складається з:

алгоритмів скремблювання;

алгоритмів шифрування;

алгоритмів розділення;

алгоритмів перемішування; й

алгоритмів додавання та/або видалення небажаних даних.

27. Спосіб за п. 26, який включає в себе:

ініціювання пристрою клієнта виконувати другу операцію приховування на другому пакеті даних відповідно до одного або більше алгоритмів у другому переліку алгоритмів приховування, один або більше алгоритмів використовуються пристроєм клієнта в реалізації другої операції приховування, вибраної відповідно до динамічного стану;

ініціювання пристрою клієнта до передавання другого пакета даних, змішаного пакета даних, що включає другий пакет даних, або компонента субпакета другого пакета даних до шлюзового вузла; й

ініціювання пристрою клієнта до передавання шлюзового вузла або до сервера, пов'язаного зі шлюзовим вузлом цифрового значення, що представляє динамічний стан, який використовується пристроєм клієнта при реалізації другої операції приховування до другого пакета даних.

28. Спосіб за п. 27, який включає в себе ініціювання шлюзового вузла до реалізації інверсії другої операції приховування так, щоб відтворити другий пакет даних у формі так, щоб другий пакет даних існував до того, як пристрій клієнта виконав другу операцію приховування на другому пакеті даних, використовуючи один або більше алгоритмів приховування, що використовуються пристроєм клієнта в реалізації другої операції приховування до другого пакета даних.

29. Спосіб за п. 28, у якому сервер, пов'язаний зі шлюзовим вузлом, включає шлюзовий DMZ-сервер, спосіб включає:

зберігання спільних секретів і другого набору спільних секретів на шлюзовому DMZ-сервері, шлюзовий DMZ-сервер ізолюється від мережі таким чином, що жоден із медіа-вузлів у мережі, включаючи шлюзовий вузол, і перший та другий медіа-вузли, не мають доступу до спільних секретів або другого набору спільних секретів; й

ініціювання пристрою клієнта до генерування початкового стану й ініціювання початкового стану на доставку до шлюзового DMZ-сервера, початковий стан включає цифрове значення, що представляє динамічний стан, який використовується пристроєм клієнта в реалізації другої операції приховування на другому пакеті даних.

30. Спосіб за п. 29, який включає в себе ініціювання шлюзового DMZ-сервера до використання початкового стану для ідентифікації одного або більше алгоритмів з другого переліку алгоритмів приховування, що використовуються пристроєм клієнта в реалізації другої операції приховування на другому пакеті даних й інструктуванні шлюзового вузла для реалізації інверсії другої операції приховування на другому пакеті даних, використовуючи один або більше алгоритмів з другого переліку алгоритмів приховування.

31. Спосіб за п. 30, який включає в себе:

ініціювання шлюзового DMZ-сервера до вибору щонайменше одного алгоритму приховування зі спільних секретів відповідно до динамічного стану й інструктування шлюзового вузла до реалізації третьої операції приховування на другому пакеті даних, третя операція приховування

відрізняється від інших, першої та другої операцій приховування, й ініціювання шлюзового вузла до відправлення другого пакета даних, змішаного пакета даних, що включає другий пакет даних, або компонента субпакета другого пакета даних на третьому медіа-вузлі у мережі.

32. Спосіб за п. 1, який включає в себе періодичну зміну спільних секретів шляхом зміни алгоритмів приховування у переліку алгоритмів приховування, порядок алгоритмів приховування у переліку алгоритмів приховування, або цифрове значення ідентифікує алгоритми приховування.

33. Спосіб за п. 1, який включає в себе визначення маршруту пакета даних через щонайменше один проміжний медіа-вузол між першим і другим медіа-вузлами.

34. Спосіб за п. 33, який включає в себе визначення маршруту пакета даних через множину проміжних медіа-вузлів між першим і другим медіа-вузлами, і рескремблювання та/або дешифрування пакета даних у щонайменше деяких проміжних медіа-вузлах, у якому алгоритм скремблювання й/або алгоритм шифрування, що використовуються для скремблювання та/або шифрування пакета даних на кожному з проміжних медіа-вузлів, у яких пакет даних рескремблюється та/або дешифрується, відрізняються від алгоритму скремблювання й/або алгоритму шифрування, що використовуються для скремблювання пакета даних в кожному іншому проміжному медіа-вузлі, в якому пакет даних рескремблюється та/або дешифрується.

35. Спосіб за п. 1, у якому перша операція приховування включає розділення пакета даних на щонайменше два субпакети, щонайменше два субпакети включають перший субпакет і другий субпакет, спосіб включає:

визначення маршруту першого субпакета через першу серію проміжних медіа-вузлів між першим медіа-вузлом і другим медіа-вузлом;

визначення маршруту другого субпакета через другу серію проміжних медіа-вузлів між першим медіа-вузлом і другим медіа-вузлом; і

перемішування першого та другого субпакетів у другому медіа-вузлі.

36. Спосіб за п. 35, у якому перша серія проміжних медіа-вузлів не містить будь-який медіа-вузол, що міститься в другій серії проміжних медіа-вузлів.

37. Спосіб за п. 35, у якому перша серія проміжних медіа-вузлів містить щонайменше один медіа-вузол, що міститься в другій серії проміжних медіа-вузлів, і щонайменше один медіа-вузол, який не міститься в другій серії проміжних медіа-вузлів.

38. Спосіб за п. 1, у якому перша операція приховування включає перемішування пакета даних шляхом комбінування пакета даних зі щонайменше одним іншим пакетом даних для утворення змішаного пакета даних, в якому змішаний пакет даних містить щонайменше одне з наступного: два або більше заголовків;

два або більше ідентифікуючих тегів;

дві або більше адреси призначення; та

два або більше сегментів даних, у яких операція приховування була реалізована відповідно до різних значень динамічного стану, відповідно.

39. Спосіб за п. 1, у якому перший пристрій клієнта приєднується до вхідного шлюзового вузла у мережі за допомогою з'єднання "перша миля", а другий пристрій клієнта приєднується до вихідного шлюзового вузла у мережі за допомогою з'єднання "остання миля", спосіб включає:

забезпечення одного або більше сигнальних серверів;

забезпечення сигнального сервера адресою кожного з першого та другого пристроїв клієнта;

ініціювання сигнального сервера до вироблення плану мережевого маршруту, план мережевого маршруту визначає щонайменше деякі з медіа-вузлів на маршруті пакета даних через мережу в зв'язку від першого пристрою клієнта до другого пристрою клієнта, жоден із медіа-вузлів не має доступу до плану мережевого маршруту; й

ініціювання сигнального сервера до відправлення команди та контрольних пакетів до медіа-вузлів, що визначені у плані мережевого маршруту, кожна команда та контрольний пакет інформують медіа-вузол, що визначений у плані мережевого маршруту, куди відправляти вхідний пакет даних на наступній транзитній ділянці у плані мережевого маршруту.

40. Спосіб за п. 39, у якому сигнальний сервер зберігає перелік мережевого вузла, перелік мережевого вузла містить перелік медіа-вузлів і пристроїв клієнта, в якому сигнальний сервер виробляє план мережевого маршруту з урахуванням затримки поширення між медіа-вузлами у переліку мережевого вузла у порядку до зниження транзитного часу пакета даних через мережу в зв'язку від першого пристрою клієнта до другого пристрою клієнта.

41. Спосіб за п. 39, у якому сигнальний сервер зберігає перелік мережевого вузла, перелік мережевого вузла містить перелік медіа-вузлів і пристроїв клієнта, спосіб включає:

ініціювання першого пристрою клієнта до передавання на сигнальний сервер ідентифікатора другого пристрою клієнта та запит адреси другого пристрою клієнта; й

ініціювання сигнального сервера передати адресу другого пристрою клієнта на перший пристрій клієнта.

42. Спосіб за п. 39, у якому щонайменше одне з команди та контрольних пакетів інструктують медіа-вузол, що визначений у плані мережевого маршруту, розділити вхідний пакет даних на субпакети або перемішати вхідний пакет даних з іншим пакетом з утворенням змішаного пакета даних й інструктування медіа-вузла, куди відправляти кожний з субпакетів або змішаний пакет даних.

43. Спосіб за п. 39, у якому жоден із медіа-вузлів у мережі, крім вхідного шлюзового вузла, не знає адресу першого пристрою клієнта, і жоден із медіа-вузлів у мережі, крім вихідного шлюзового вузла, не знає адресу другого пристрою клієнта.

44. Спосіб за п. 39, який включає:

забезпечення іменного серверного вузла, ім'я серверного вузла містить одне або більше імен серверів і зберігає перелік мережевого вузла, перелік мережевого вузла містить перелік активних медіа-вузлів і пристроїв клієнта;

ініціювання першого пристрою клієнта для передавання на іменний серверний вузол ідентифікації другого пристрою клієнта та запиту адреси другого пристрою клієнта;

ініціювання іменного серверного вузла на передачу адреси другого пристрою клієнта на перший пристрій клієнта; й

ініціювання першого пристрою клієнта на передачу адреси другого пристрою клієнта на сигнальний сервер.

45. Спосіб за п. 1, у якому перший пристрій клієнта з'єднаний з входом шлюзового вузла у мережі за допомогою з'єднання "перша миля", а другий пристрій клієнта з'єднаний з виходом шлюзового вузла у мережі за допомогою з'єднання "остання миля", мережа містить третій медіа-вузол, третій медіа-вузол виконує функцію сервера доменних імен і сигнальну функцію, спосіб включає:

забезпечення третього медіа-вузла адресою кожного з першого та другого пристроїв клієнта;

ініціювання третього медіа-вузла до розробки плану мережевого маршруту, план мережевого маршруту визначає щонайменше деякі з медіа-вузлів на маршруті пакета даних через мережу при обміні інформацією від першого пристрою клієнта до другого пристрою клієнта, жоден із медіа-вузлів, крім третього медіа-вузла, не мають доступу до плану мережевого маршруту; й

ініціювання третього медіа-вузла до відправлення команди та контрольних пакетів до медіа-вузлів, що визначені у плані мережевого маршруту, кожна команда та контрольний пакет інформують медіа-вузол, що визначений у плані мережевого маршруту, куди відправляти вхідний пакет даних на наступній транзитній ділянці у плані мережевого маршруту.

46. Спосіб за п. 45, у якому третій медіа-вузол зберігає перелік мережевого вузла, перелік мережевого вузла містить перелік активних медіа-вузлів і пристроїв клієнта, спосіб включає:

ініціювання першого пристрою клієнта до передачі на третій медіа-вузол ідентифікації другого пристрою клієнта та запит на адресу другого пристрою клієнта; й

ініціювання третього медіа-вузла на передачу адреси другого пристрою клієнта на перший пристрій клієнта.

47. Спосіб за п. 45, у якому третій медіа-вузол містить вхідний шлюзовий вузол.

48. Спосіб за п. 1, у якому перший пристрій клієнта з'єднаний з входом шлюзового вузла у мережі за допомогою з'єднання "перша миля", а другий пристрій клієнта з'єднаний з виходом шлюзового вузла у мережі за допомогою з'єднання "остання миля", спосіб включає ініціювання першого пристрою клієнта скремблювати та/або шифрувати пакет даних і передавати секретні облікові дані на другий пристрій клієнта, секретні облікові дані дозволяють другому пристрою клієнта рескремблювати та/або дешифрувати пакет даних, щоб відтворити пакет даних як він існував до того, як пакет даних був скремблований та/або шифрований першим пристроєм клієнта, секретні облікові дані не передаються або не відомі будь-якому медіа-вузлу в мережі.

49. Спосіб за п. 48, у якому перший пристрій клієнта передає секретні облікові дані на другий пристрій клієнта через сигнальний сервер.

50. Спосіб за п. 1, у якому перший пристрій клієнта з'єднаний з входом шлюзового вузла у мережі за допомогою з'єднання "перша миля", а другий пристрій клієнта з'єднаний з виходом шлюзового вузла у мережі за допомогою з'єднання "остання миля", спосіб включає:

ініціювання першого пристрою клієнта до розділення пакета даних з утворенням множини субпакетів і створення копії субпакета;

ініціювання першого пристрою клієнта на відправлення субпакета на другий пристрій клієнта за першим маршрутом через хмару та відправлення копії субпакета на другий пристрій клієнта за другим маршрутом через хмару, другий маршрут відрізняється від першого маршруту; й

ініціювання другого пристрою клієнта поєднати будь-який субпакет і копію субпакета, що надійшов першим, разом з іншими з множини субпакетів так, щоб відтворити пакет даних.

51. Спосіб за п. 50, який включає в себе ініціювання другого пристрою клієнта відхилити будь-який з субпакета та копії субпакета, що надійшли пізніше.

52. Спосіб передавання пакетів даних в захищеному режимі від першого пристрою клієнта до другого пристрою клієнта через хмару, хмара містить мережу медіа-вузлів, медіа-вузли розміщуються на серверах, кожний з медіа-вузлів отримує пакети даних з інших медіа-вузлів у мережі та передає пакети даних до інших медіа-вузлів у мережі, перший пристрій клієнта з'єднаний з вхідним шлюзовим вузлом у мережі за допомогою з'єднання "перша миля", а другий пристрій клієнта з'єднаний з виходом шлюзового вузла у мережі за допомогою з'єднання "остання миля", спосіб включає:

забезпечення одного або більше сигнальних серверів;

забезпечення сигнального сервера адресою кожного з першого та другого пристроїв клієнта;

ініціювання сигнального сервера до розробки плану мережевого маршруту, план мережевого маршруту визначає щонайменше деякі з медіа-вузлів на маршруті пакета даних через мережу при обміні інформацією від першого пристрою клієнта до другого пристрою клієнта, жоден із медіа-вузлів не має доступу до плану мережевого маршруту; й

ініціювання сигнального сервера на відправлення команди та контрольних пакетів до медіа-вузлів, що визначені у плані мережевого маршруту, кожна команда та контрольний пакет інформують медіа-вузол, що визначений у плані мережевого маршруту, куди відправляти вхідний пакет даних на наступній транзитній ділянці у плані мережевого маршруту.

53. Спосіб за п. 52, у якому вхідний пакет даних ідентифікується тегом і команда та контрольний пакет, отриманий медіа-вузлом, інформують медіа-вузол, що визначений у плані мережевого маршруту, який тег застосувати до пакета даних перед відправленням пакета даних до наступного медіа-вузла у плані мережевого маршруту.

54. Спосіб за п. 52, у якому сигнальний сервер зберігає перелік мережевого вузла, перелік мережевого вузла містить перелік медіа-вузлів і пристроїв клієнта, спосіб включає в себе:

ініціювання першого пристрою клієнта до передавання на сигнальний сервер ідентифікації другого пристрою клієнта та запит адреси другого пристрою клієнта; й

ініціювання сигнального сервера до передачі адреси другого пристрою клієнта на перший пристрій клієнта.

55. Спосіб за п. 54, у якому перший пристрій клієнта передає на сигнальний сервер ідентифікацію другого пристрою клієнта та запит адреси другого пристрою клієнта через вхідний сигнальний сервер.

56. Спосіб за п. 52, у якому сигнальний сервер розробляє план мережевого маршруту з урахуванням затримки поширення між медіа-вузлами в мережі у порядку зниження транзитного часу пакета даних через мережу при обміні інформацією від першого пристрою клієнта до другого пристрою клієнта.

57. Спосіб за п. 52, який включає в себе автоматичне виведення медіа-вузла в офлайн, якщо завантаження на медіа-вузол при отриманні та передачі пакетів даних падає нижче заданого рівня.

58. Спосіб за п. 52, у якому перший пристрій клієнта ідентифікується за адресою мережі, відомою медіа-вузлам в мережі, але недоступною через інтернет, і за інтернет-адресою, доступною через інтернет, спосіб включає ініціювання першого пристрою клієнта зареєструватися в мережі шляхом передавання обох, мережевої адреси й інтернет-адреси, на сигнальний сервер.

59. Спосіб за п. 52, який включає в себе забезпечення резервного сигнального сервера, функція резервного сигнального сервера полягає в автоматичному прийманні завдань, що виконуються сигнальним сервером, якщо один із пристроїв клієнта або медіа-вузлів не здатен досягти сигнального сервера або якщо сигнальний сервер впав або атакований.

60. Спосіб за п. 52, який включає в себе:

забезпечення іменного серверного вузла, ім'я серверного вузла містить одне або більше імен серверів і зберігає перелік мережевого вузла, перелік мережевого вузла містить перелік активних медіа-вузлів і пристроїв клієнта;

ініціювання першого пристрою клієнта для передавання на іменний серверний вузол ідентифікації другого пристрою клієнта та запит адреси другого пристрою клієнта;

ініціювання іменного серверного вузла на передачу адреси другого пристрою клієнта на перший пристрій клієнта; й

ініціювання першого пристрою клієнта на передачу адреси другого пристрою клієнта на сигнальний сервер.

61. Спосіб за п. 60, який включає в себе:

ініціювання іменного серверного вузла для передавання на сигнальний сервер переліку медіа-вузлів, що потрібні для розробки плану мережевого маршруту; й

5 ініціювання сигнального сервера на розробку плану мережевого маршруту з використанням переліку медіа-вузлів.

62. Спосіб за п. 60, у якому перший пристрій клієнта ідентифікується мережевою адресою, відомою медіа-вузлам в мережі, але не доступною через інтернет, і доступний за інтернет-адресою через інтернет, спосіб включає ініціювання першого пристрою клієнта зареєструватися в мережі шляхом передавання обох, мережевої адреси й інтернет-адреси, на іменний сервер.

10 63. Спосіб за п. 60, який включає в себе забезпечення резервного іменного сервера, функція резервного іменного сервера полягає в автоматичному прийманні завдань, що виконуються іменним сервером, якщо один із пристроїв клієнта або медіа-вузлів не здатен досягти іменного сервера, або якщо іменний сервер впав або атакований.

15 64. Спосіб за п. 52, у якому жоден із медіа-вузлів у мережі, крім вхідного шлюзового вузла, не знає адресу першого пристрою клієнта, і жоден із медіа-вузлів у мережі, крім вихідного шлюзового вузла, не знає адресу другого пристрою клієнта.

20 65. Спосіб передавання пакетів даних в захищеному режимі від першого пристрою клієнта до другого пристрою клієнта через хмару, хмара містить мережу медіа-вузлів, медіа-вузли розміщуються на серверах, кожний з медіа-вузлів отримує пакети даних з інших медіа-вузлів у мережі та передає пакети даних до інших медіа-вузлів у мережі, перший пристрій клієнта з'єднаний з вхідним шлюзовим вузлом у мережі за допомогою з'єднання "перша миля", а другий пристрій клієнта з'єднаний з виходом шлюзового вузла у мережі за допомогою з'єднання "остання миля", мережа містить перший медіа-вузол, перший медіа-вузол реалізує функцію іменного сервера та сигнальну функцію, спосіб включає:

25 забезпечення першого медіа-вузла у мережі адресами кожного з першого пристрою клієнта та другого пристрою клієнта;

ініціювання першого медіа-вузла розробити план мережевого маршруту, план мережевого маршруту визначає щонайменше деякі з медіа-вузлів на маршруті пакета даних через мережу при обміні інформацією від першого пристрою клієнта до другого пристрою клієнта, жоден із медіа-вузлів, крім першого медіа-вузла, не має доступу до плану мережевого маршруту; й ініціювання першого медіа-вузла відправити команду та контрольні пакети до медіа-вузлів, що визначені у плані мережевого маршруту, кожна команда та контрольний пакет інформують медіа-вузол, що визначений у плані мережевого маршруту, куди відправляти вхідний пакет даних на наступній транзитній ділянці у плані мережевого маршруту.

35 66. Спосіб за п. 65, у якому вхідний пакет даних визначений тегом, і команда та контрольний пакет інформують медіа-вузол, що визначений у плані мережевого маршруту, який тег застосувати до пакета даних перед відправкою пакета даних до наступного медіа-вузла у плані мережевого маршруту.

40 67. Спосіб за п. 65, у якому перший медіа-вузол зберігає перелік мережевого вузла, перелік мережевого вузла містить перелік медіа-вузлів і пристроїв клієнта, спосіб включає:

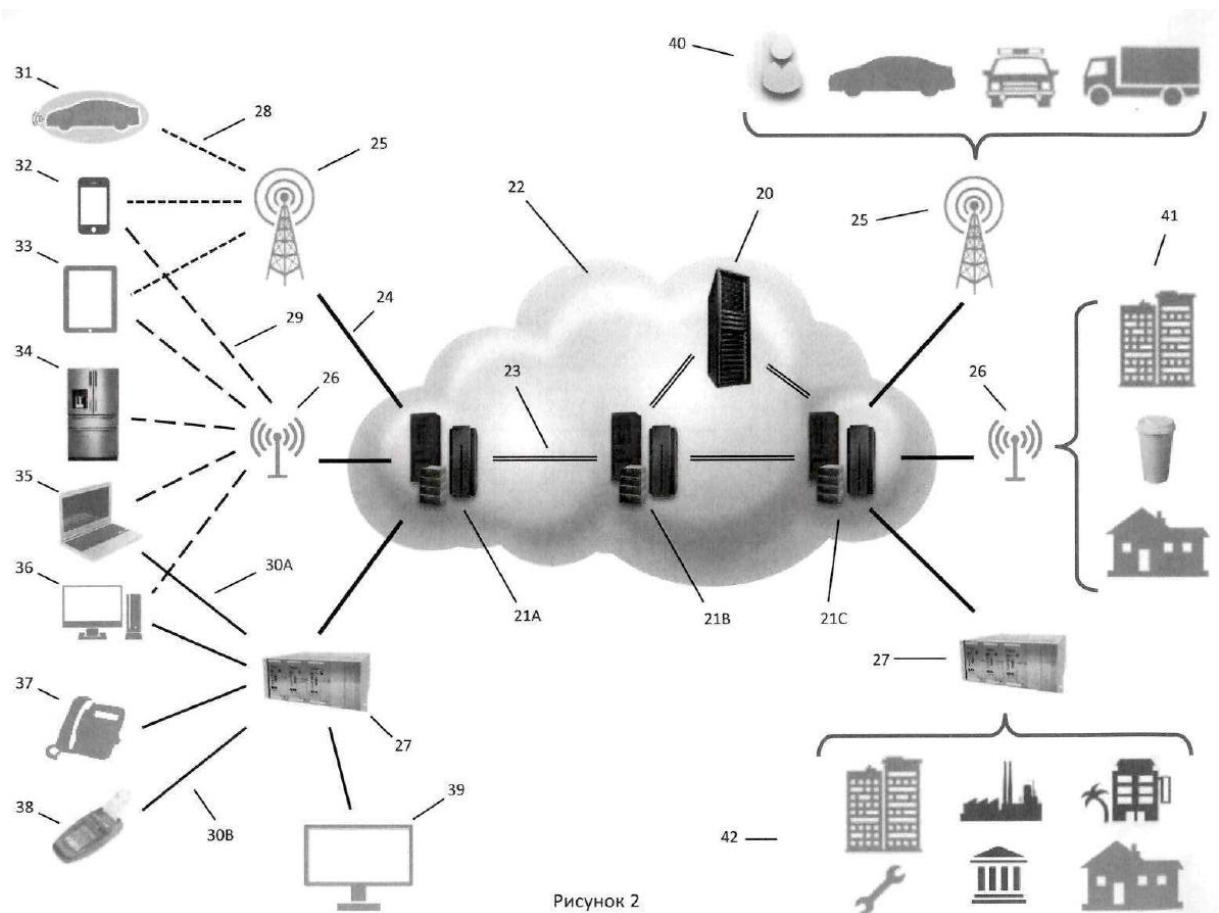
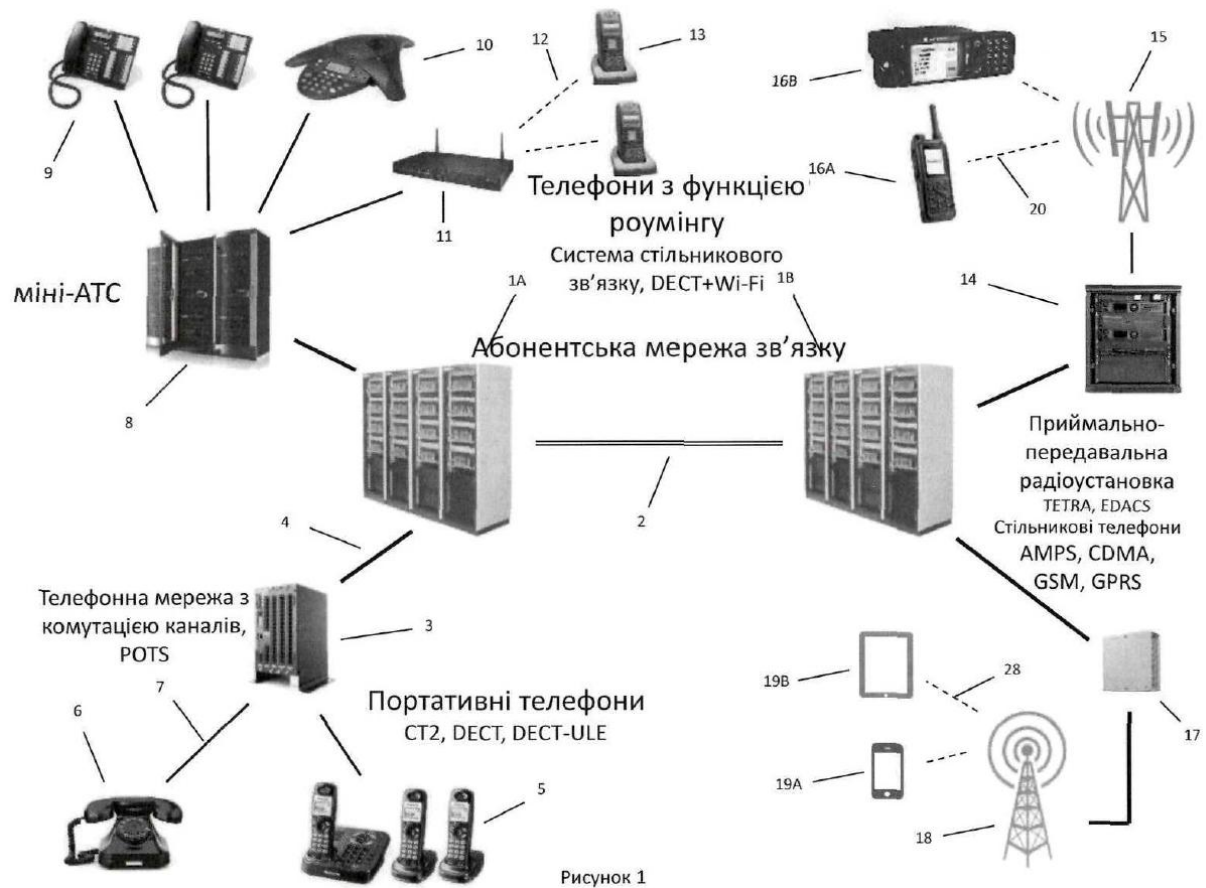
ініціювання першого пристрою клієнта для передавання на перший медіа-вузол ідентифікації другого пристрою клієнта та запит адреси другого пристрою клієнта; й

ініціювання першого медіа-вузла передати адресу другого пристрою клієнта на перший пристрій клієнта.

45 68. Спосіб за п. 65, у якому перший медіа-вузол розробляє план мережевого маршруту з урахуванням затримки поширення між медіа-вузлами в мережі у порядку зниження транзитного часу пакета даних через мережу при обміні інформацією від першого пристрою клієнта до другого пристрою клієнта.

50 69. Спосіб за п. 65, у якому жоден із медіа-вузлів у мережі, крім вхідного шлюзового вузла, не знає адресу першого пристрою клієнта, і жоден із медіа-вузлів у мережі, крім вихідного шлюзового вузла, не знає адресу другого пристрою клієнта.

70. Спосіб за п. 65, у якому перший медіа-вузол містить вхідний шлюзовий вузол.



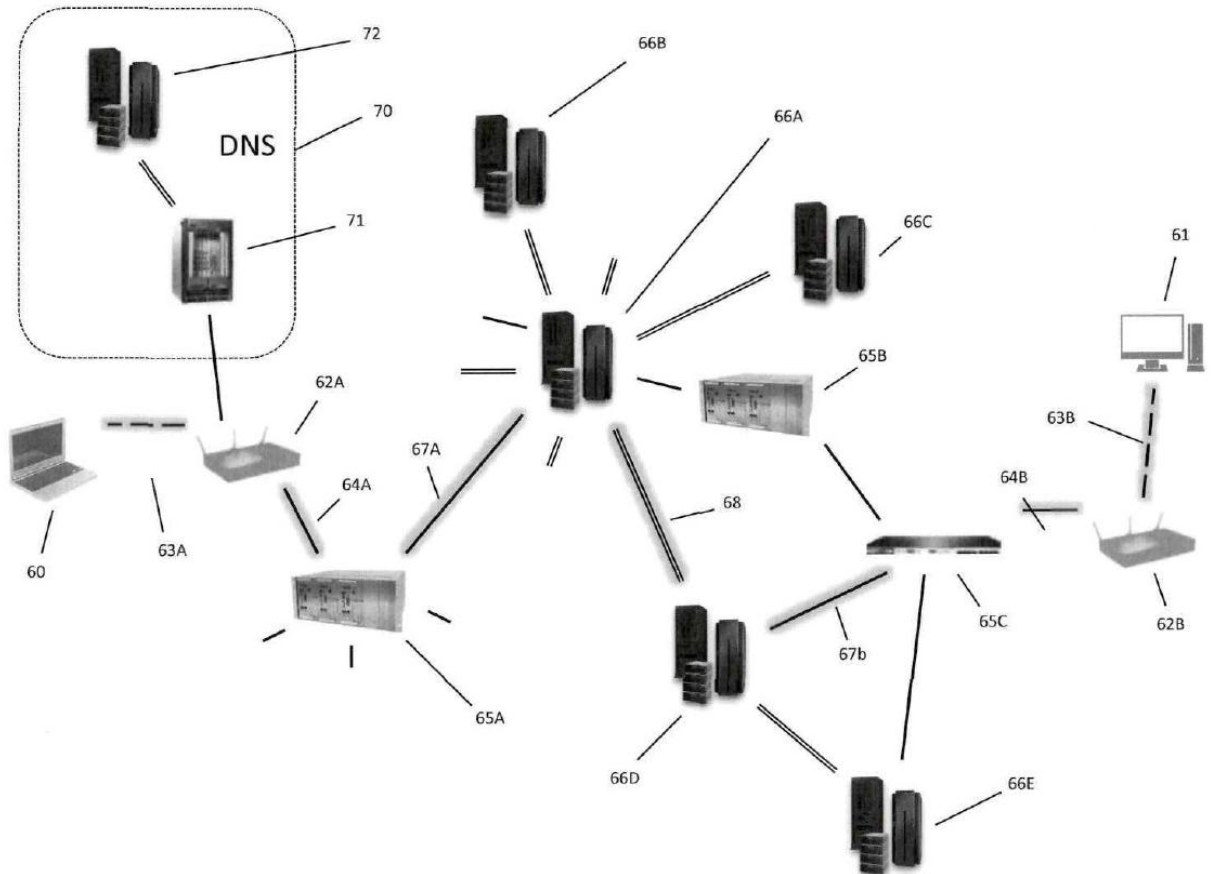


Рисунок 3



OSI-рівень	Категорія даних	Приклади
7. Прикладний рівень	Дані (верхні рівні, Прикладні рівні, корисне навантаження)	TELNET, файли (FTP, SFTP, FTAM, ZIP), e-mail (SMTP, IMAP, POP3), веб (DNS, HTTP, Safari, Firefox, Chrome і т. д.), зв'язок (SIP, IRC, NNTP, OTT), мережеве керування (SNMP, DHCP, BGP, LDAP, CMIP), драйвери (принтер, сканер, камера), резервування (NFS, оперативне, замовне)
6. Рівень представлення		текст (ASCII, EBCDIC, ZIP), графіка (PNG, JPG, GIF, BMP), аудіо та відео (MP4, WMV, MOV, AVI, MIDI), документи (PDF, DOC, PPT, HTML, XML, MIME), трансляція (RTP, RTSP, RTMP), шифрування (TLS/SSL, SSH)
5. Сеансовий рівень		SOCKS, RPC, аутентифікація, повний/напівдуплекс, симплексне з'єднання
4. Транспортний рівень	Сегменти, датаграма	UDP, TCP, DCCP, VPN (GRE, PPTP, SSTM, SSH...), LCM
3. Мережевий рівень	Пакет, датаграма	IP, ICMP, IGMP
2. Канальний рівень	Біт, кадр, VLAN (802.1Q)	802.11, 802.3, ATM, PPP, HSPA, LTE, DOCSIS, LLC
1. Фізичний рівень	Електричні, оптичні та радіосигнали (включаючи потоки двійкових сигналів)	Wi-Fi, Ethernet, послідовна/паралельна/волоконно-оптична лінія зв'язку, T1, 4G, кабель, мідний дріт, радіомовлення (FM, TV, DAB, DMB), Bluetooth, біти

Рисунок 4

OSI-рівень 1 : фізичний рівень
Висока пропускна спроможність

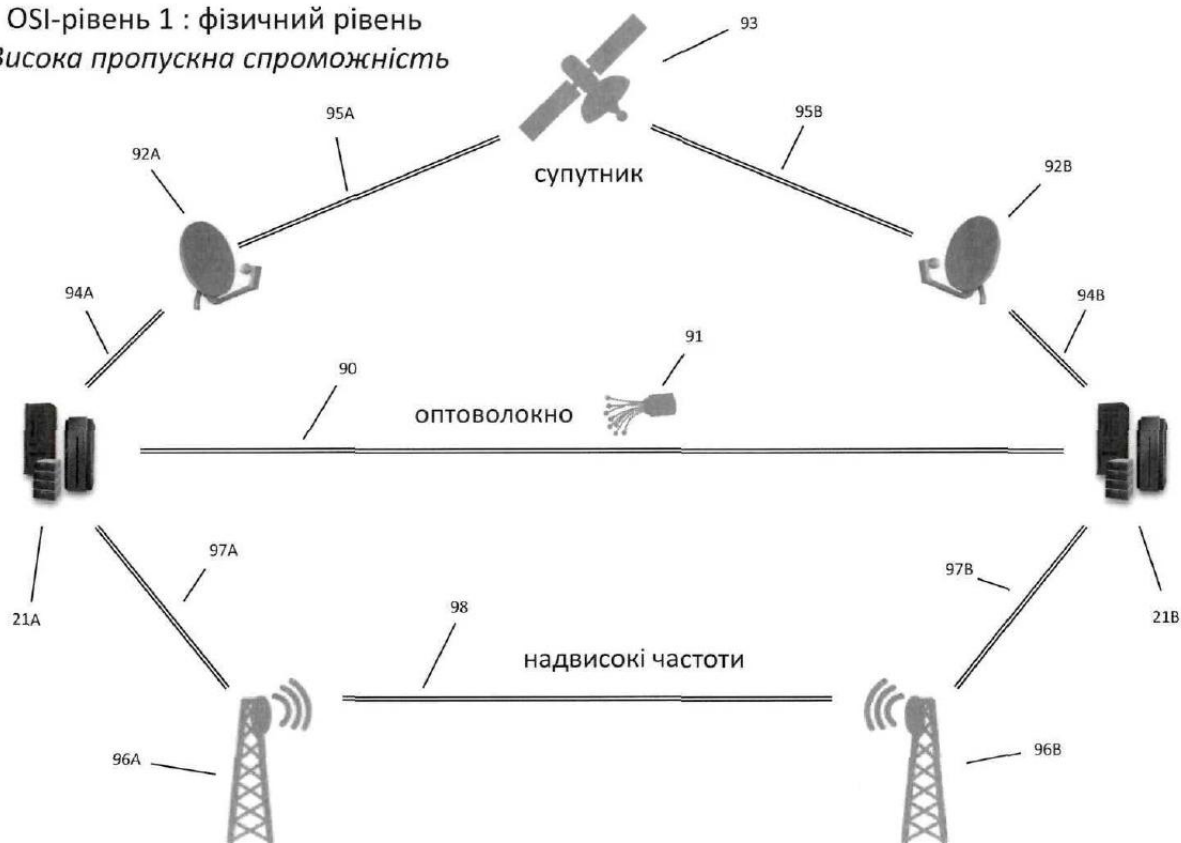
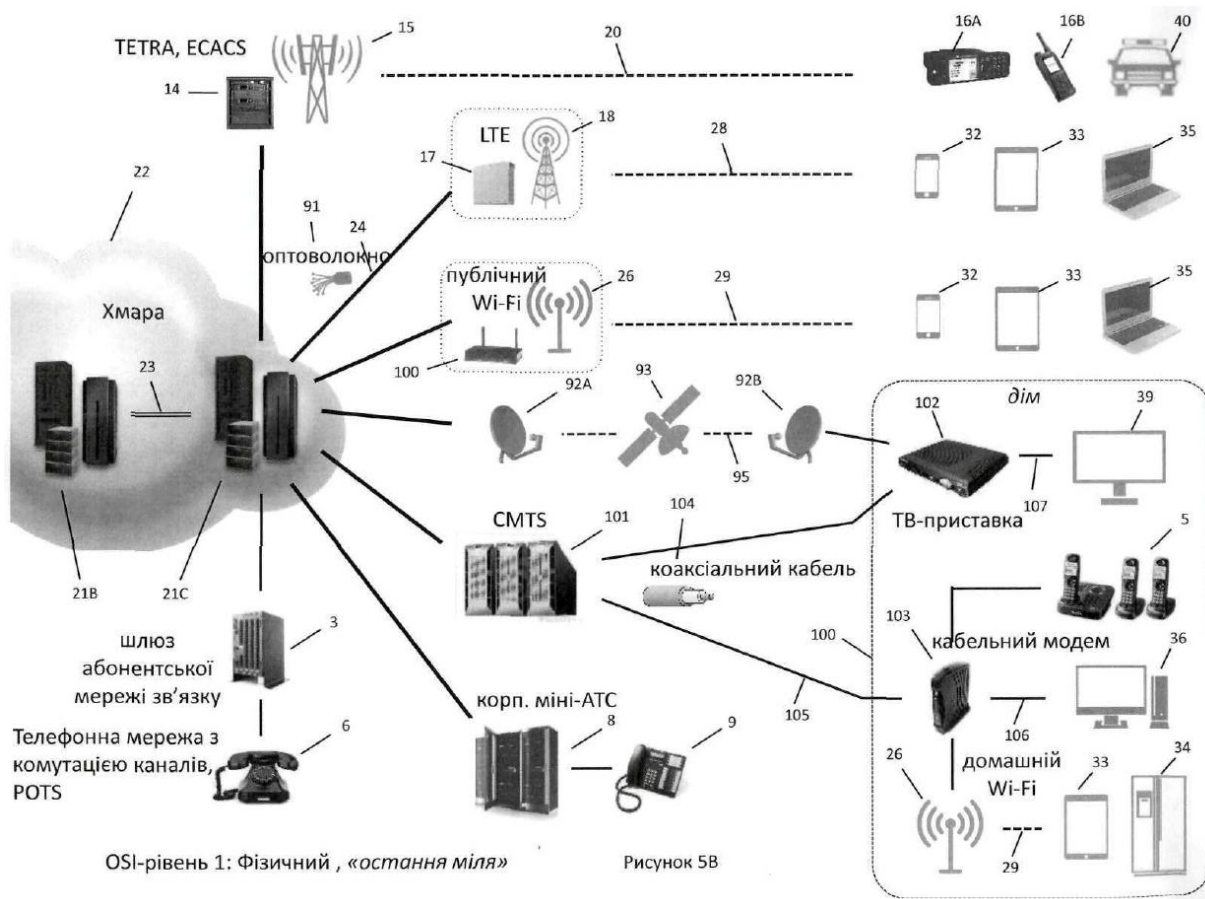


Рисунок 5А



OSI-рівень 1: Фізичний , «остання міля»

Рисунок 5В

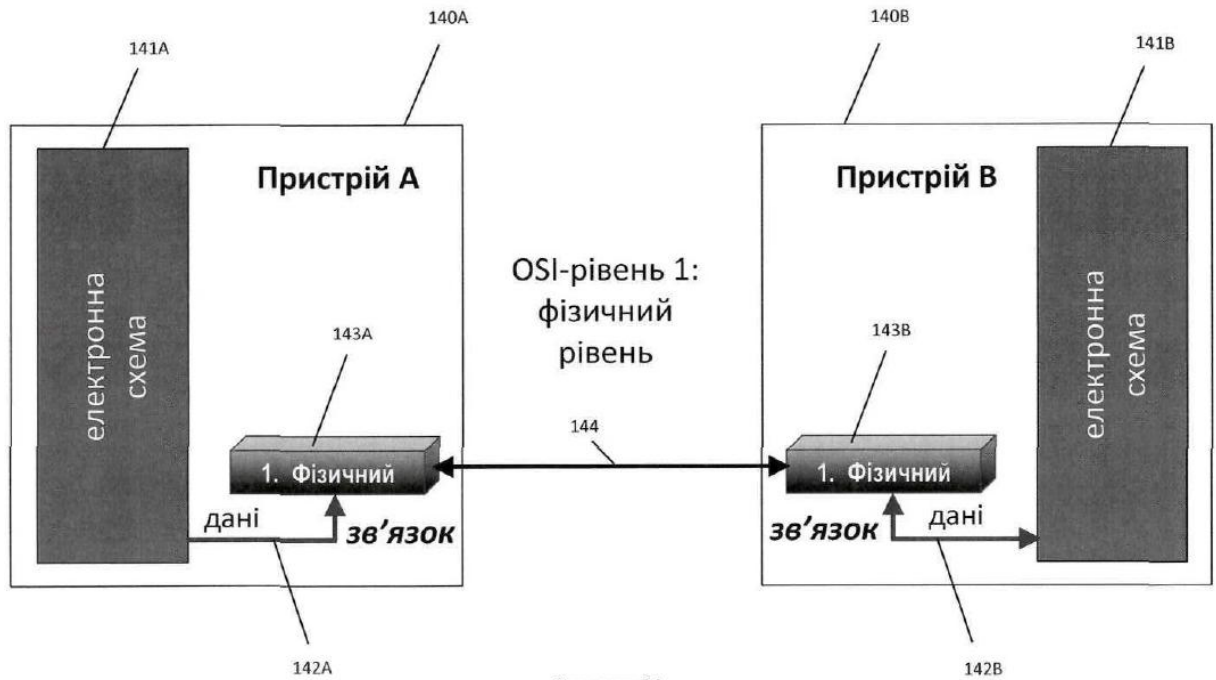


Рисунок 6А

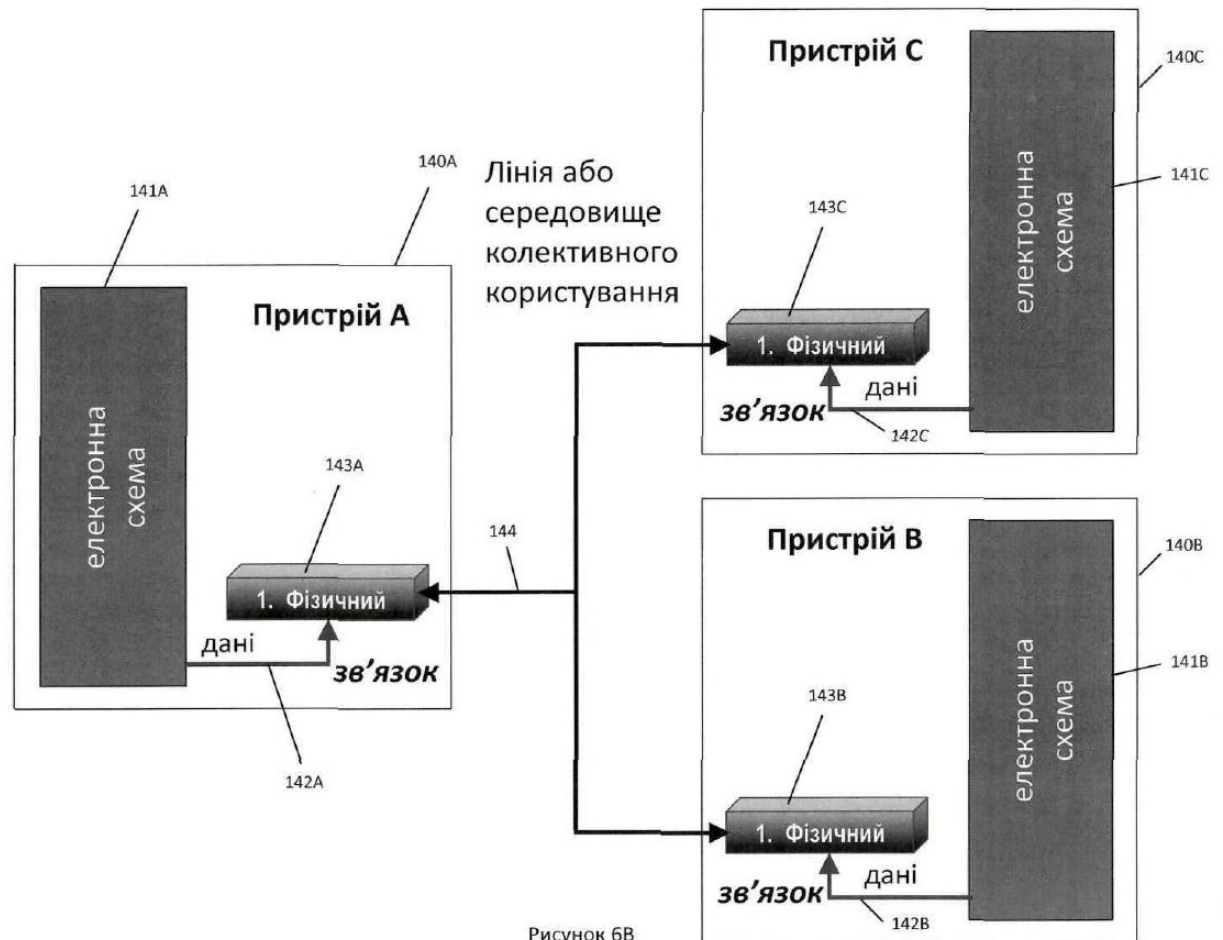


Рисунок 6В

OSI-рівень 2: Канальний рівень (архітектура шини)

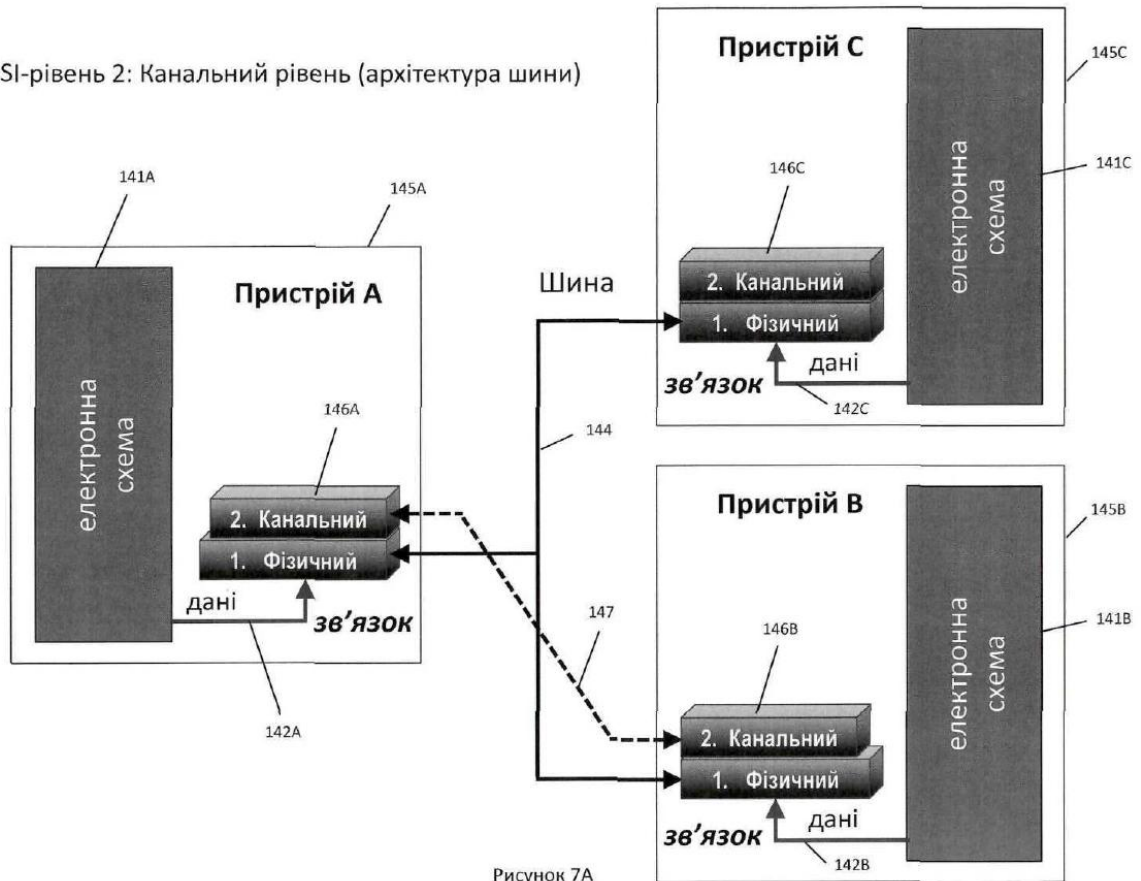


Рисунок 7А

OSI-рівень 2: Канальний рівень (хабова архітектура)

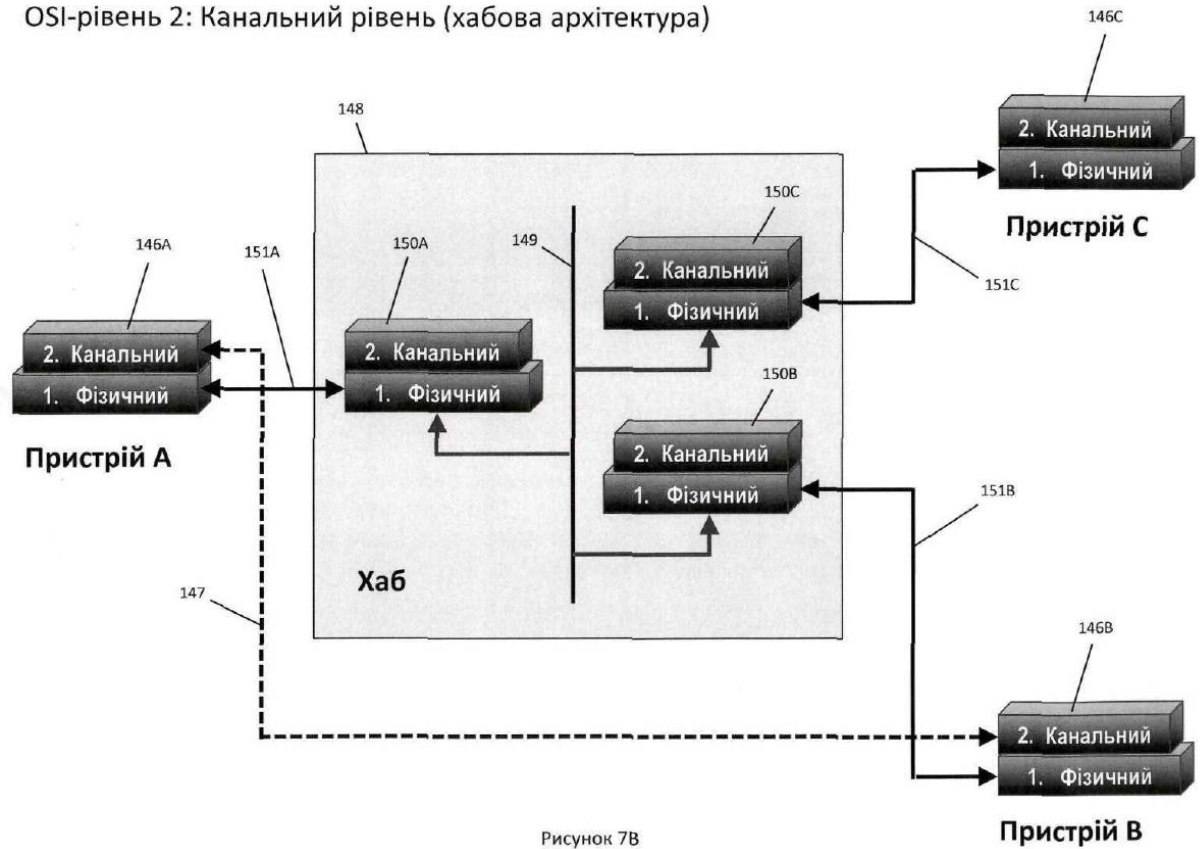


Рисунок 7В

OSI-рівень 2: Канальний (архітектура послідовного підключення)

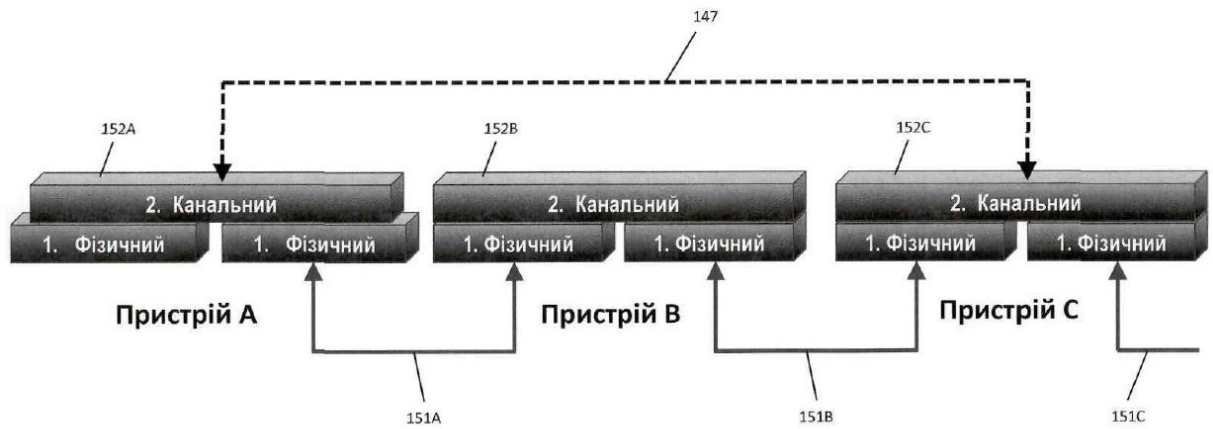


Рисунок 7С

OSI-рівень 2: Канальний (архітектура локальних мереж)

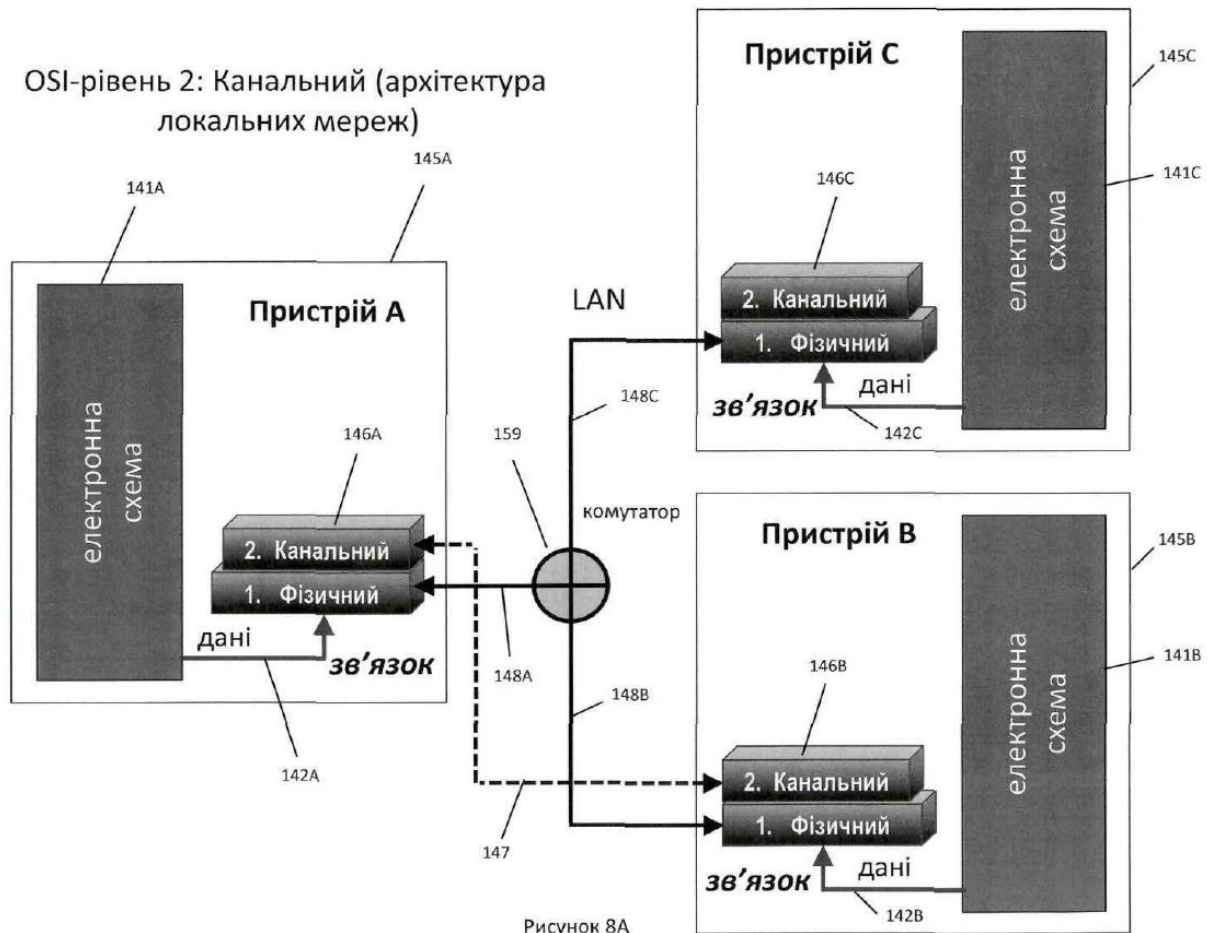


Рисунок 8А

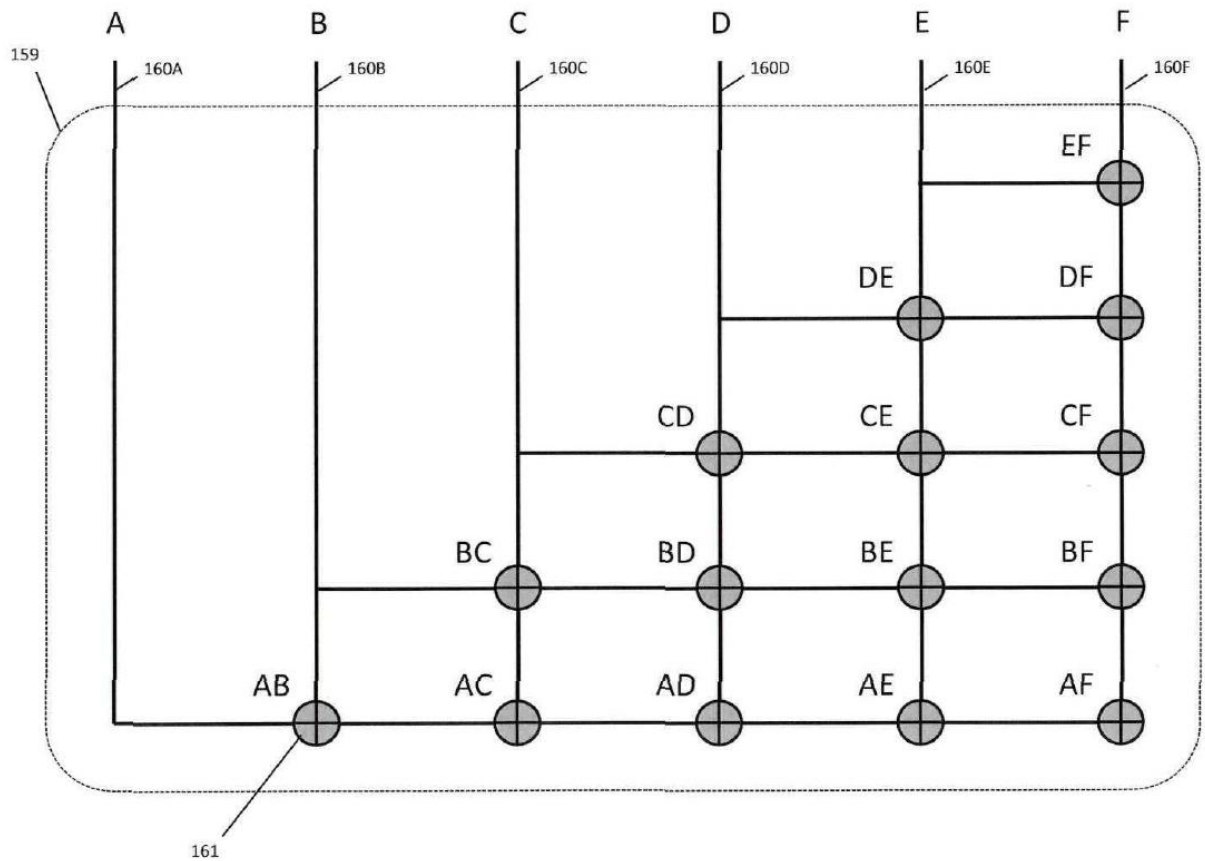


Рисунок 8В

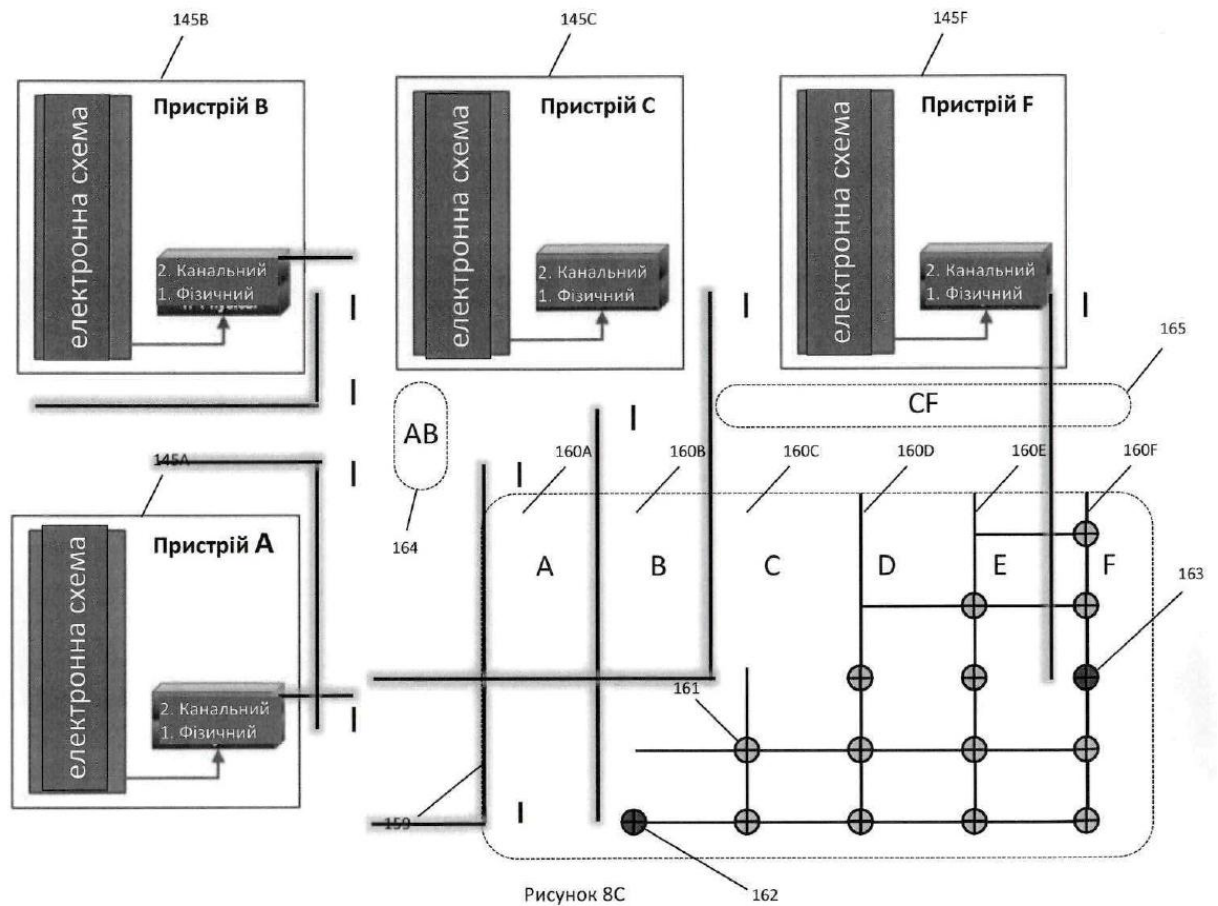
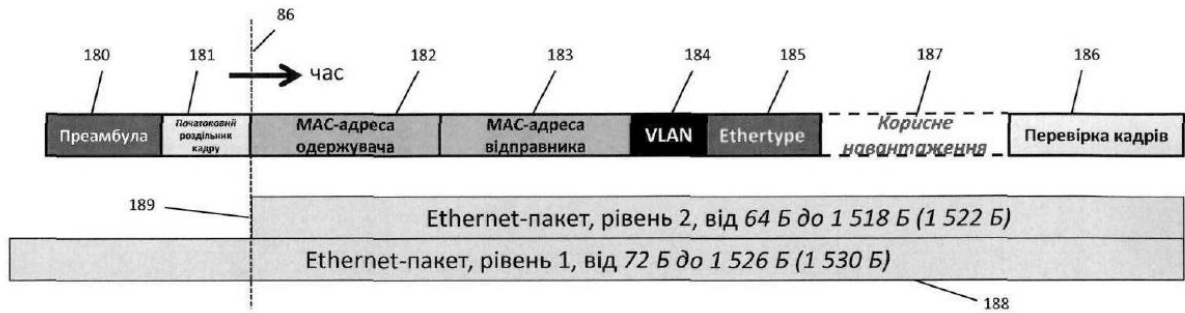


Рисунок 8С



OSI-рівні 1 і 2: канальний рівень Ethernet-пакетів (IEEE 802.3)

Рівень	Поле пакета	Розмір	Функція
1	Прембула	7 Б	Заголовок кадру даних
1	Початковий роздільник кадру	1 Б	Синхронізація апаратних пристроїв (Початковий роздільник кадру)
2	MAC-адреса одержувача	6 Б	MAC-адреса одержувача в LAN
2	MAC-адреса відправника	6 Б	MAC-адреса відправника в LAN
2	VLAN	(4 Б)	Tag 802.1Q (опціонально)
2	Ethertype	2 Б	Тип каналу (Ethernet II) або довжина (IEEE 802.3)
2-7	Корисне навантаження	42 Б – 1 500 Б	Вміст пакета (включаючи мережу, IP, прикладна інформація і дані)
2	Перевірка кадрів	4 Б	Контрольна сума цілого пакета (завершувач каналу передачі даних CRC-32)

Рисунок 9

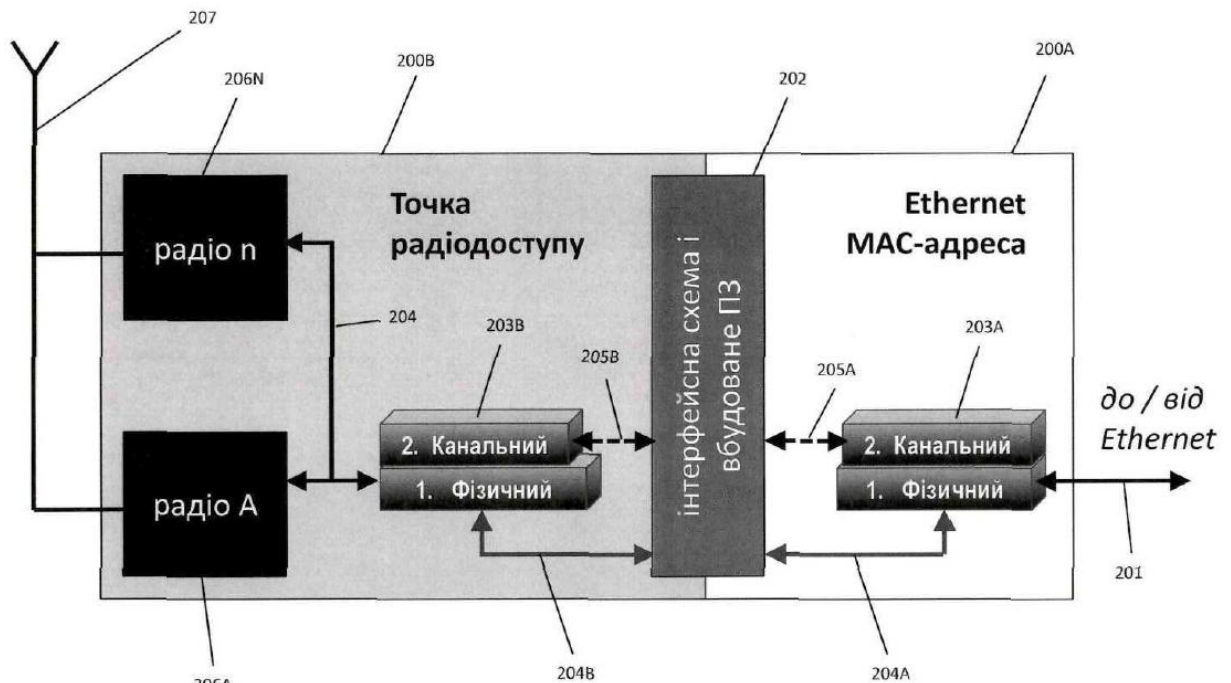
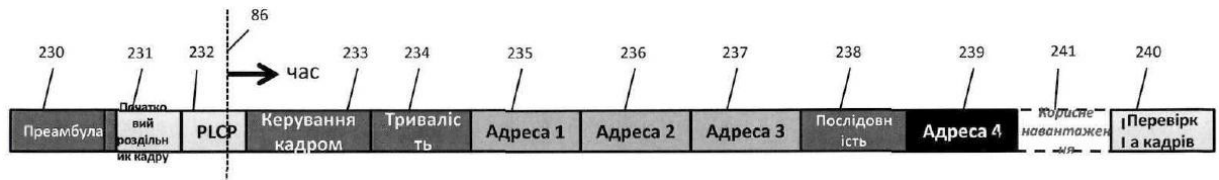


Рисунок 10



OSI-рівні 1 та 2: каналний рівень WLAN-пакетів (Wi-Fi, IEEE 802.11 і т. д.)

Рівень	Поле пакета	Розмір	Функція
1	Прееамбула	10 Б	Заголовок кадру даних
1	Початковий роздільник кадру	2 Б	Синхронізація апаратних пристроїв (Початковий роздільник кадру)
1-2	Процедура конвергенції фізичного рівня, PLCP	2 Б	Процедура визначення відповідності фізичного рівня (довжина, швидкість, контроль помилок у заголовку)
2	Керування кадрам	2 Б	Тип версії (керування, контроль, дані, резервування)
2	Тривалість та ID	2 Б	Тривалість NAV (крім режиму енергозбереження, далі ID станції)
2	Адреса 1	6 Б	Адреса приймальної станції ПБС (залежить до/від налаштування розподіленої системи)
2	Адреса 2	6 Б	Адреса передавальної станції ПБС (залежить до/від налаштування розподіленої системи)
2	Адреса 3	6 Б	Умовна адреса (залежить до/від налаштування розподіленої системи)
2	Керування послідовністю	2 Б	Номер послідовності та фрагмента, який визначає кадр пакета
2	Адреса 4	6 Б	Адреса станції-джерела в режимі бездротової розподіленої системи (БСР)
2-7	Корисне навантаження	0 – 2 312 Б	Вміст пакета (включаючи мережу, IP, прикладна інформація і дані)
2	Перевірка кадру	4 Б	Контрольна сума цілого пакета (завершувач каналу передачі даних CRC-32)

Рисунок 11

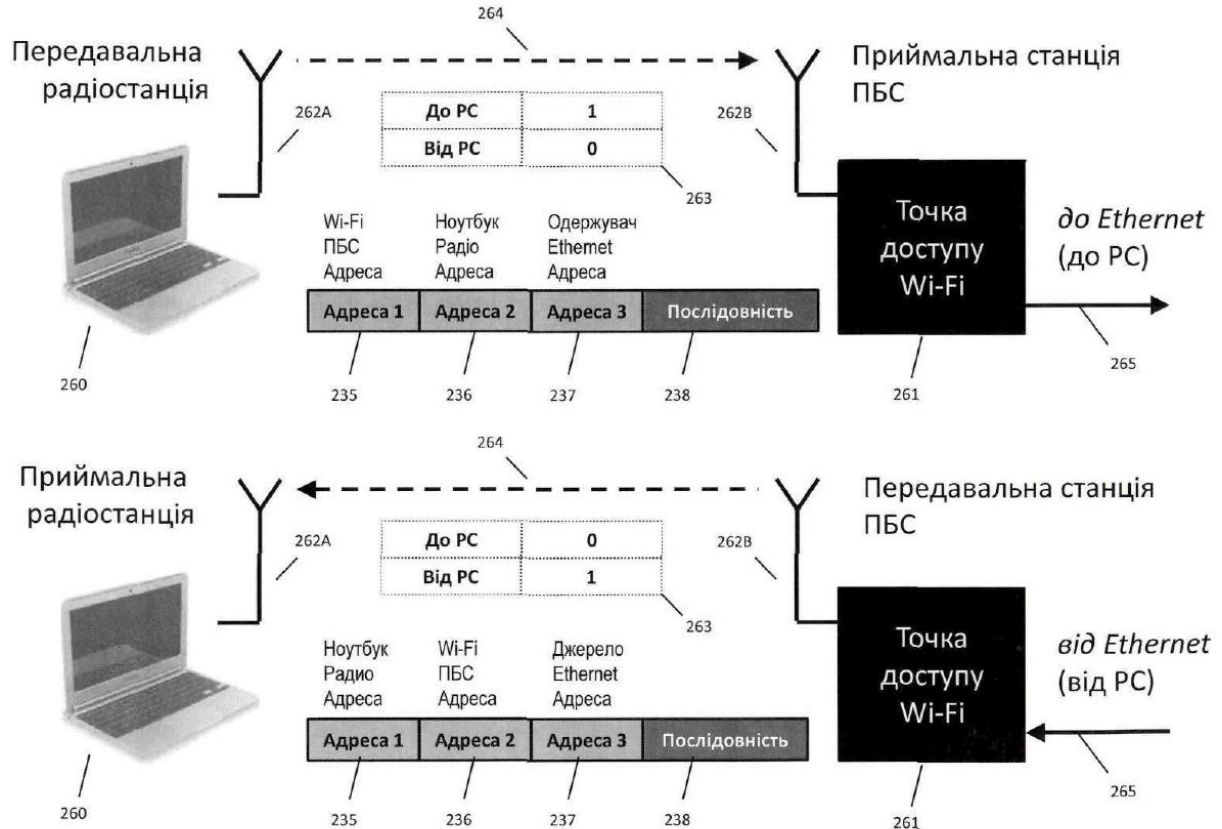


Рисунок 12A

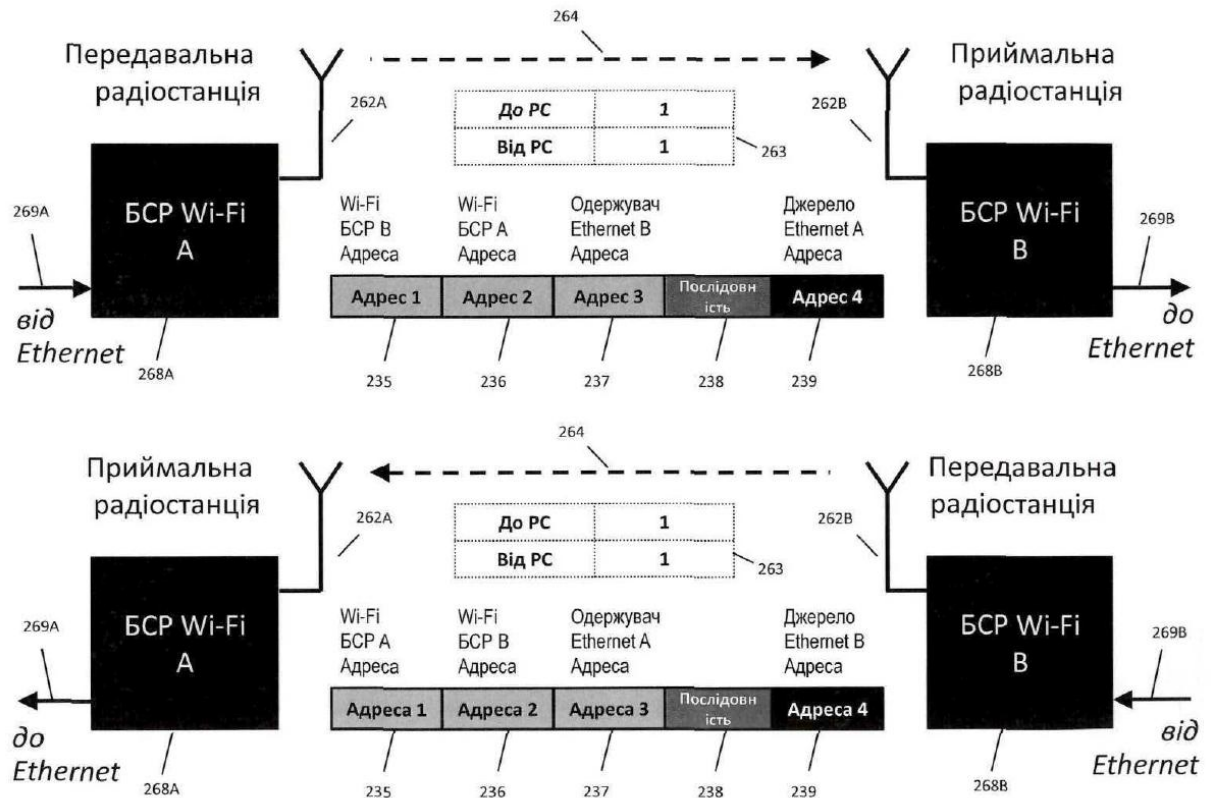


Рисунок 12В

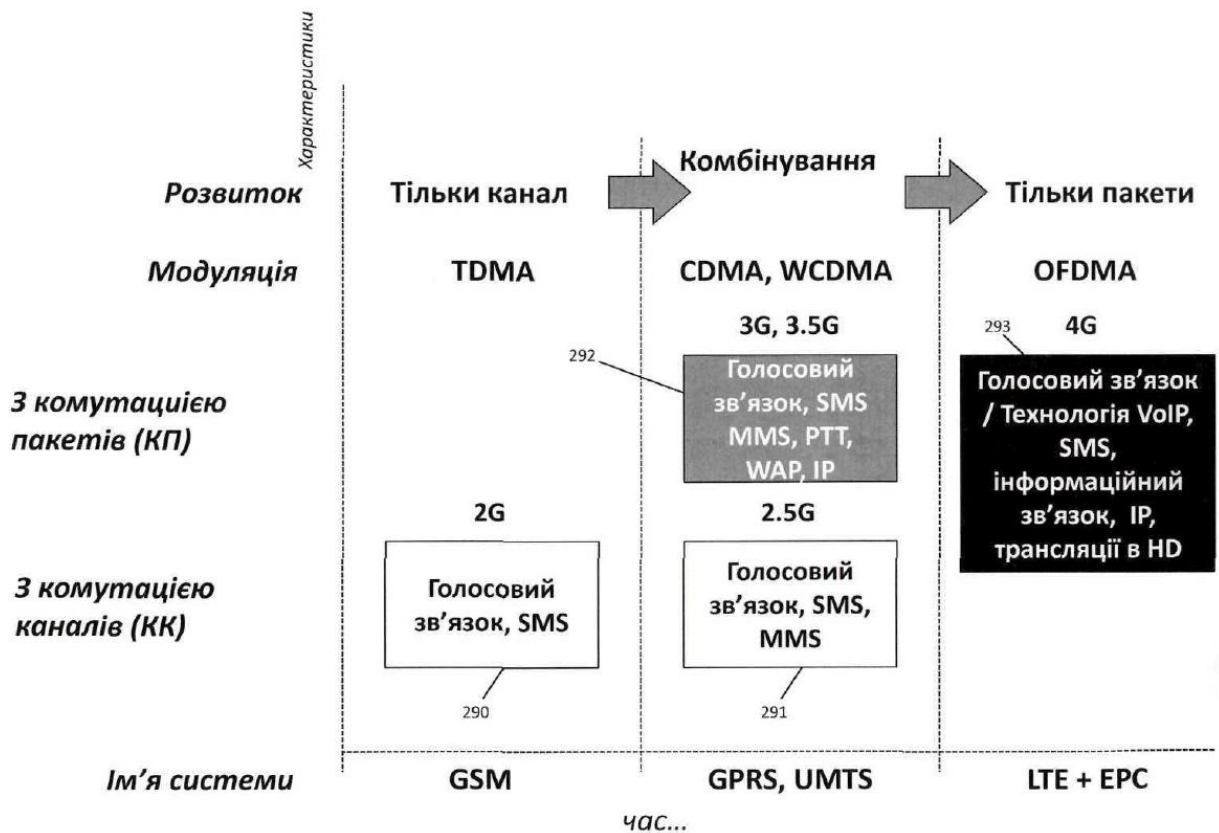


Рисунок 13

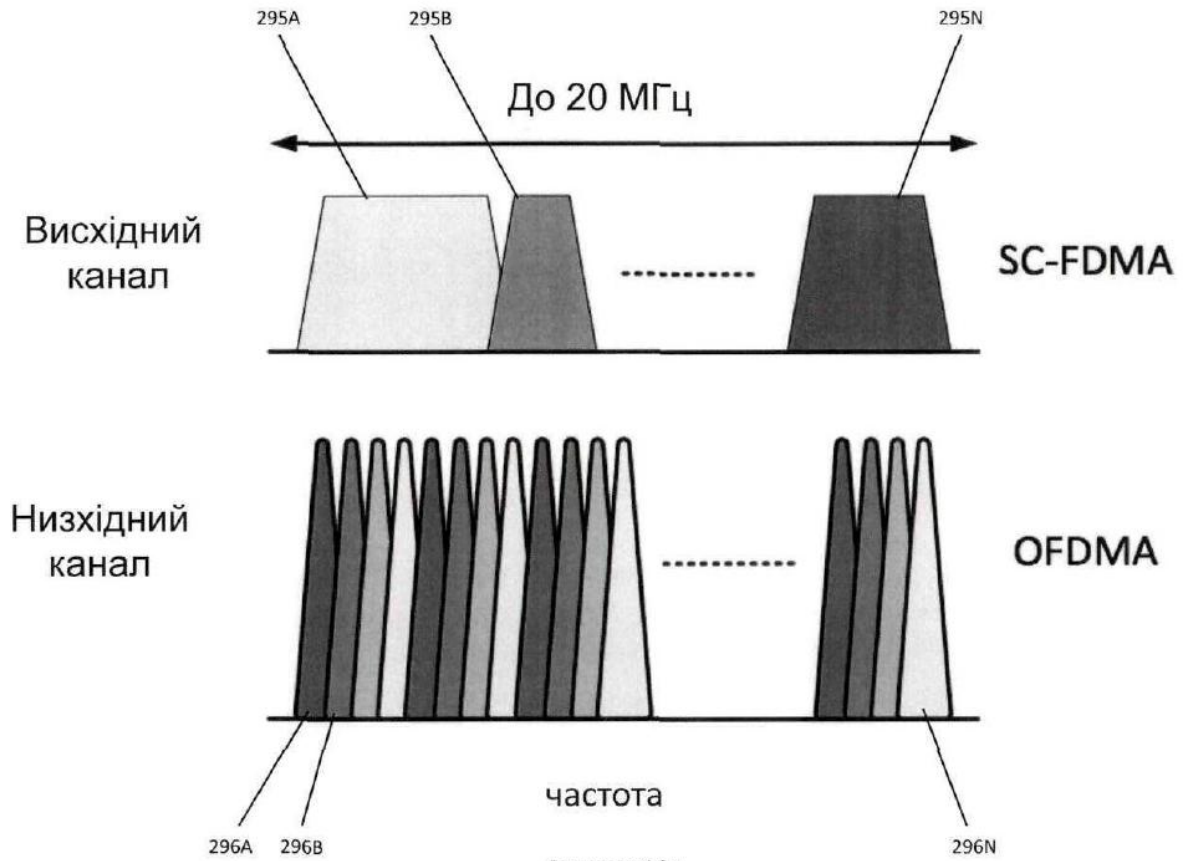


Рисунок 14А

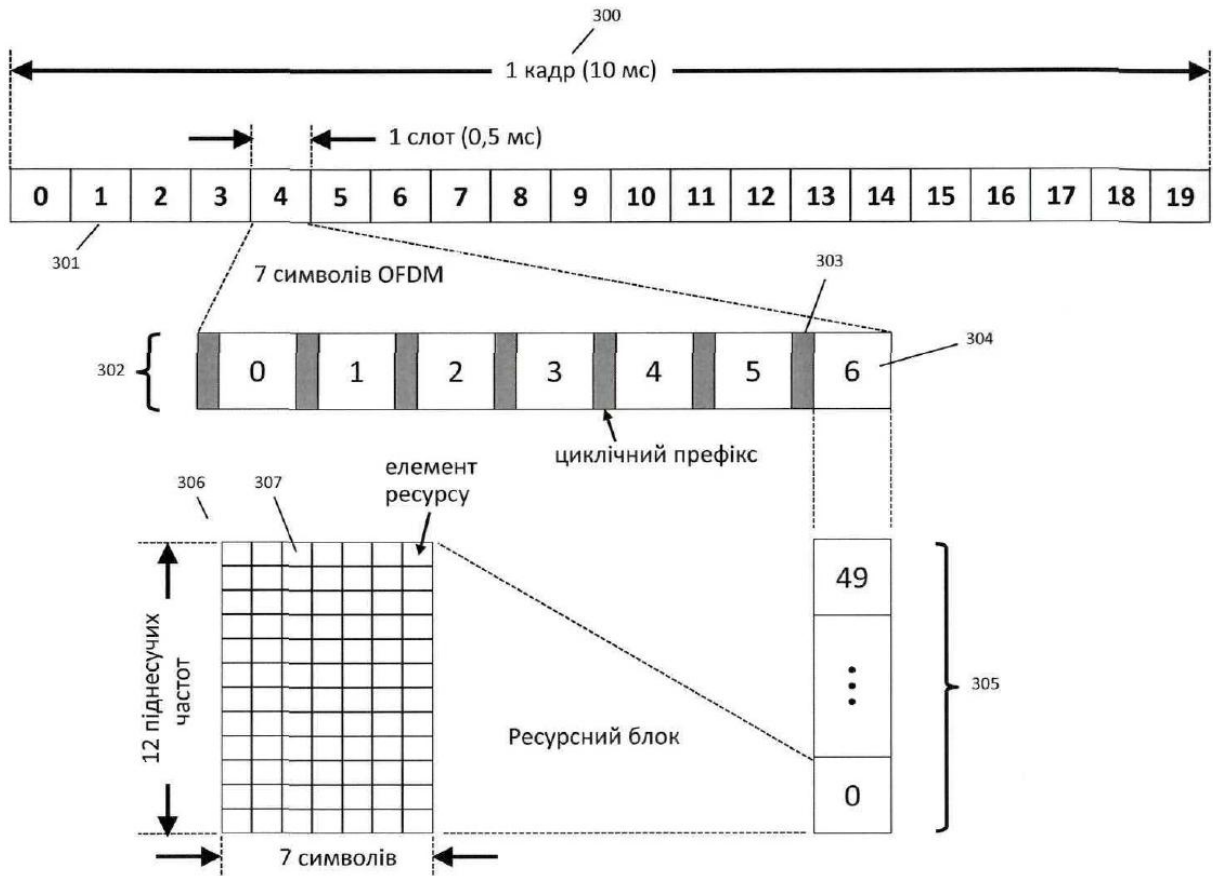


Рисунок 14В

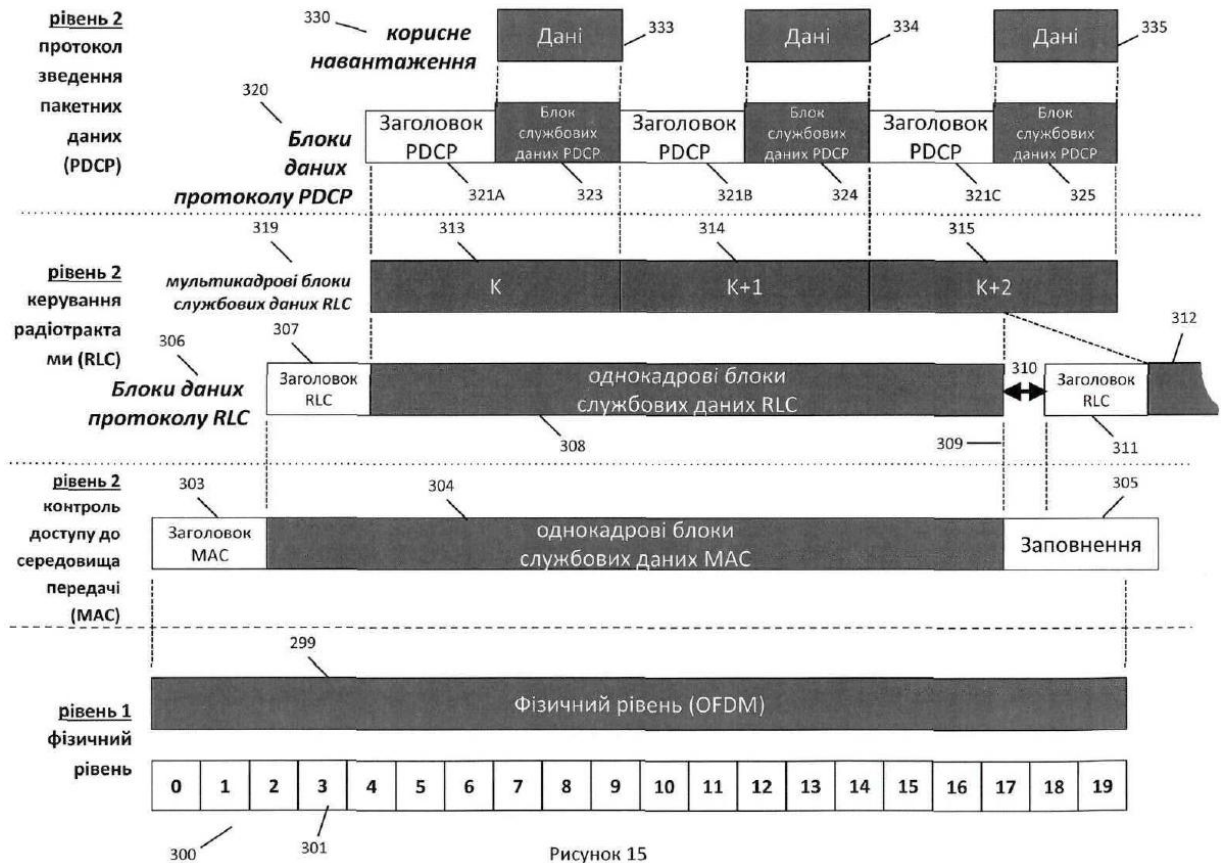


Рисунок 15

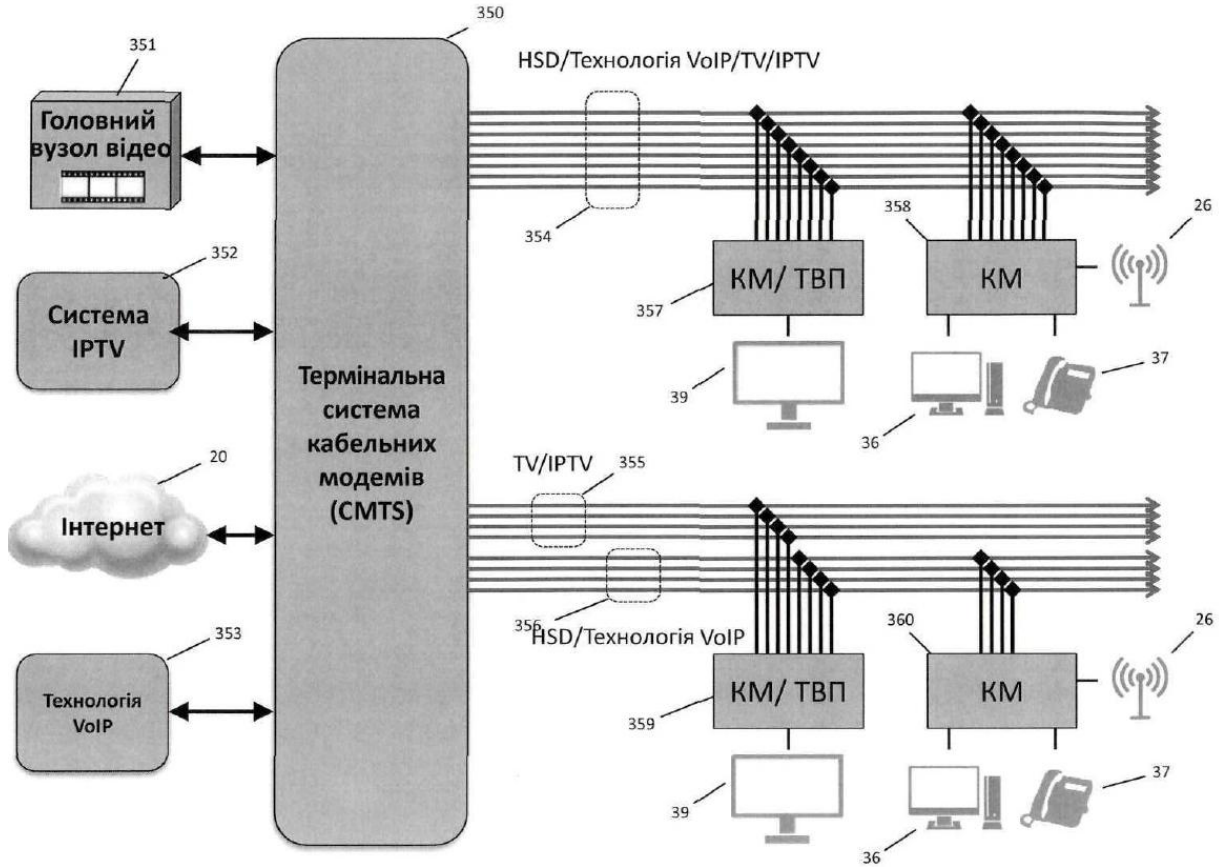


Рисунок 16

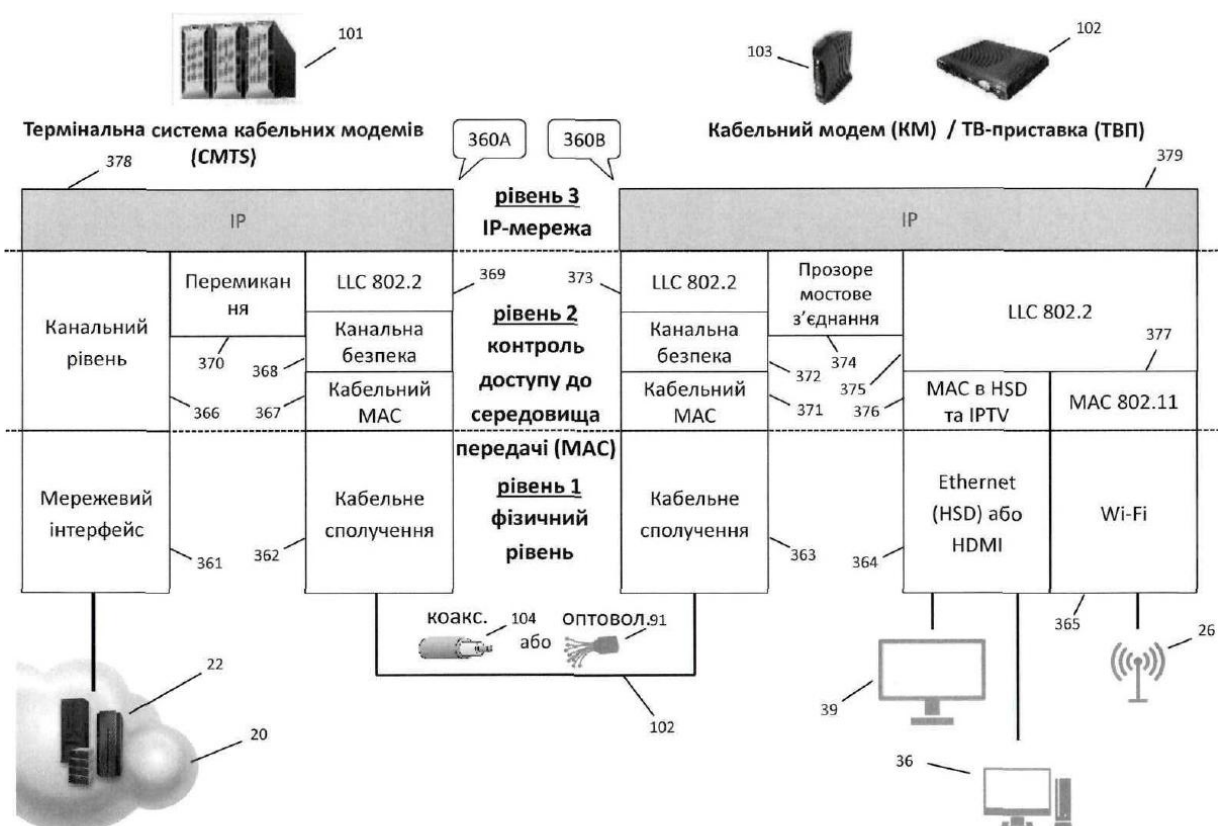


Рисунок 17

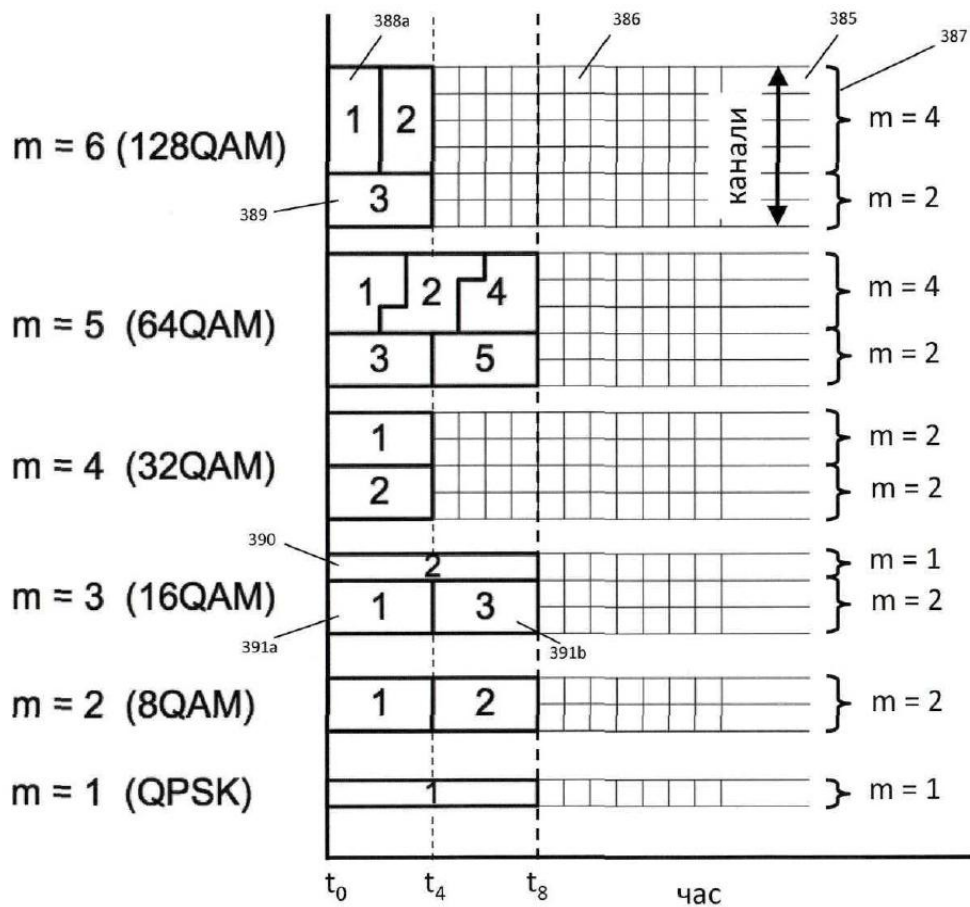


Рисунок 18

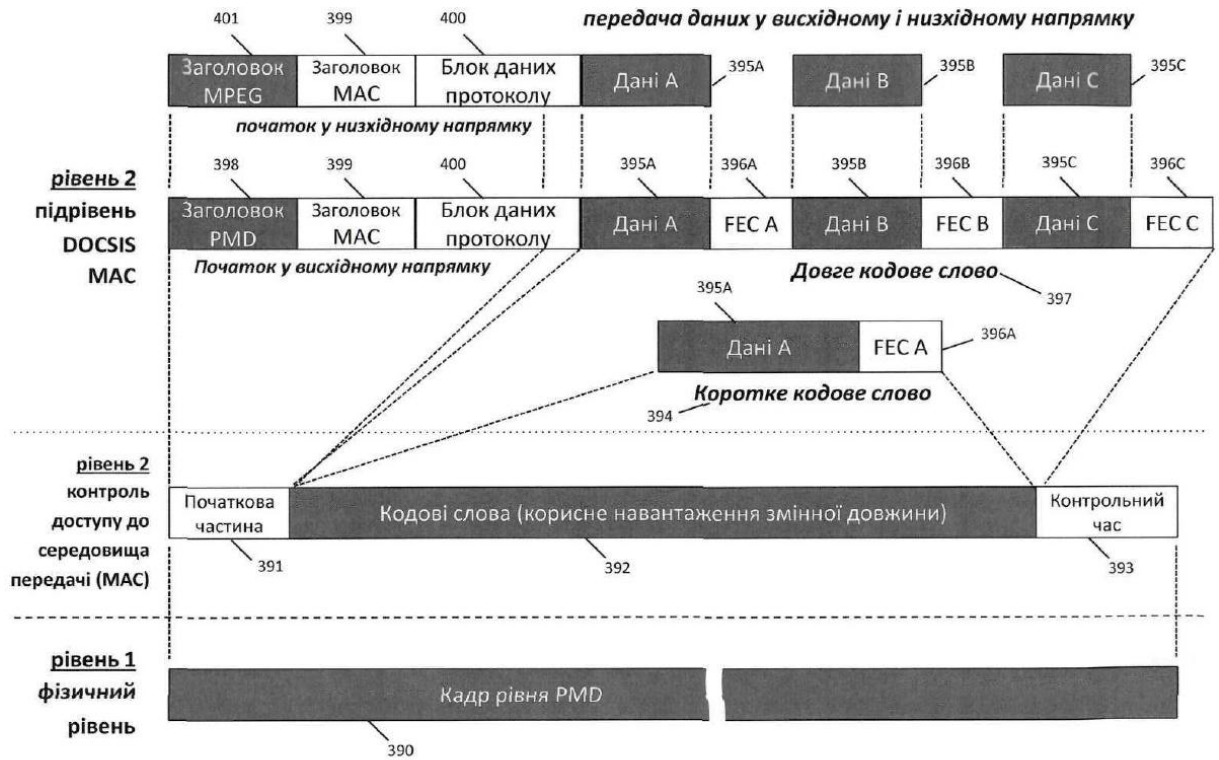


Рисунок 19

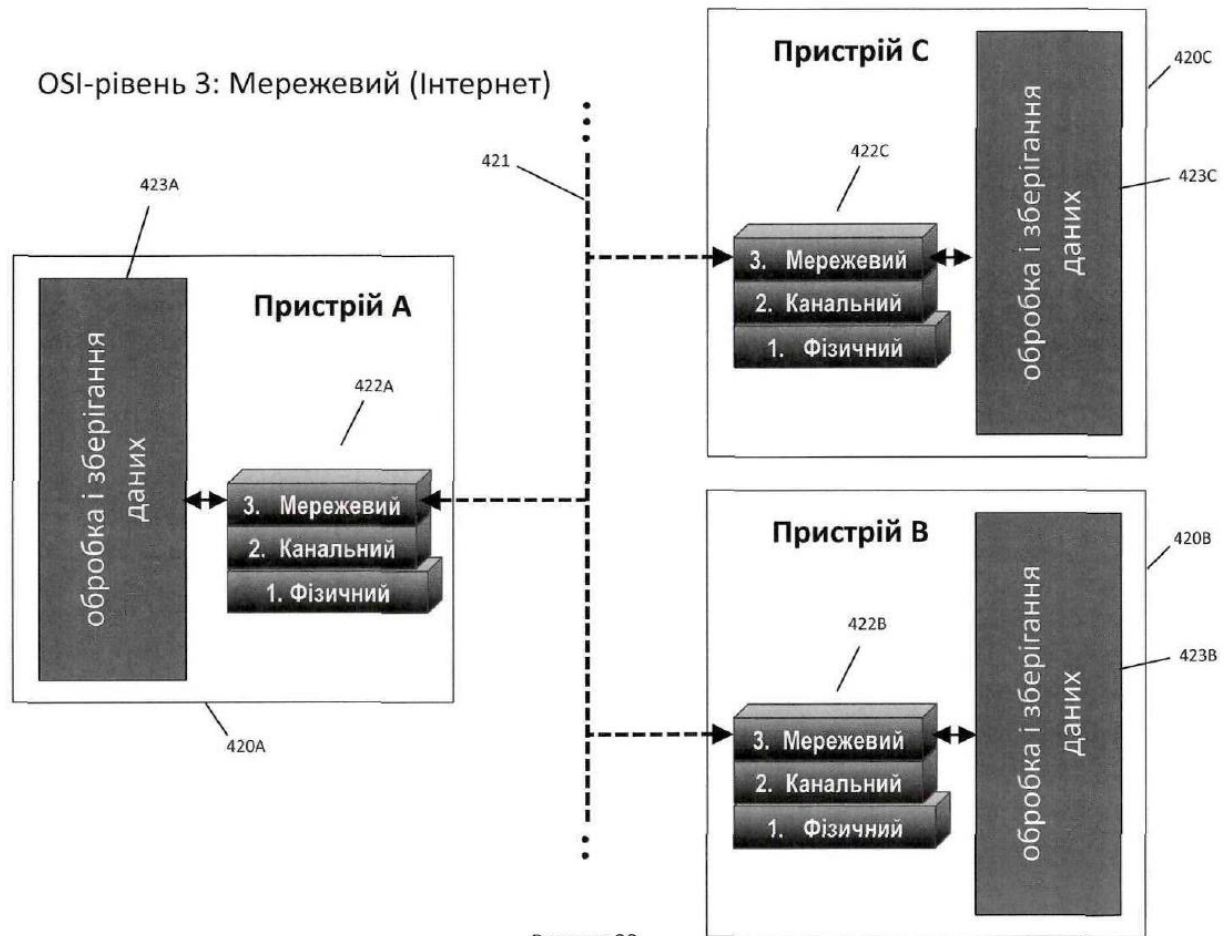


Рисунок 20

OSI-рівні від 1 до 7

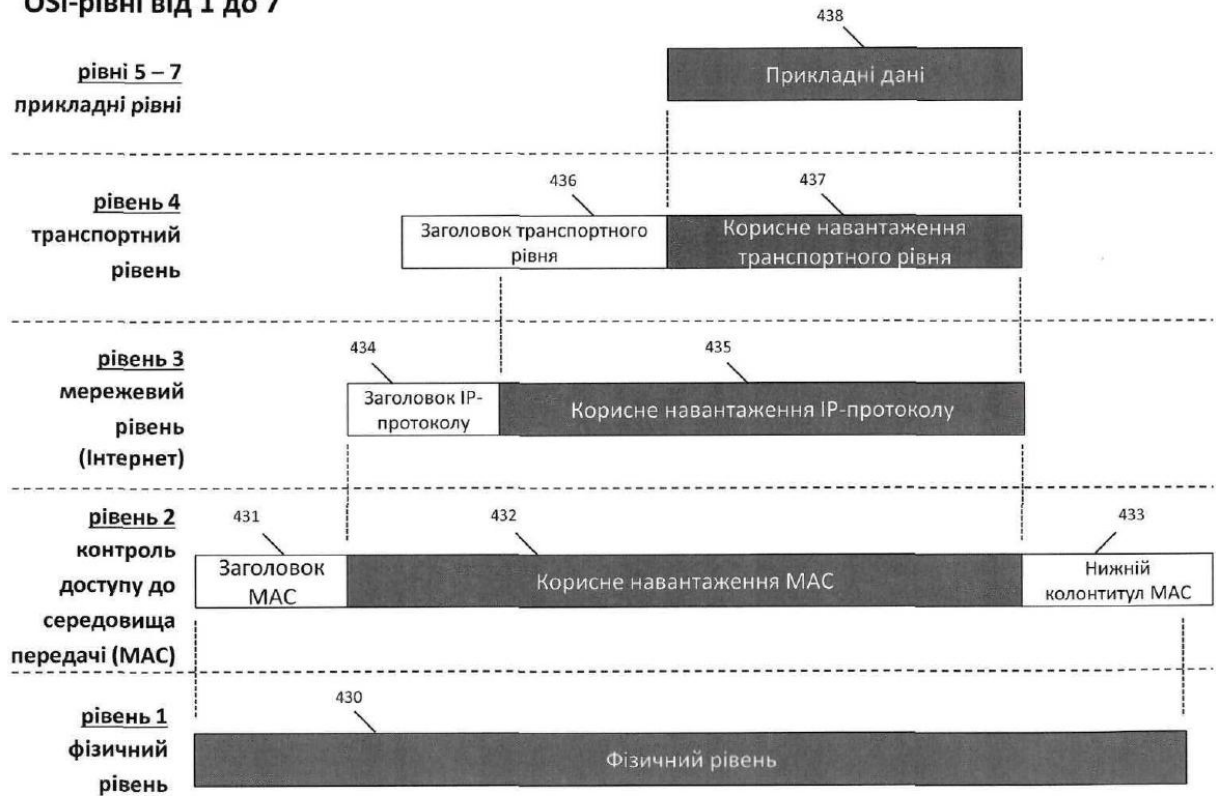


Рисунок 21

OSI-рівень 3: Мережевий (Інтернет)

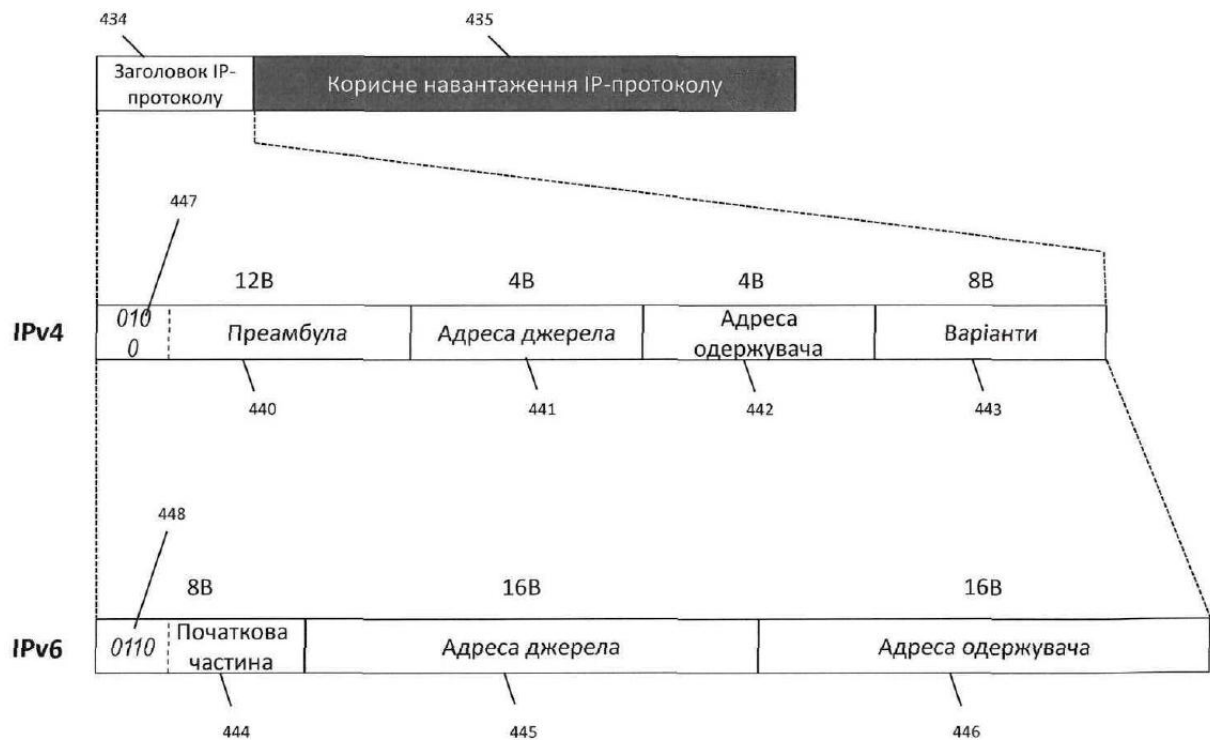


Рисунок 22

Інтернет-протокол IPv4



Рівень	Поле пакета IPv4		Розмір		Функція	
3	Версія	Розмір заголовка	4 біти	4 біти	Двійкове представлення (0100) IP-адреси у версії 4	Кіль-ть 32-бітових слів у заголовку від 20 до 60 Б
3	DSCP	ECN	6 біт	2 біти	Диференційовані служби для технології QoS	Явне повідомлення про перевантаження
3	Повна довжина		2 Б		Довжина IP-пакета, включаючи заголовок і дані, 20-65 535 Б	
3	Ідентифікація		2 Б		Унікальна ідентифікація групи фрагментів однієї IP-датаграми	
3	Пропор и	Зміщення фрагмента	3 біти	13 біт	Прапори для керування фрагментацією	Зміщення фрагмента в початку пакета
3	TTL	Протокол	1 Б	1 Б	Час життя для запобігання відмовам	Описує тип корисних даних IP-пакета
3	Контрольна сума заголовка		2 Б		16-бітна контрольна сума, що використовується для перевірки помилок заголовка, знижує пакети помилок	
3	Адреса джерела		4 Б		IPv4-адреса відправника пакета, NAT може змінювати IP-адресу	
3	Адреса одержувача		4 Б		IPv4-адреса одержувача пакета, NAT може змінювати IP-адресу	
3	Варіанти (якщо розмір заголовка > 5)		4 Б		Нечасто використовуване поле, яке піддається ризикам безпеки, активне для розміру заголовка = 6-15	

Рисунок 23

Інтернет-протокол IPv6



Рівень	Поле пакета IPv6		Розмір		Функція	
3	Версія	Клас трафіку	4 біти	8 біт	Двійкове представлення (0110) IP-адреси у версії 6	Диференційовані служби (6 біт) і ECN (2 біти)
3	Мітка потоку		20 біт		Підтримка маршруту, щоб уникнути перевпорядкування у додатках реального часу	
3	Довжина корисного навантаження		2 Б		Розмір корисного навантаження в октетах, включаючи заголовки розширень	
3	Наступний заголовок		1 Б		Зазначення типу наступного заголовка, що відповідає полю протоколу IPv4	
3	Ліміт переходів		1 Б		Заміна часу IPv4 на автоматично оновлюване поле, зменшення на 1 для кожного вузла	
3	Адреса відправника		16 Б		IPv6-адреса вузла-відправника	
3	Адреса одержувача		16 Б		IPv6-адреса вузла-одержувача	

Рисунок 24

OSI-рівень 3: Мережевий (Інтернет)

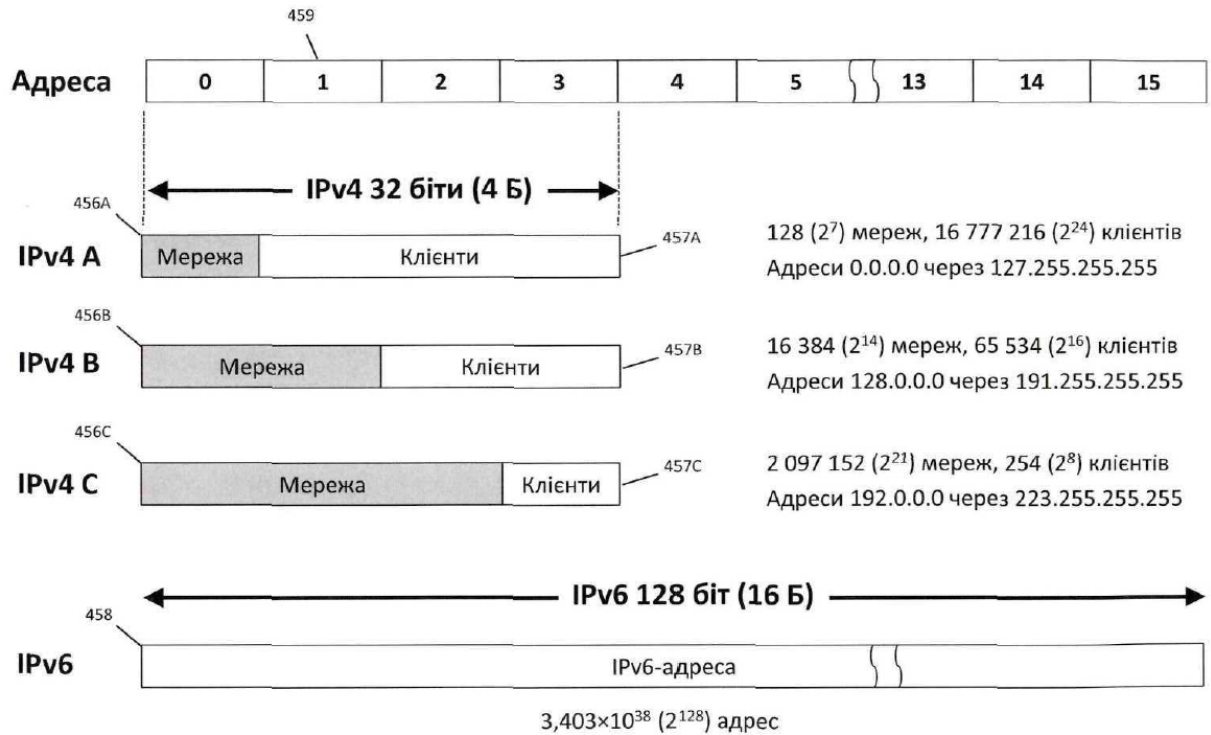


Рисунок 25

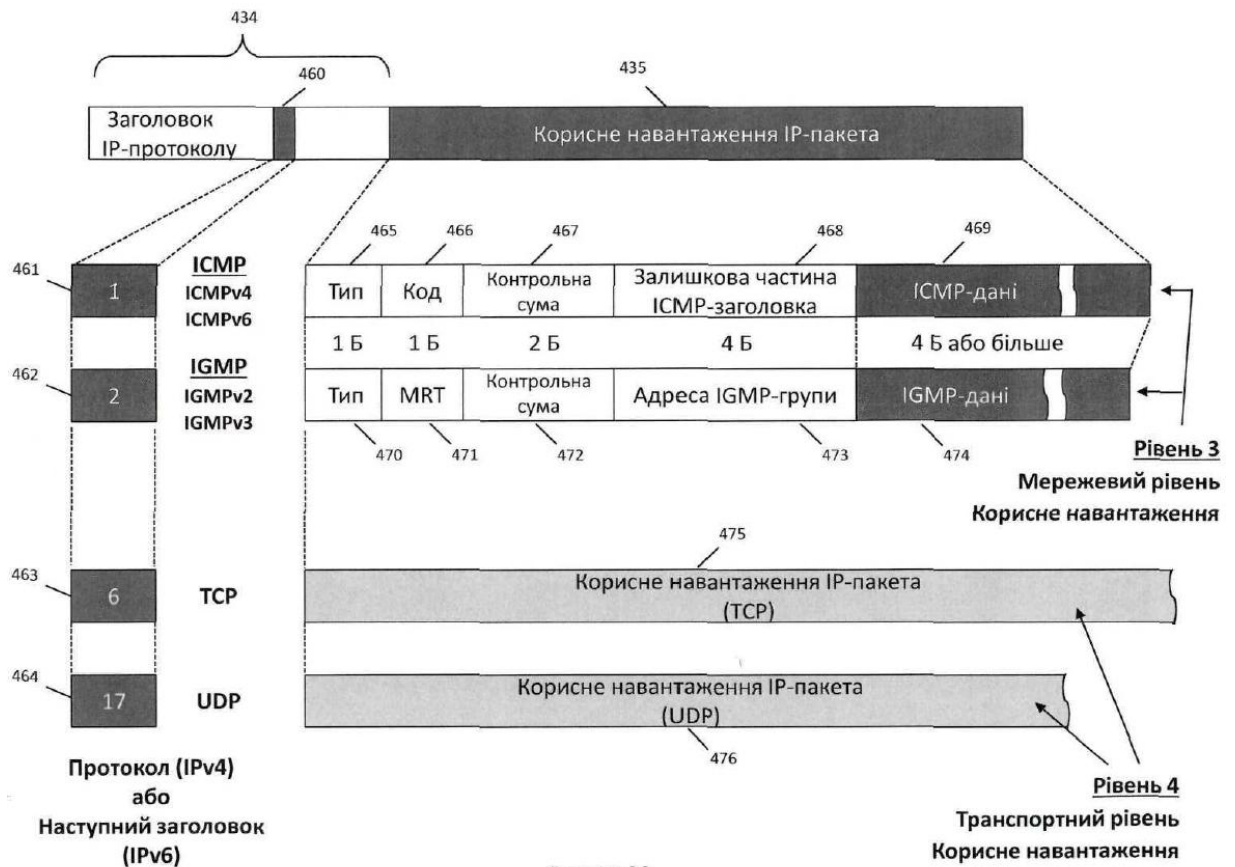


Рисунок 26

OSI-рівень 4: Транспортний

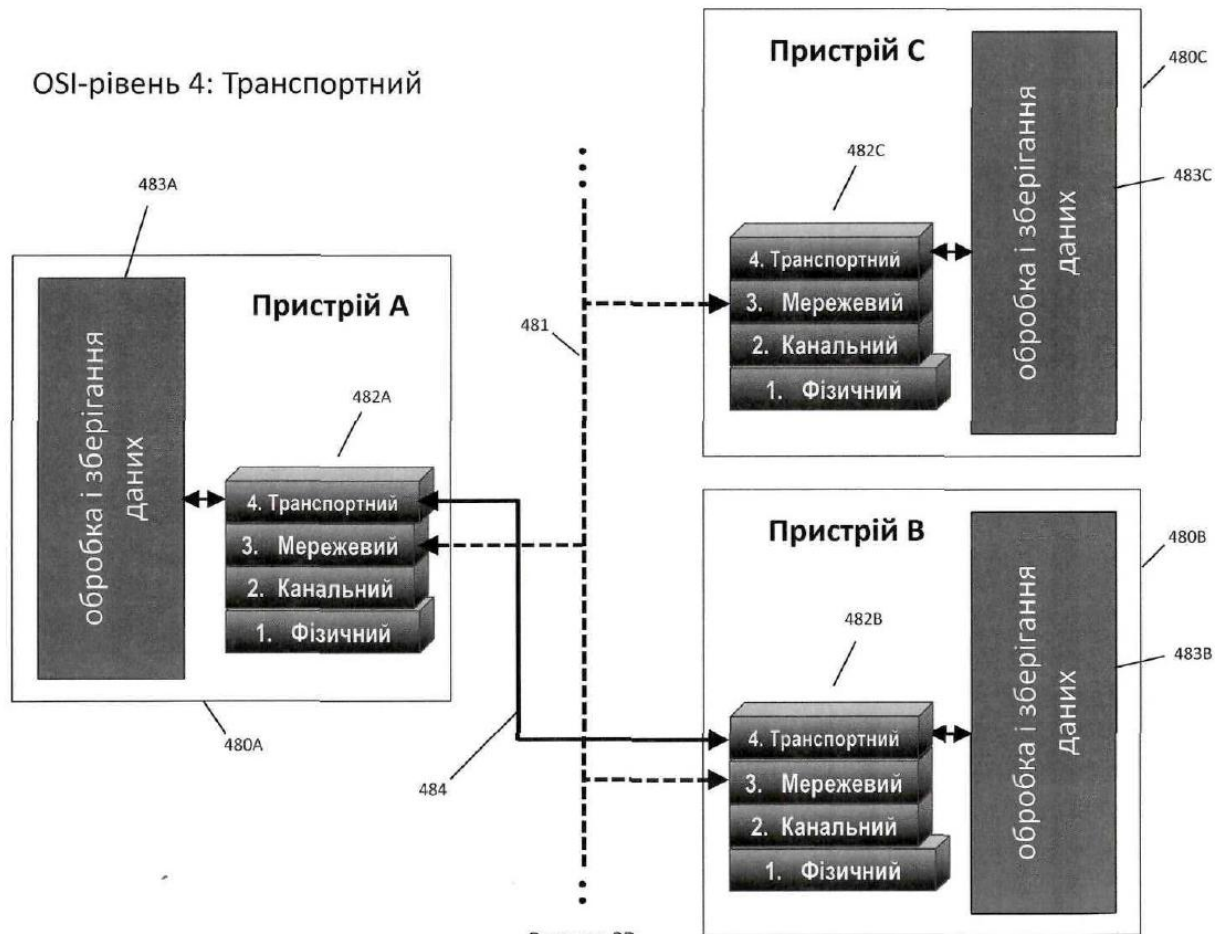


Рисунок 27

OSI-рівень 4: Транспортний (TCP)

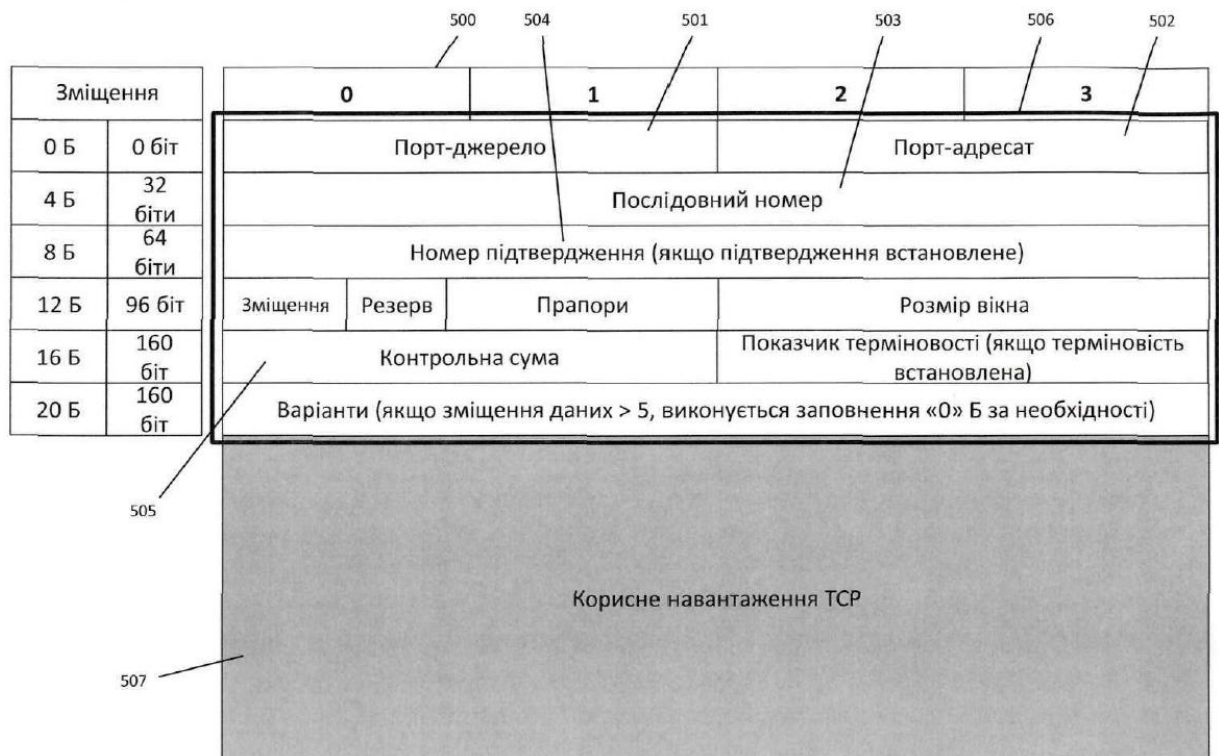


Рисунок 28A

508

Рівень	Поле TCP	Розмір	Функція TCP
4	Порт-джерело	2 Б	Порт відправника; порт, на який повинна бути надіслана відповідь
4	Порт-адресат	2 Б	Порт одержувача (обов'язково), номер стандартного порта по серверу
4	Послідовний номер	4 Б	Лічильник пакетів (SEQ), що збільшує кожну відповідь підтвердження на одну одиницю
4	Номер підтвердження	4 Б	Підтверджує отримання кожного пакета в послідовності за допомогою послідовного номера
4	Зміщення	4 біти	Розмір TCP-заголовка (зміщення від TCP-заголовка до початку даних), 20-60 Б
4	Резерв	3 біти	Зарезервовано для використання в майбутньому, за замовчуванням до 000 (двійкова система)
4	Прапори	9 біт	Керування бітами для перевантаження, терміновості, підтвердження, синхронізації та скидання
4	Розмір вікна	2 Б	Розмір вікна прийому
4	Контрольна сума	2 Б	Перевірка помилок, яка включає контрольну суму (16 біт) даних заголовка і корисного навантаження
4	Показчик терміновості	2 Б	Зміщення від SEQ, яке вказує останній терміновий байт даних (якщо терміновість встановлена на 1)
4	Варіанти	1-40 Б	Опціональний (1 Б обов'язково), довжина (1 Б опціонально), дані (змінні) згідно із зміщенням
4-7	Корисне навантаження TCP	змінний	Корисне навантаження TCP

Рисунок 288

Рівень 4 OSI : Транспортний (TCP)

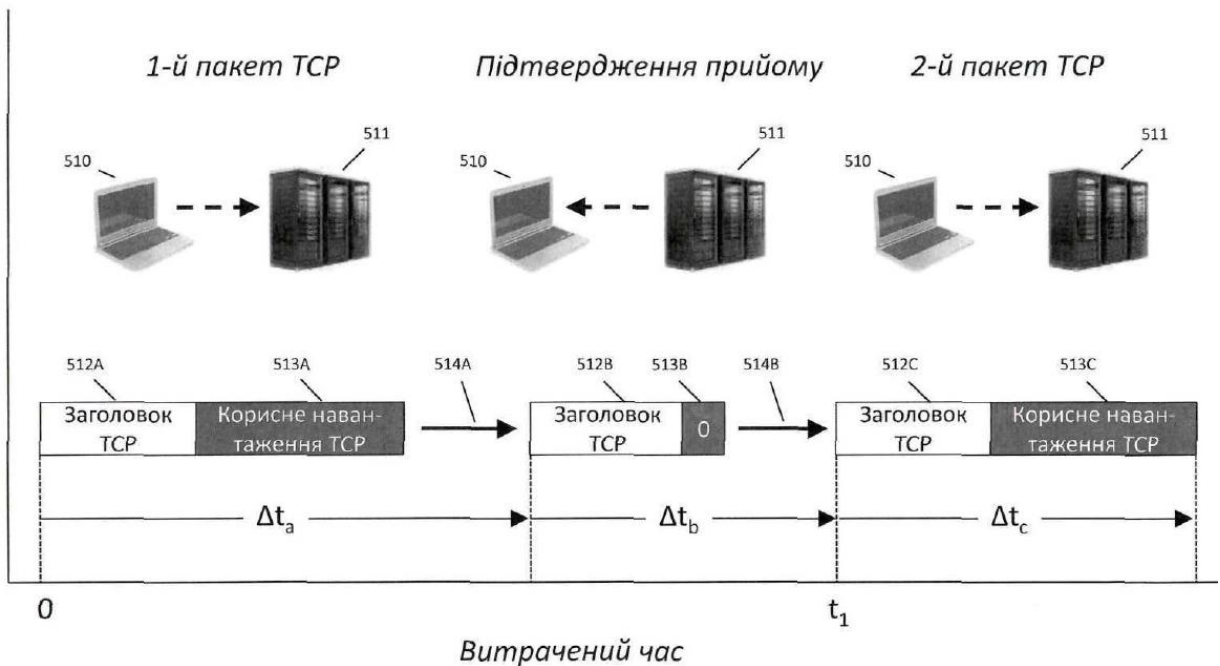
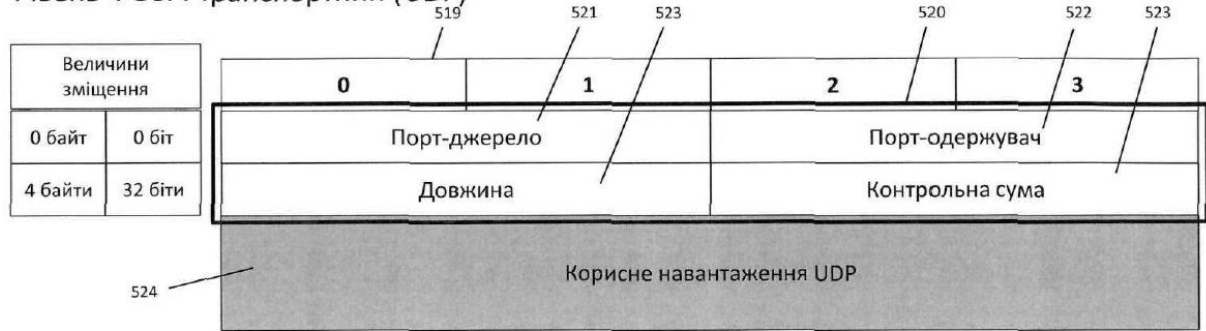


Рисунок 29

Рівень 4 OSI : Транспортний (UDP)



Рівень	Поле UDP	Розмір	Функція
4	Порт-джерело	2 байти	Порт відправника; порт, на який повинна бути надіслана відповідь
4	Порт-одержувач	2 байти	Порт одержувача (необхідний); порт із добре відомим номером у разі використання сервера
4	Довжина	2 байти	Загальна довжина заголовка UDP і корисного навантаження, від 8 байт до 65535 байт
4	Контрольна сума	2 байти	Перевірка помилок; необхідно у версії IPv6; використовуються псевдо-заголовки у версіях IPv4 и IPv6
4-7	Корисне навантаження UDP	варіюється	Дані корисного навантаження UDP, від 0 байт до 65507 байт у версії IPv4

Рисунок 30

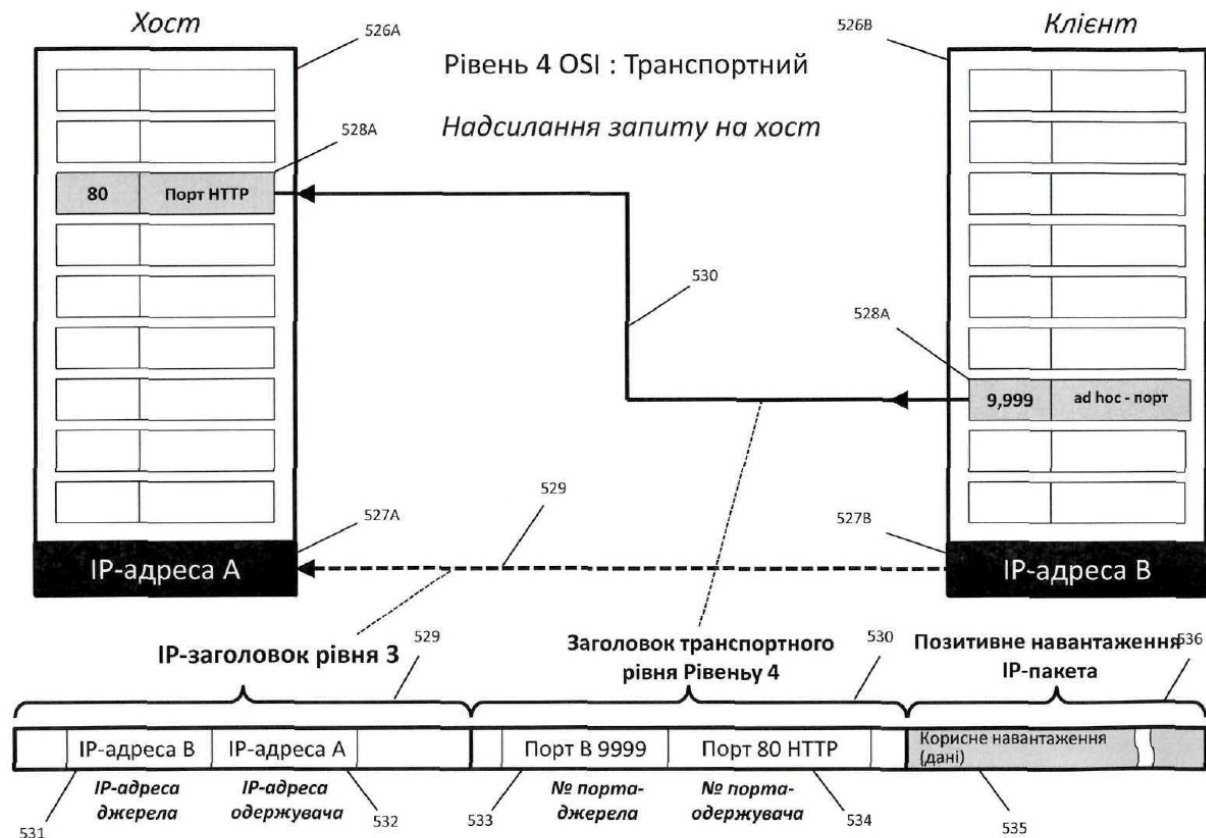


Рисунок 31А

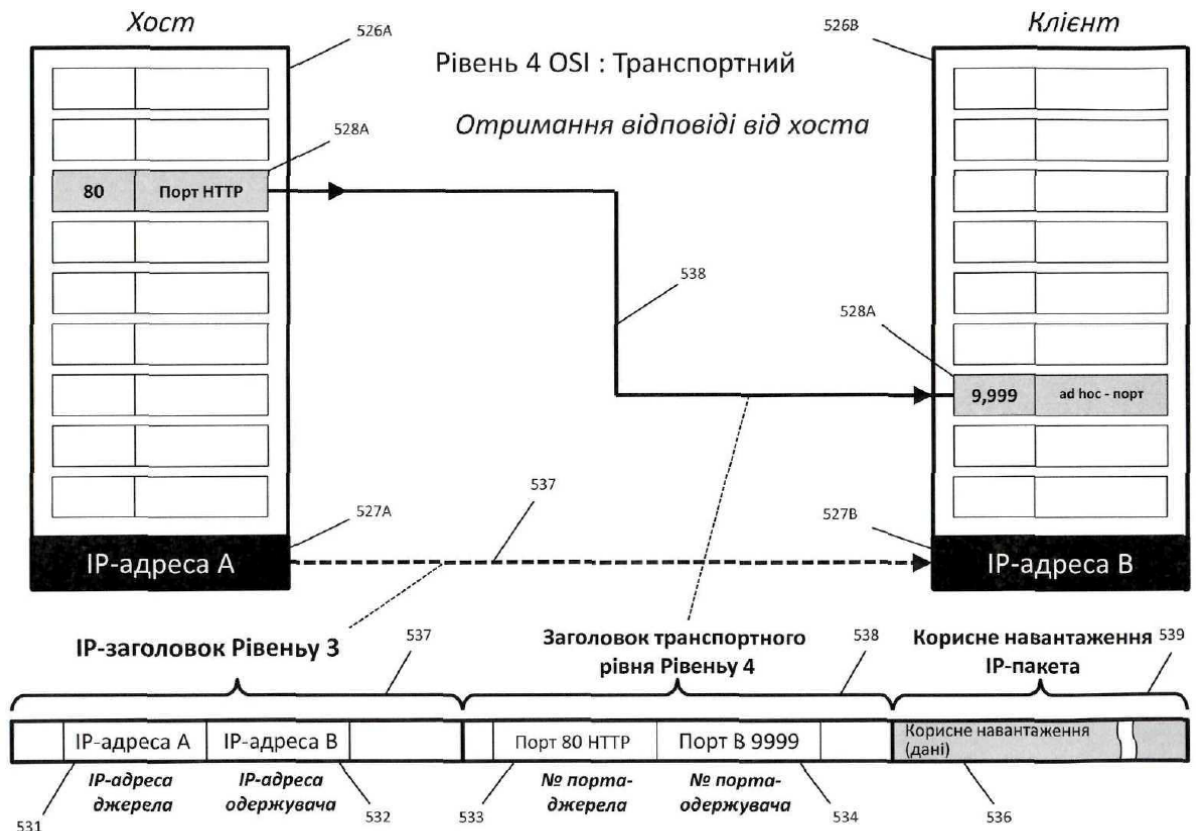


Рисунок 31В

№ порта	UDP	TCP	Опис порта	Зареєстрований
7	UDP	TCP	Echo-запити/Утиліта Ping (замінена протоколом ICMP)	Офіційно
20	UDP	TCP	Передача даних по протоколу FTP	Офіційно
21	—	TCP	Керування по протоколу FTP (передача команд)	Офіційно
22	UDP	TCP	Протокол тунелювання TCP-з'єднань (SSH) для безпечного входу в систему (SCP), Протокол передачі файлів поверх протоколу SSH (SFTP), перенаправлення портів	Офіційно
23	UDP	TCP	Протокол Telnet	Офіційно
25	—	TCP	Простий протокол пересилання пошти (SMTP)	Офіційно
26	UDP	TCP	Зашифрований протокол SMTP	Офіційно
53	UDP	TCP	Сервер доменних імен (DNS)	Офіційно
80	—	TCP	Протокол передачі гіпертекстових файлів (HTTP)	Офіційно
110	—	TCP	Поштовий офісний протокол (POP3)	Офіційно
143	—	TCP	Протокол доступу до інтернет-повідомлень (IMAP)	Офіційно
220	UDP	TCP	Протокол доступу до повідомлень Інтернет (IMAP v3)	Офіційно
443	—	TCP	Протокол передачі гіпертекстових файлів по TLS/SSL (HTTPS)	Офіційно
465	—	TCP	Протокол простого обміну електронною поштою по TLS/SSL (SMTPS)	Офіційно
989	UDP	TCP	Протокол FTPS (дані): FTP по TLS/SSL	Офіційно
990	UDP	TCP	Протокол FTPS (керування): FTP по TLS/SSL	Офіційно
992	UDP	TCP	Протокол TELNET по TLS/SSL	Офіційно
993	—	TCP	Протокол доступу до повідомлень Інтернет по TLS/SSL (IMAPS)	Офіційно
995	—	TCP	Поштовий протокол 3 по TLS/SSL (POP3S)	Офіційно

Рисунок 31С

541

Номер порта	UDP	TCP	Опис порта	Зареєстрований
від 0 до 1023	UDP	TCP	Системні порти	Як правило
від 1024 до 49151	UDP	TCP	Зареєстровані порти, відкриті (динамічні) порти	Необов'язково
від 49152 до 65535	UDP	TCP	Відкриті (динамічні), приватні та ефемерні порти	Ні

Рисунок 31D

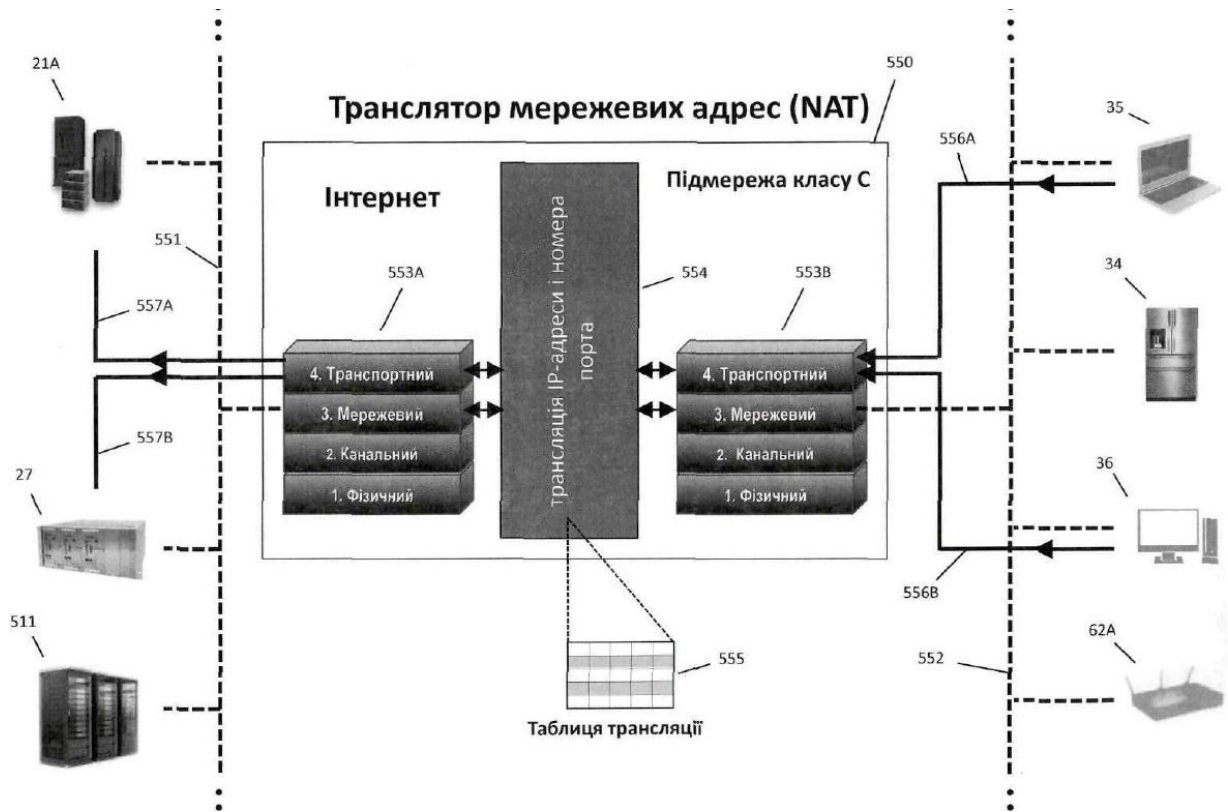


Рисунок 32A

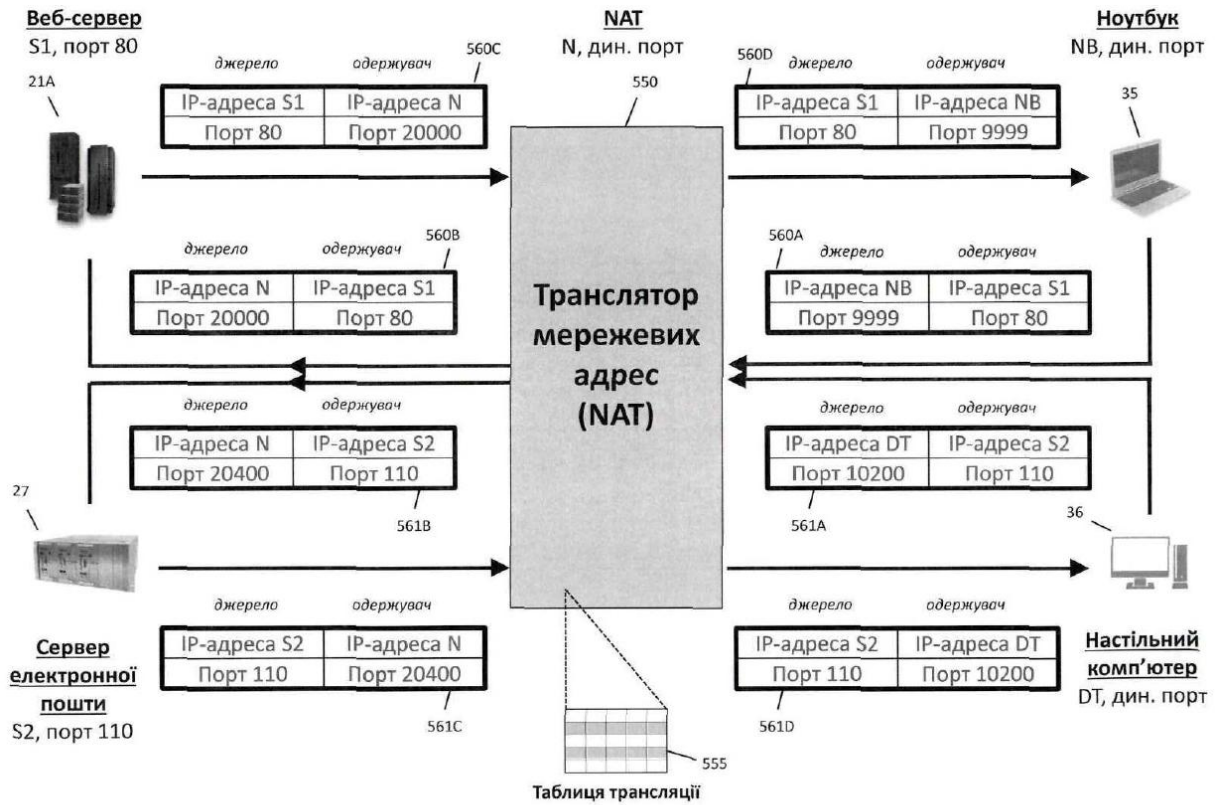


Рисунок 32В

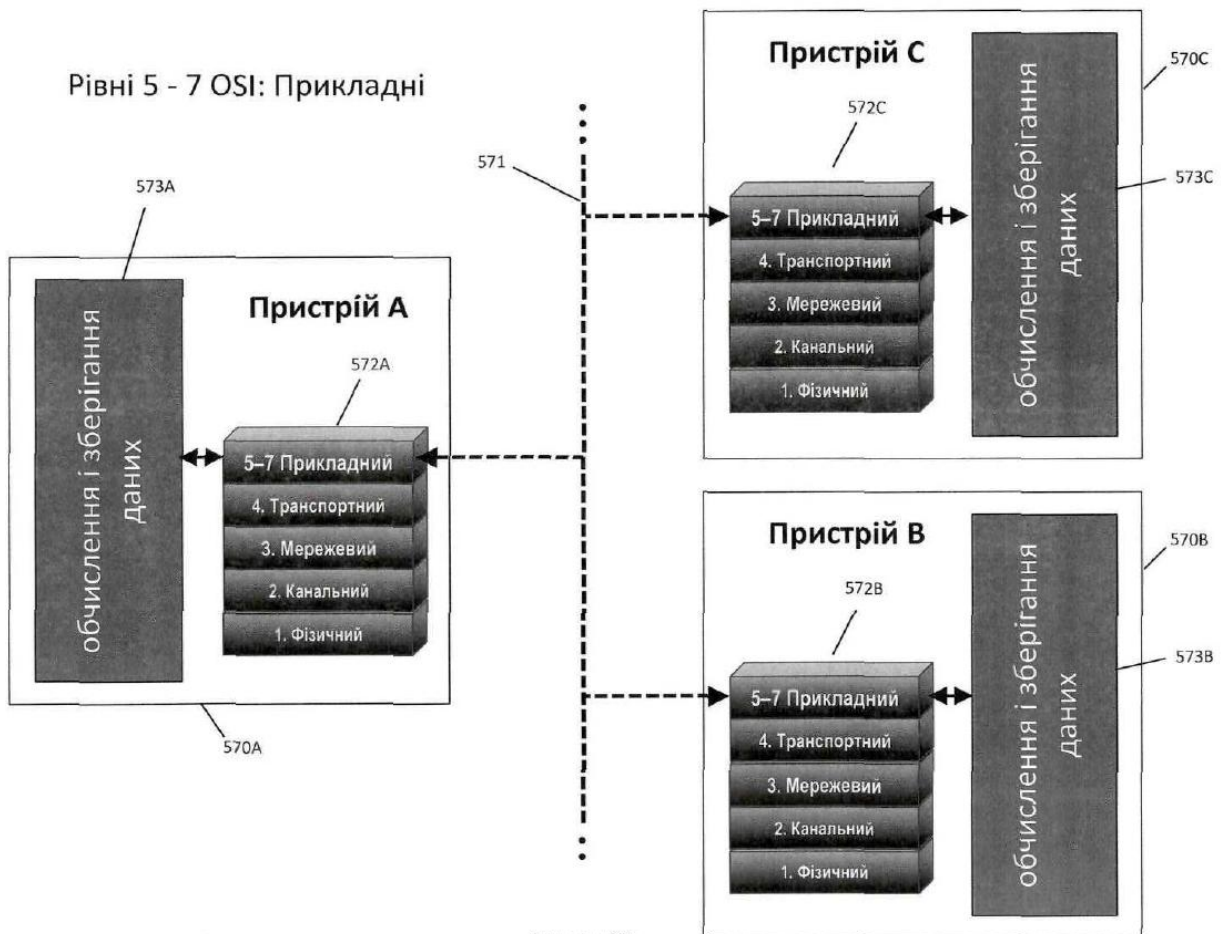
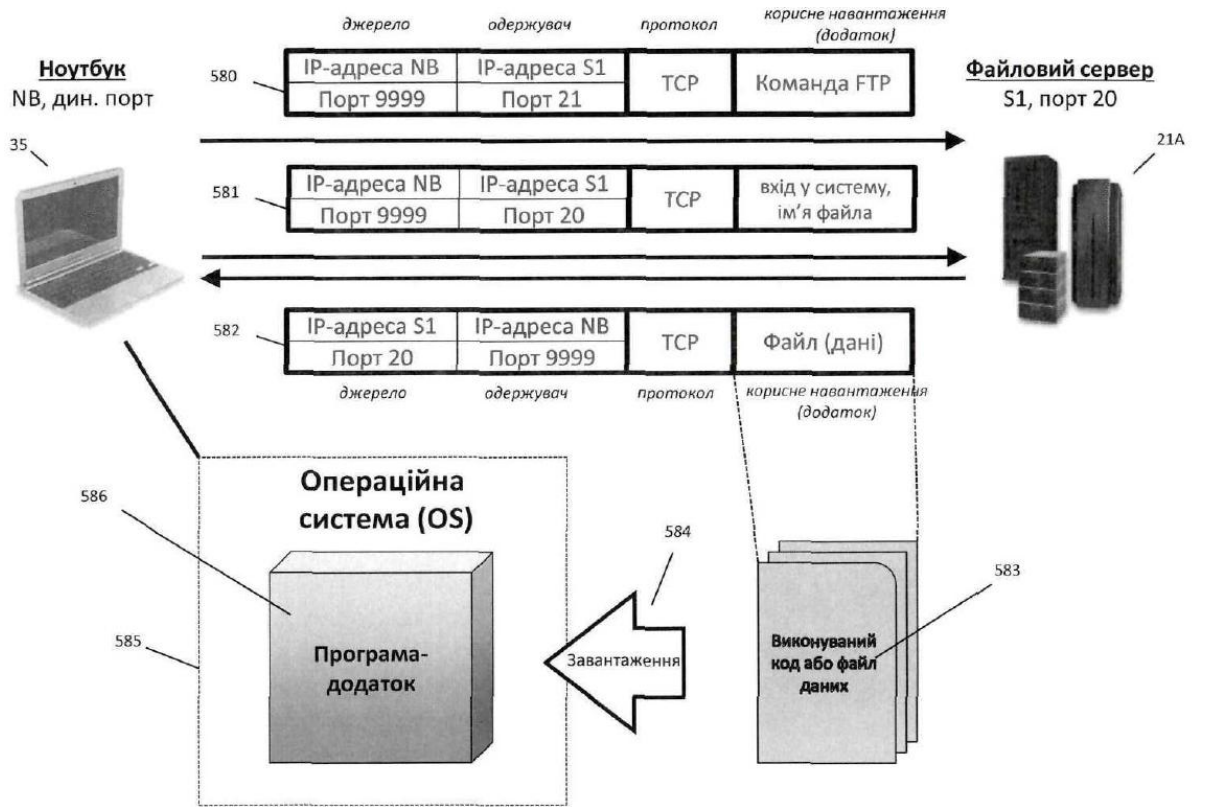


Рисунок 33



Рівень 7 OSI : Завантаження контенту через FTP

Рисунок 34

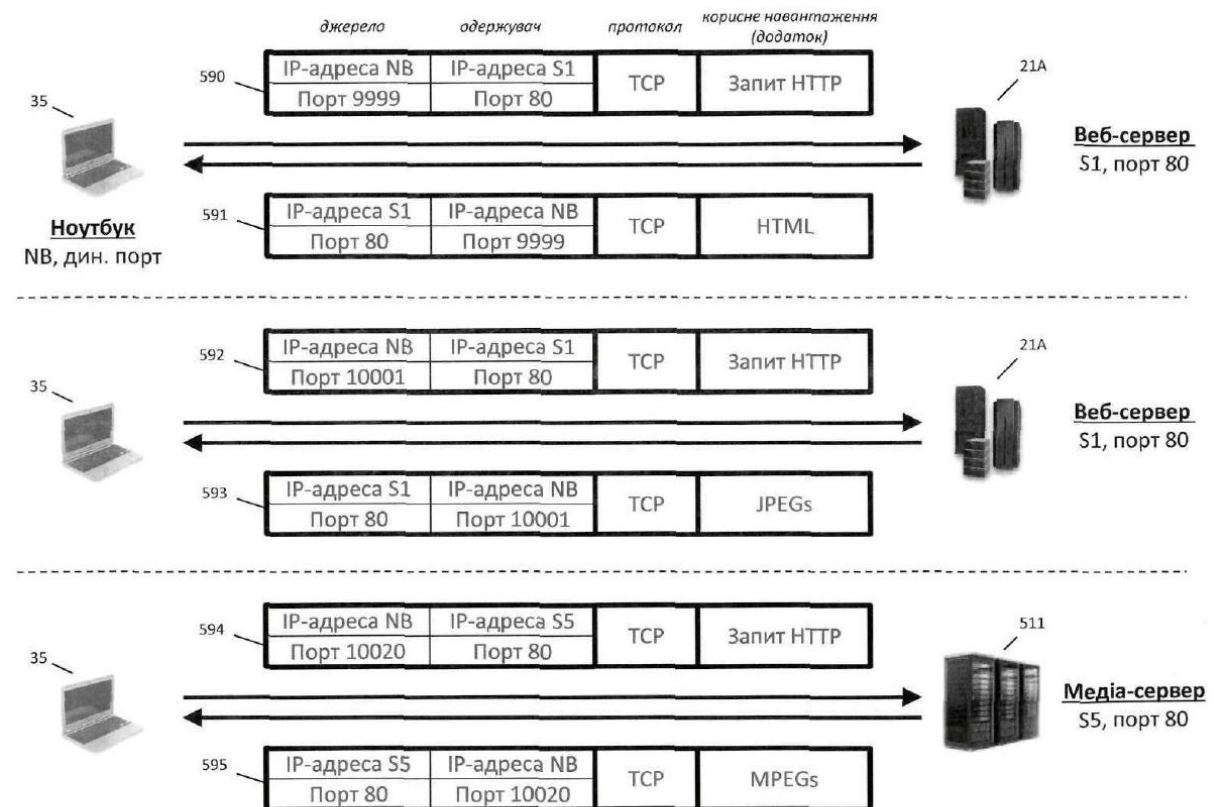


Рисунок 35A

Рівень 7 OSI : завантаження веб-сторінки через HTTP

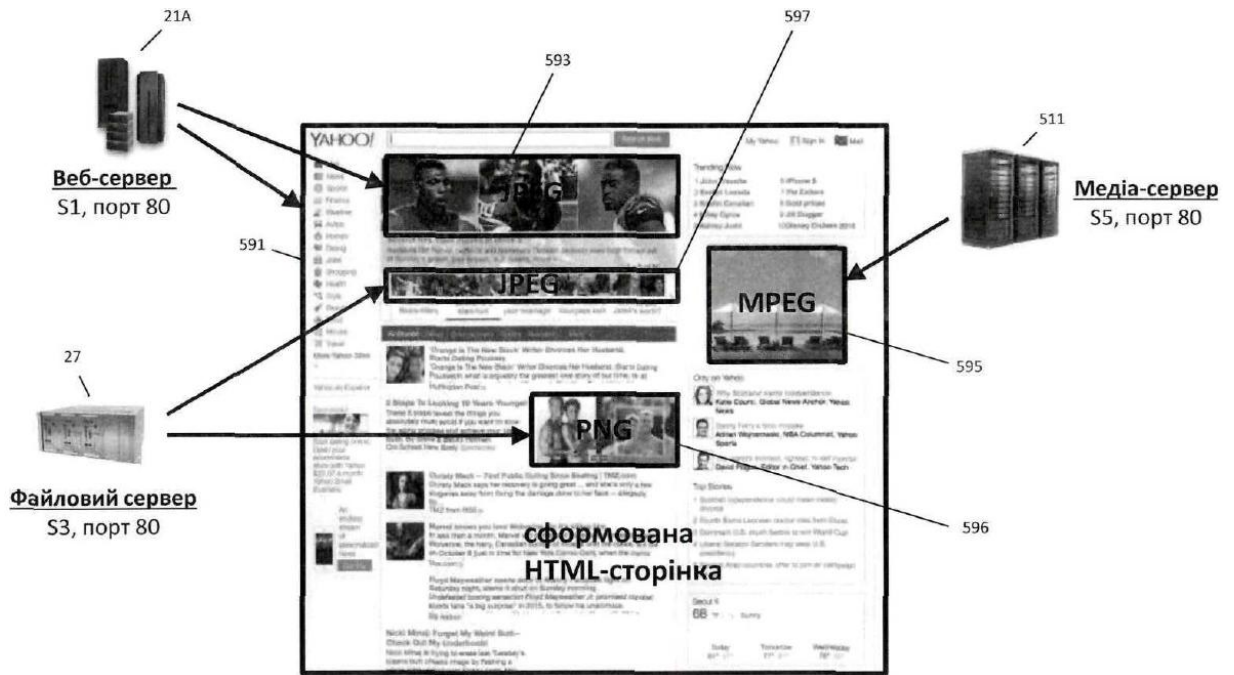
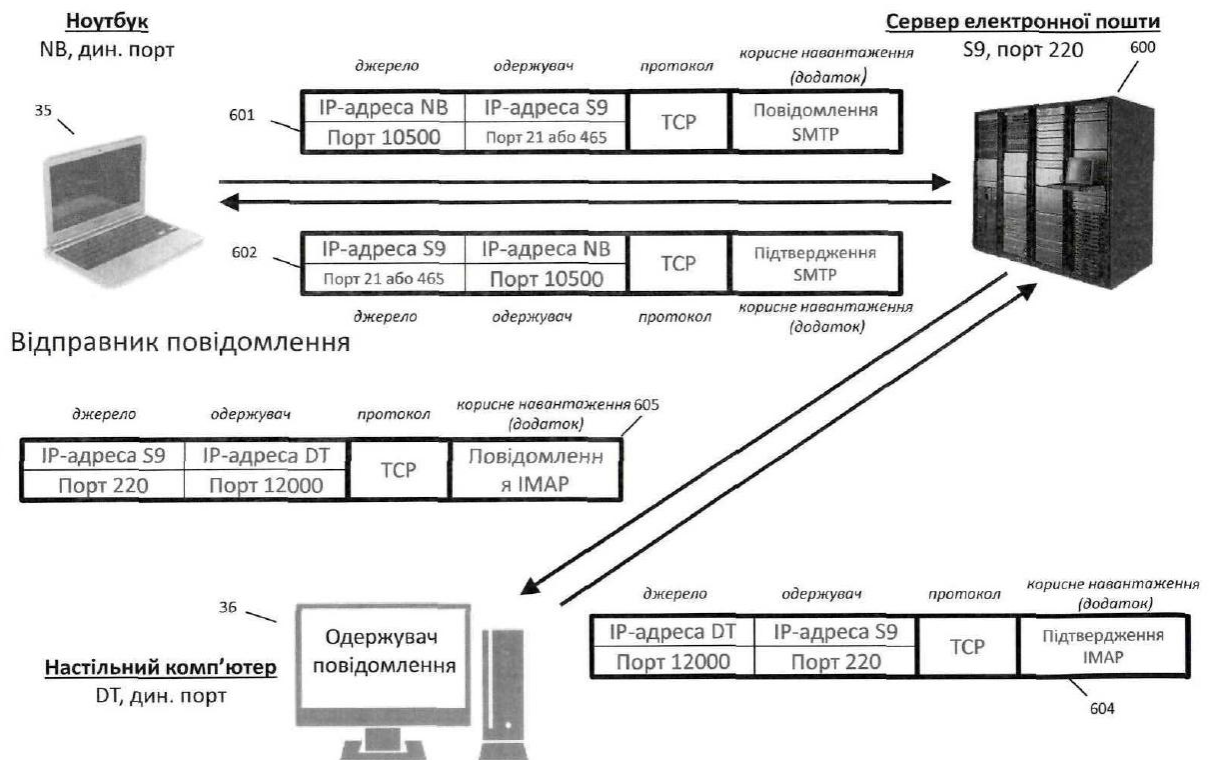


Рисунок 35В



Рівень 7 OSI : Додаток електронної пошти через IMAP

Рисунок 36

Якість послуги (технологія QoS)

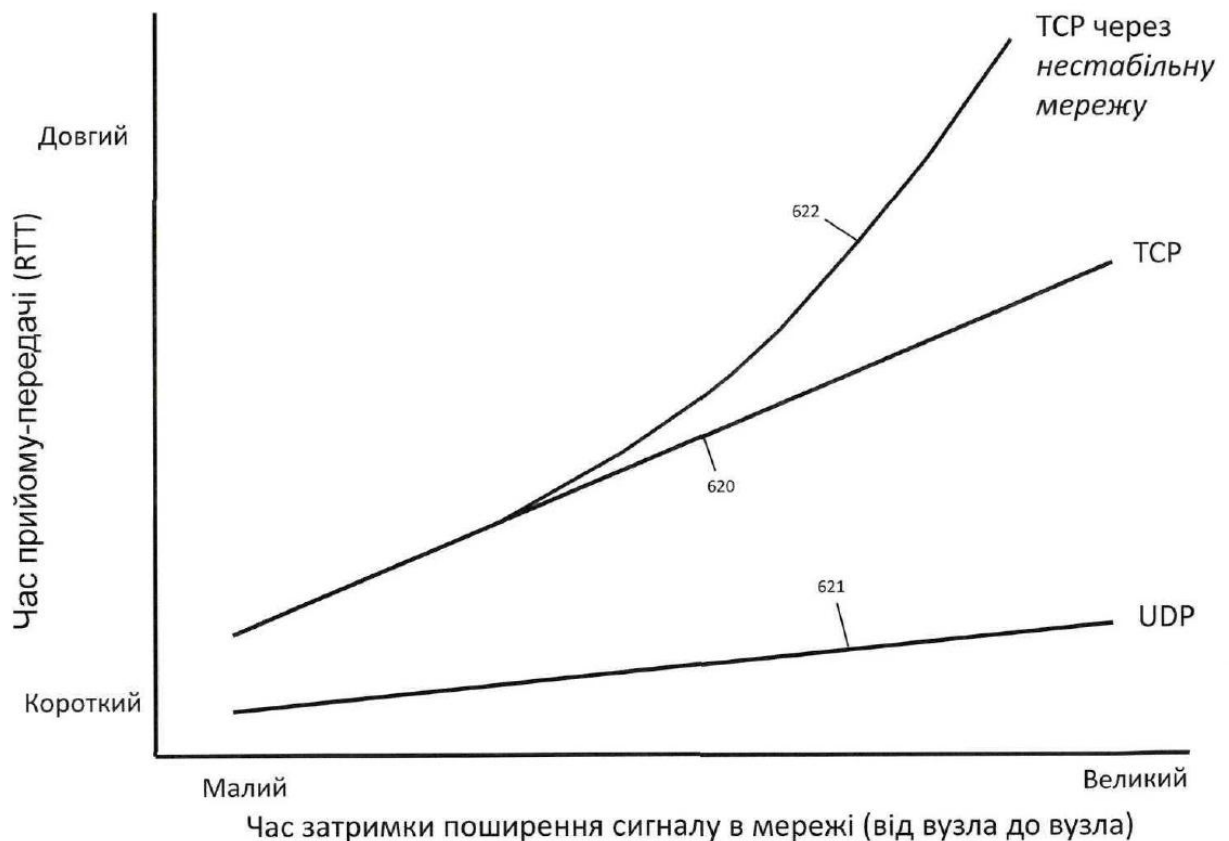
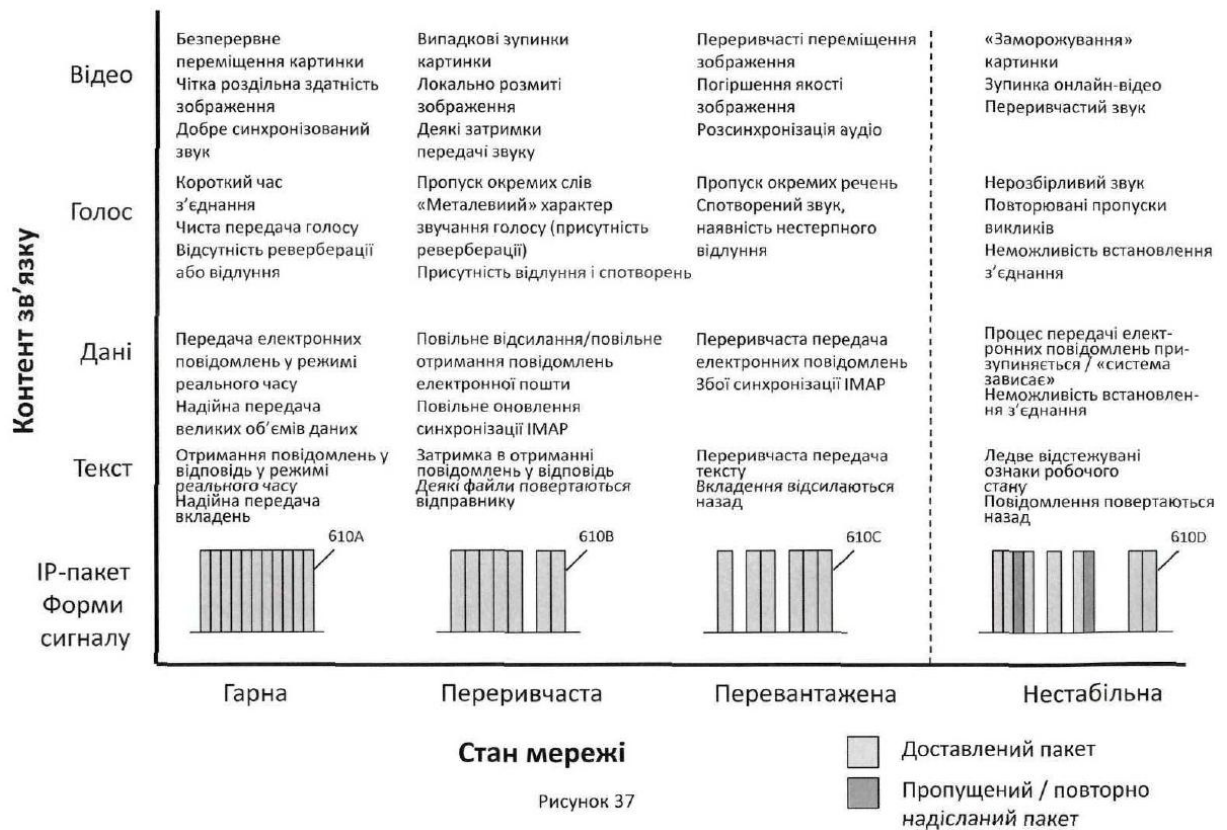


Рисунок 38

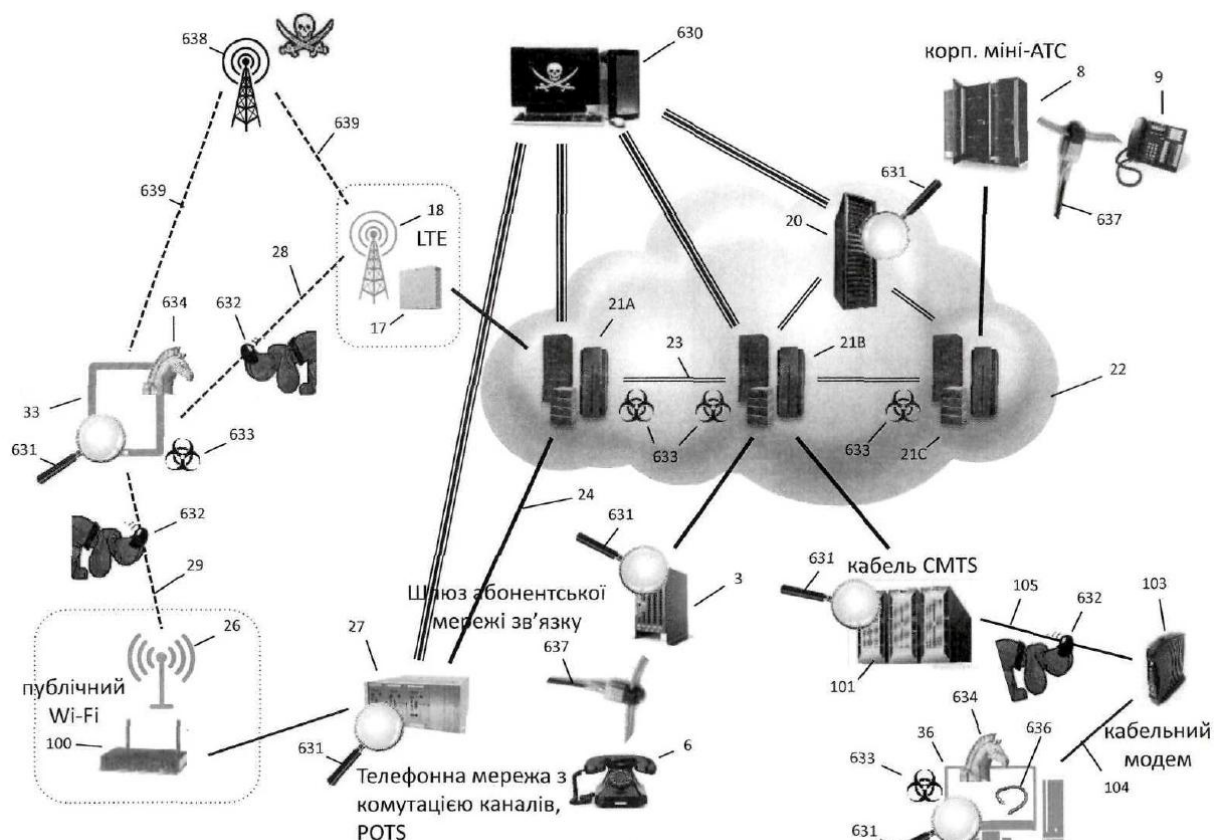


Рисунок 39

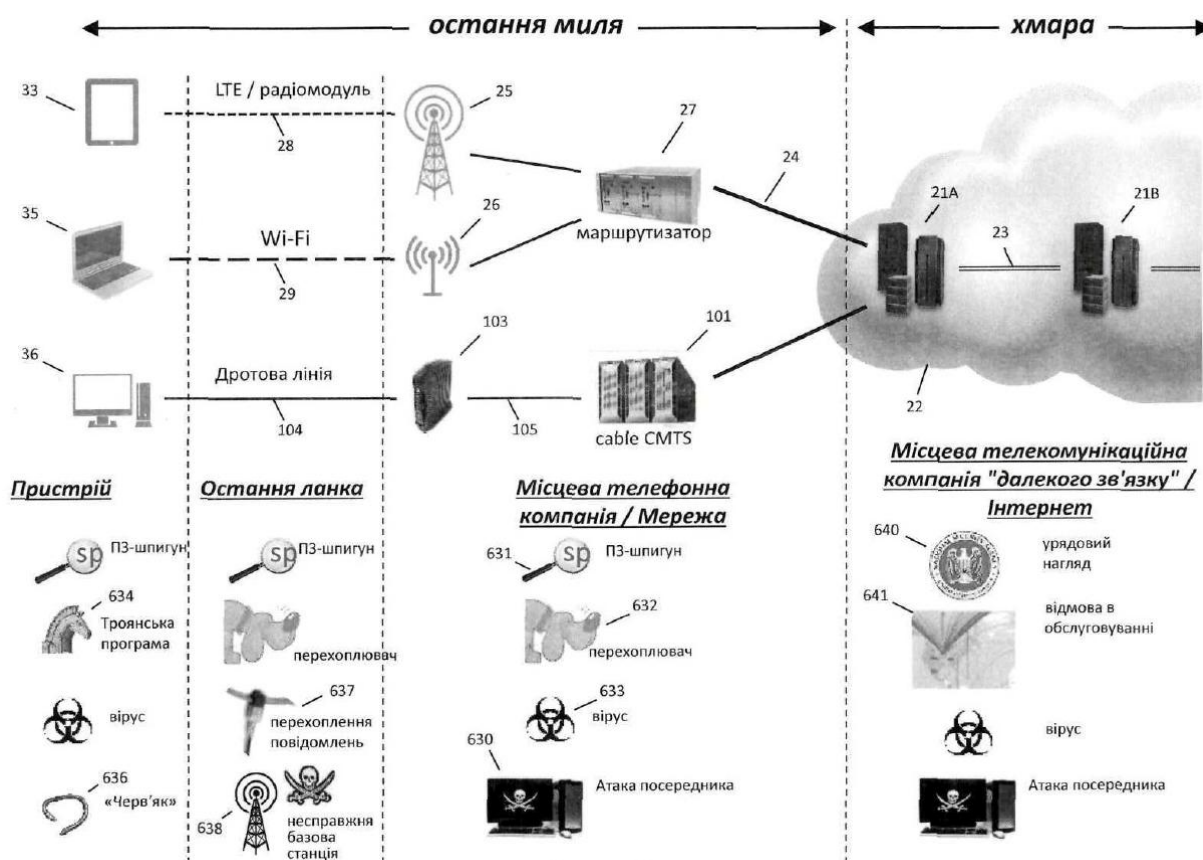


Рисунок 40

Моніторинг
мережі Ethernet



650

Моніторинг
мережі Wi-Fi



651

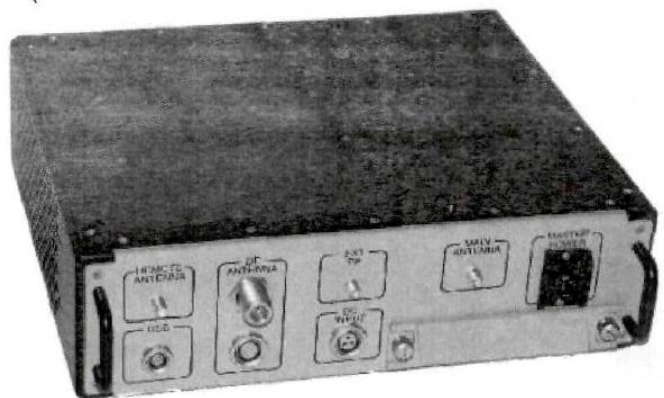
Рисунок 41А

Моніторинг стільникової мережі



652

Моніторинг стільникової мережі



653

Рисунок 41В

Моніторинг оптоволоконної мережі

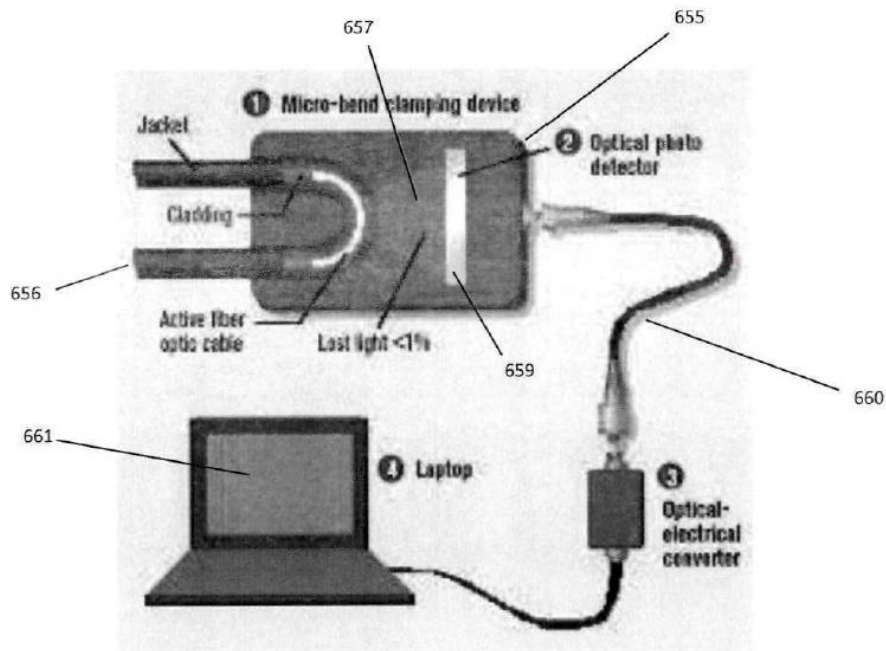


Рисунок 41С

Порівняння 10 найпоширеніших комерційних шпигунських програм

FEATURES										
OS Support ①	Android, iPhone, iPad, BlackBerry, Symbian, Nokia, Windows Mobile	Android, iPhone, BlackBerry, Symbian	iPhone, Android, BlackBerry, Windows Mobile, Symbian	Android, iPhone, BlackBerry	Android	Android, iPhone, BlackBerry, Symbian	iPhone, Android, BlackBerry, Windows Mobile, Symbian	iPhone, BlackBerry, Android, Symbian, S60, Nokia, Windows Mobile	iPhone, iPad, BlackBerry, Android, Symbian, Windows Mobile	iPhone, Android, BlackBerry, Nokia phone, Windows Mobile
SPY on Calls ①	✓	✓	View Call History	✓	Voice Call Log	✓	✓	✓	✓	✓
SPY on SMS and MMS ①	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SPY on Emails ①	✓	Email Logging	✓	✓	✓	✓	✓	✓	✓	✗
Track GPS Location ①	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Monitor Internet Use ①	Browsing History, Website Bookmarks, Blocking Websites	Browsing Website History	URL Tracking	View Visited Websites	View Visited Websites	View Visited Websites, Bookmarks	Browsing History	✗	✓	✗
Access Address Book ①	✓	✗	✓	✓	✗	✓	✓	✓	✓	✗
Access Calendar ①	✓	Monitor Appointments	✓	✓	✗	✓	✗	✗	✗	✗
Instant Messages ①	Skype, WhatsApp, iMessage	Skype, WhatsApp, BlackBerry Chat Logging	WhatsApp, iMessage, BlackBerry Message Chats	Skype, Gtalk, BBM, WhatsApp, iMessage, Facebook, Viber	✗	Skype, Gtalk, BBM, WhatsApp, iMessage, Facebook, Viber, WeChat, Yahoo	✓	✗	BBM, Facebook Chat	✗
Bugging ①	✓	✗	✗	✓	✗	✓	✗	✗	✗	✗
Control Apps ①	✓	✗	✓	✓	✗	✗	✓	✗	✗	✗
View Photos/Videos ①	✓	✓	✓	✓	✗	✗	✓	✓	✓	✗
Remote Control ①	✓	✗	✗	✓	Block Websites	✓	✓	✗	✓	✓

Рисунок 42

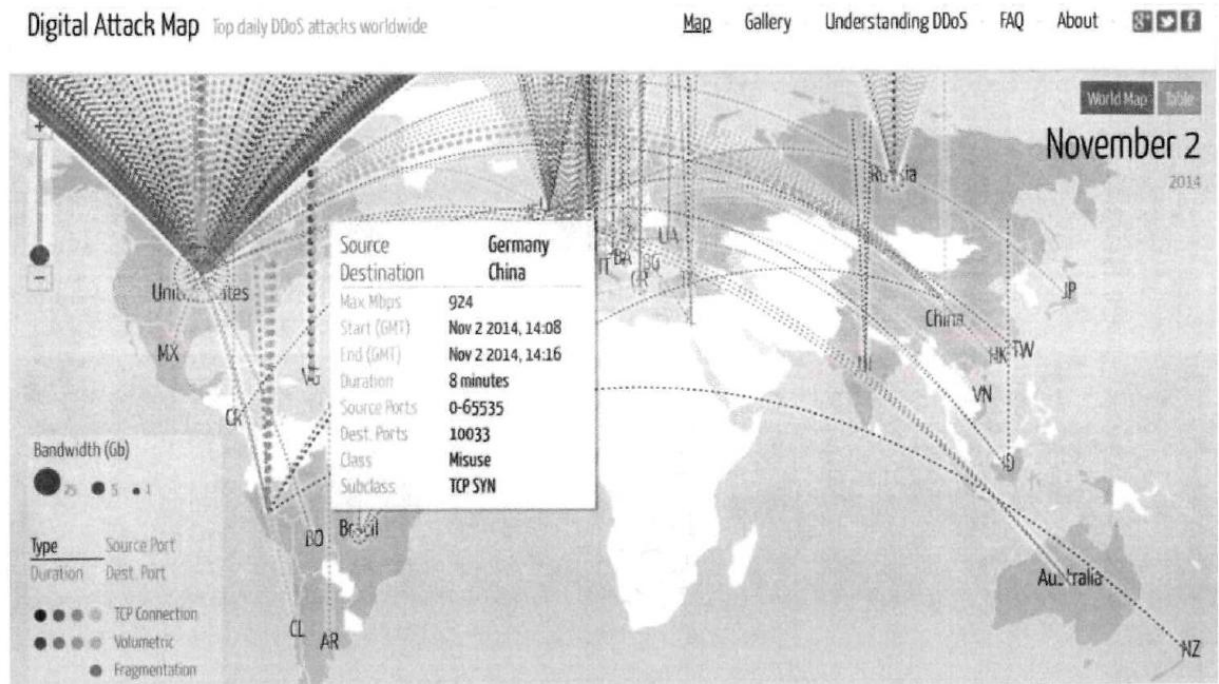


Рисунок 43

Перехоплення IP-пакетів

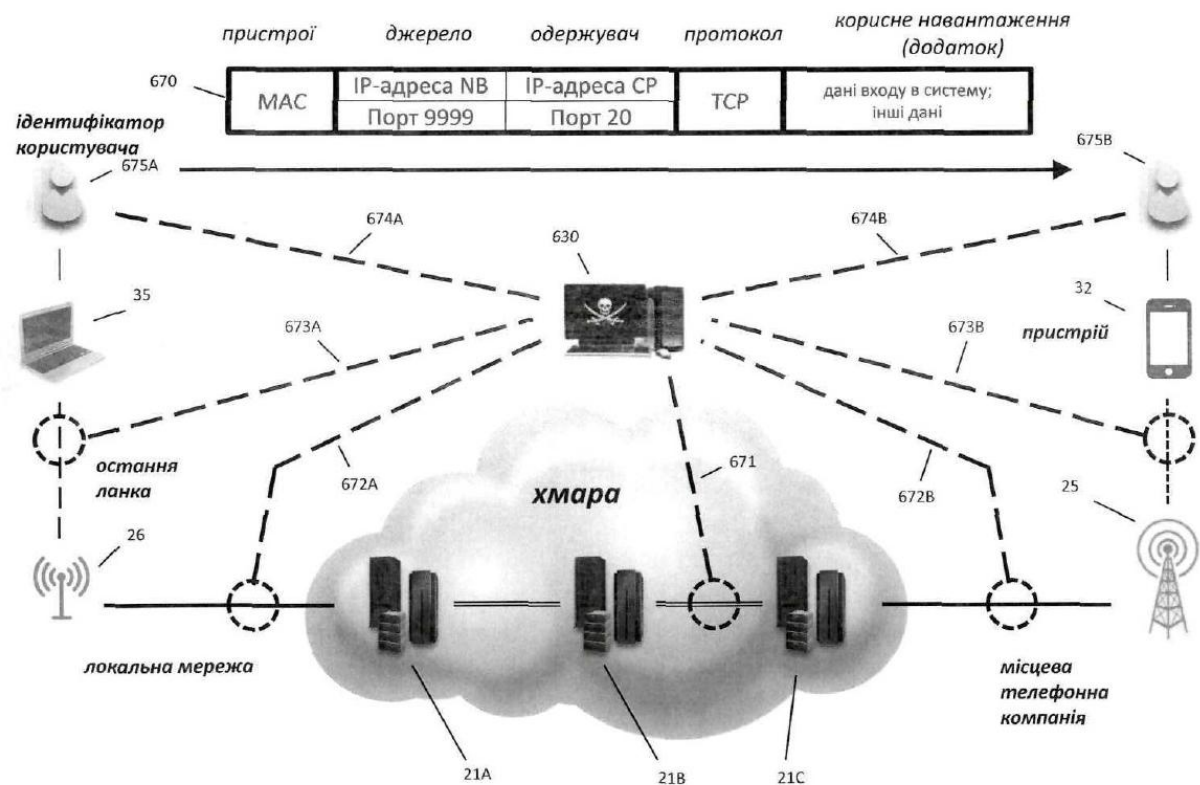


Рисунок 44

Виявлення на основі даних опитування портів

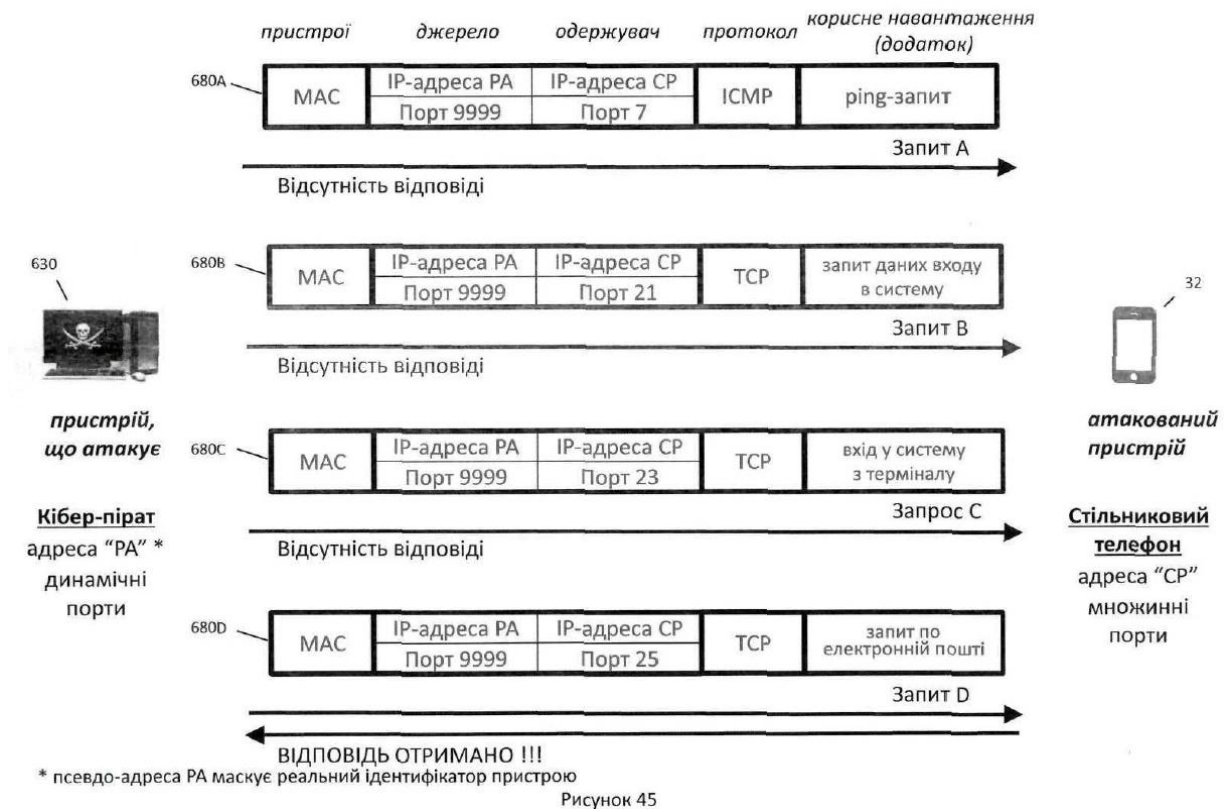


Рисунок 45

Захоплення IP-пакетів

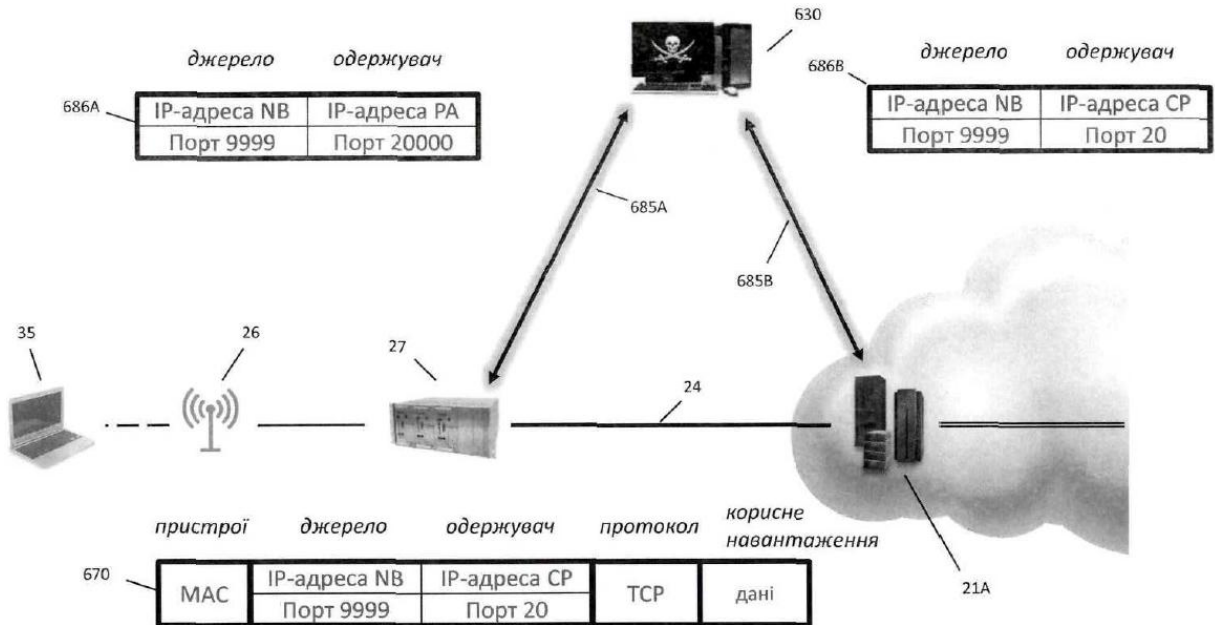


Рисунок 46

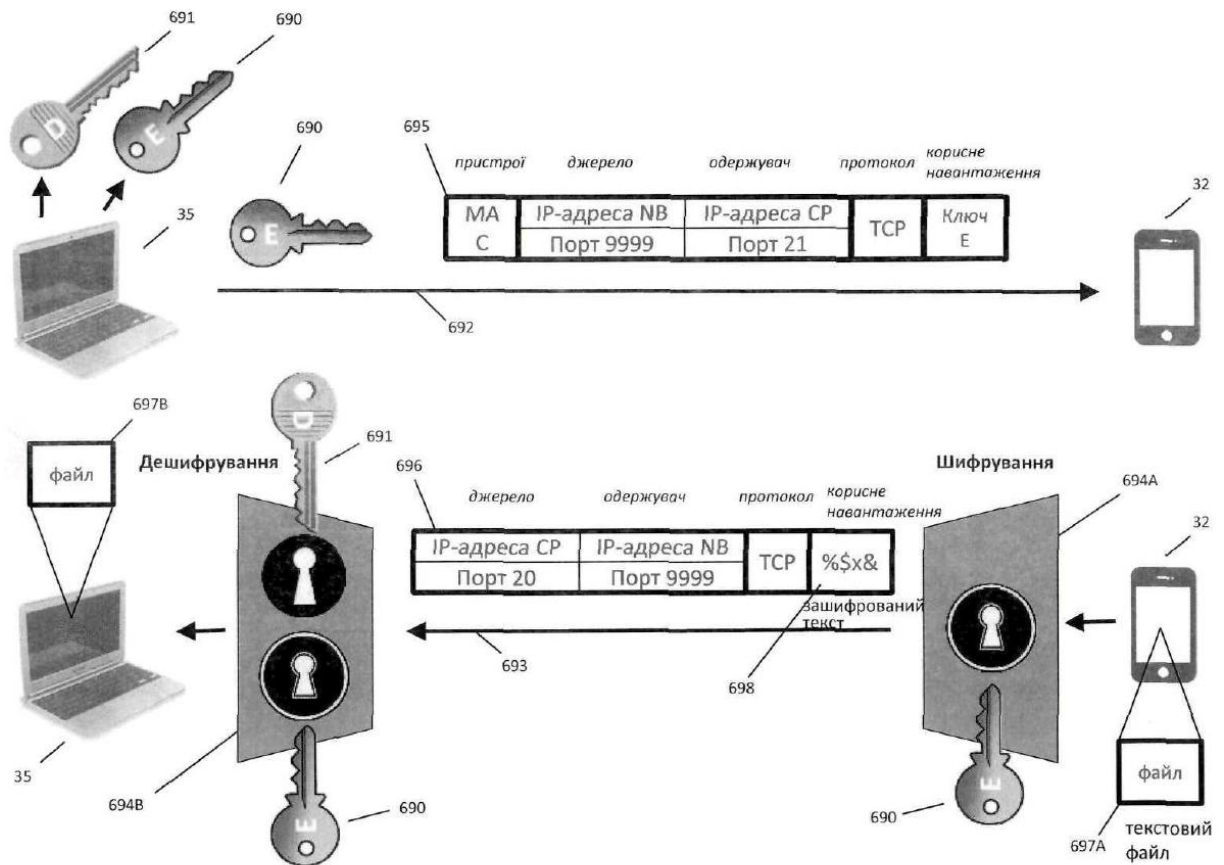


Рисунок 47

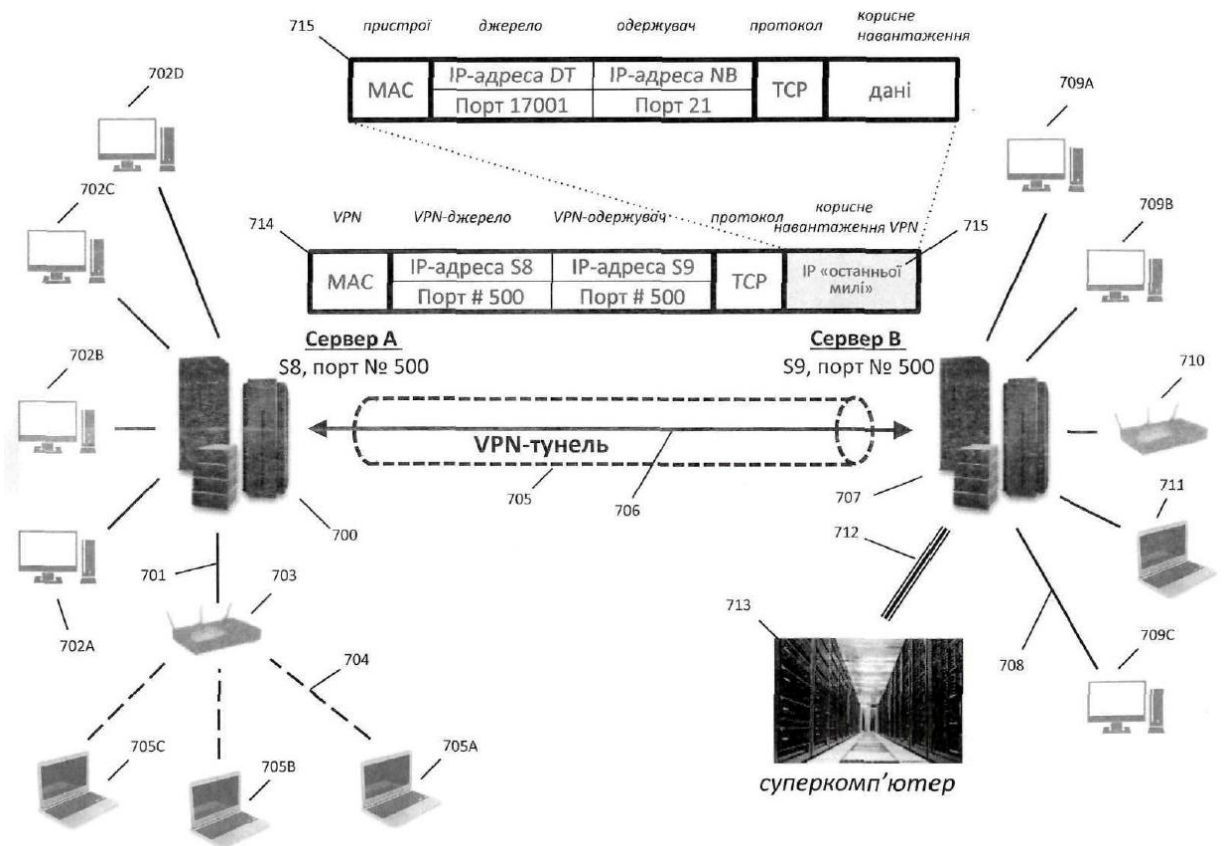


Рисунок 48А

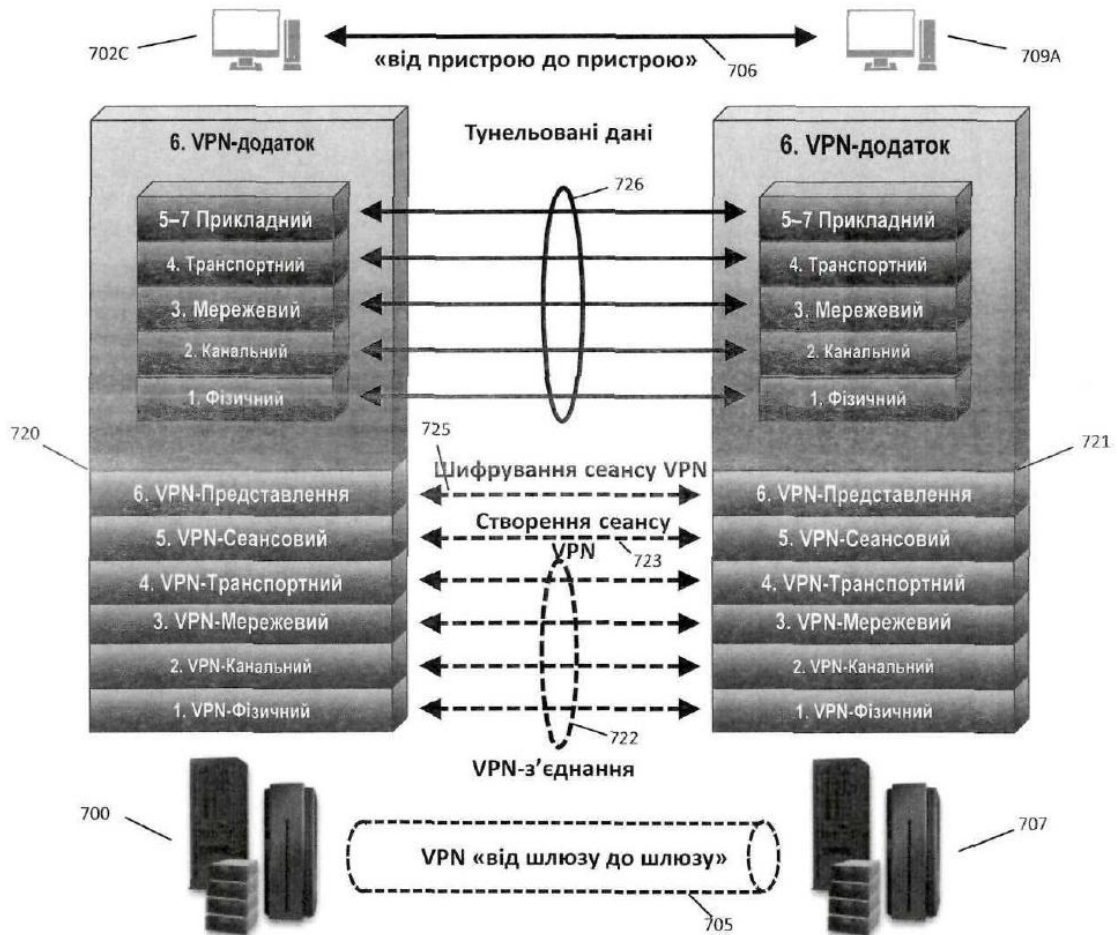


Рисунок 48В

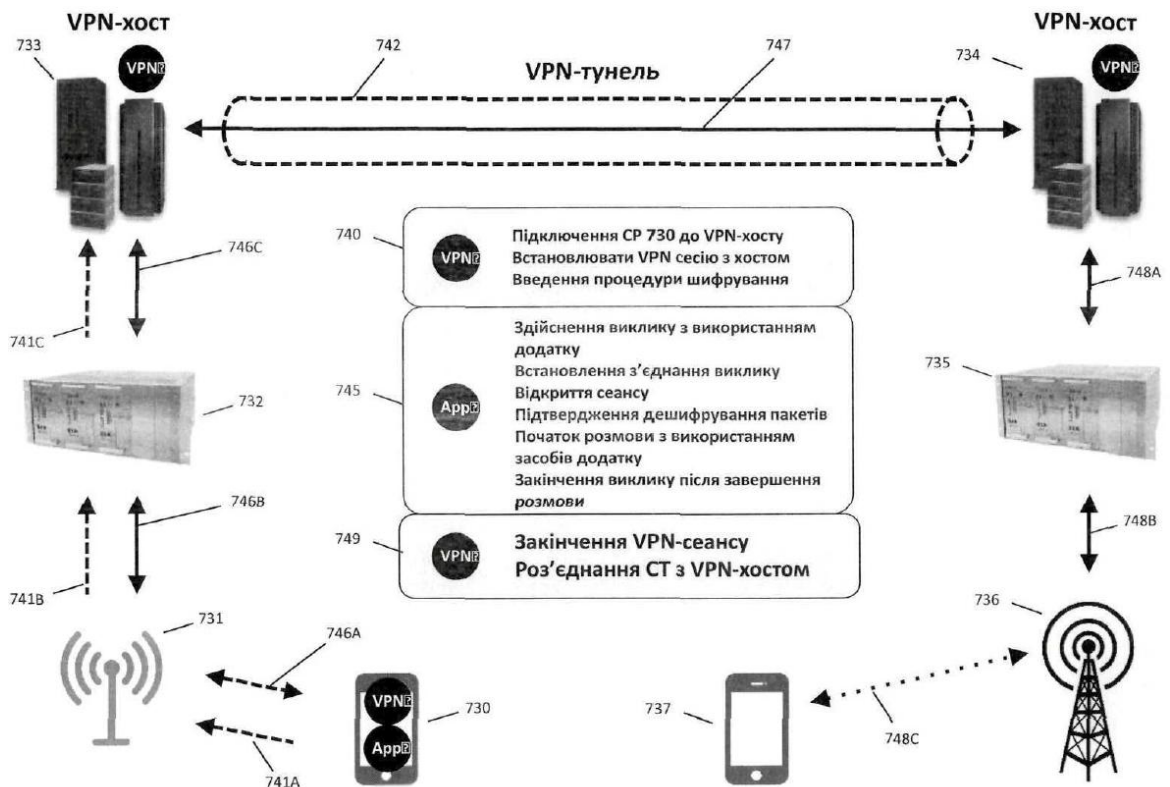


Рисунок 48С

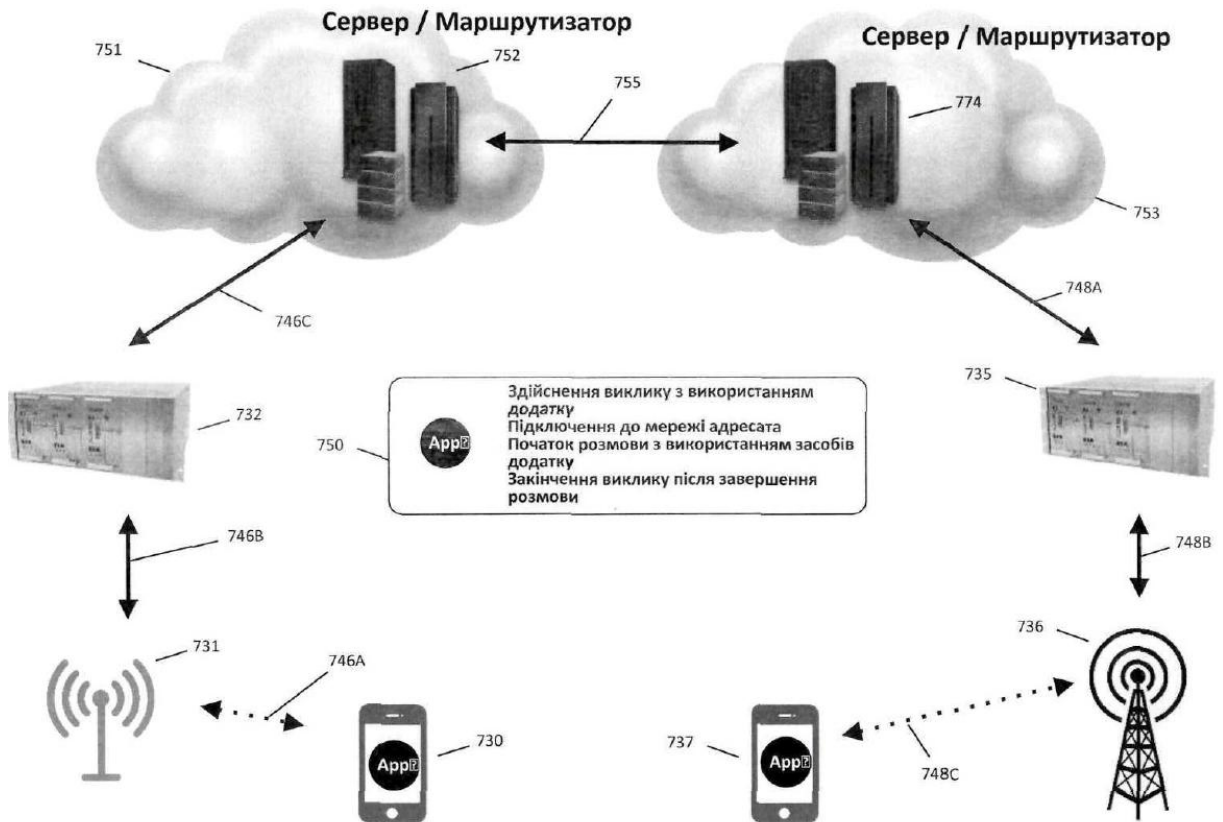


Рисунок 49А

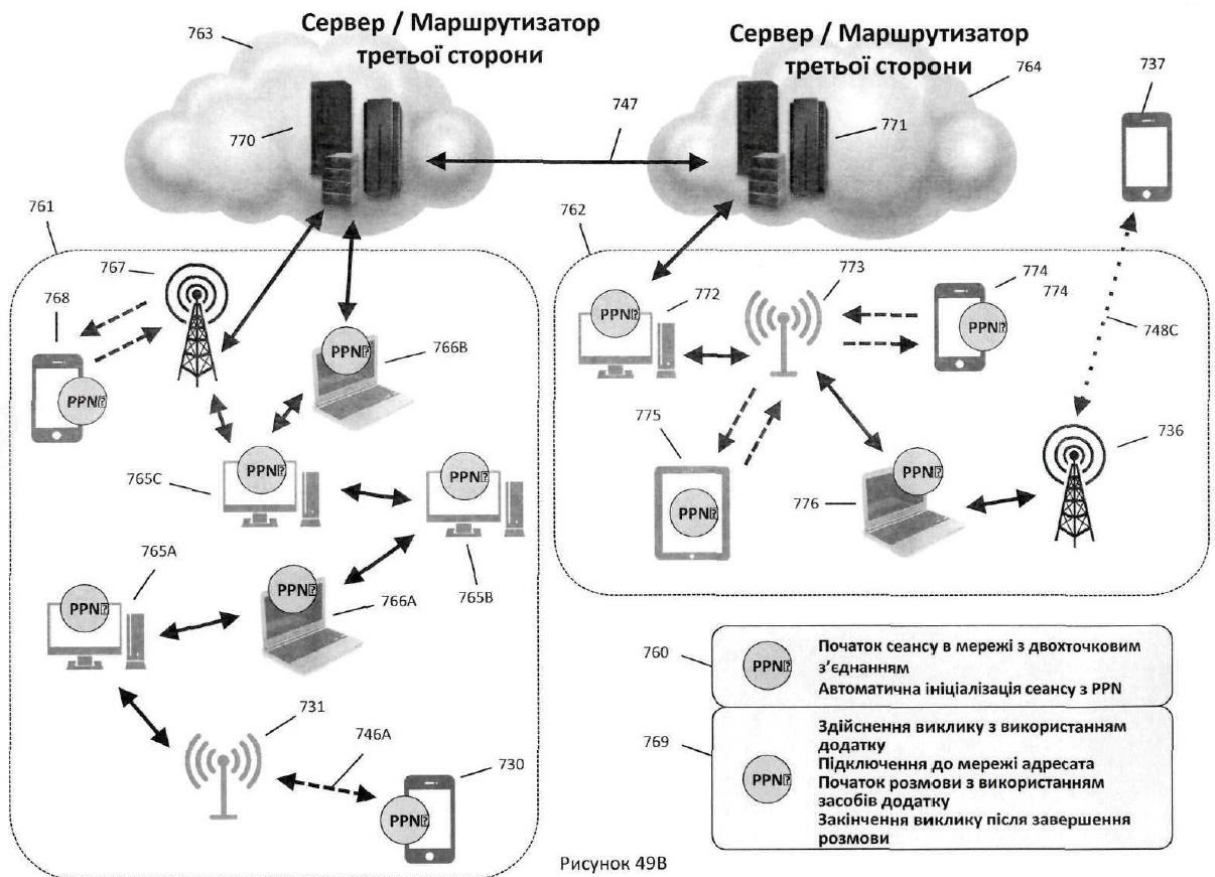


Рисунок 49В

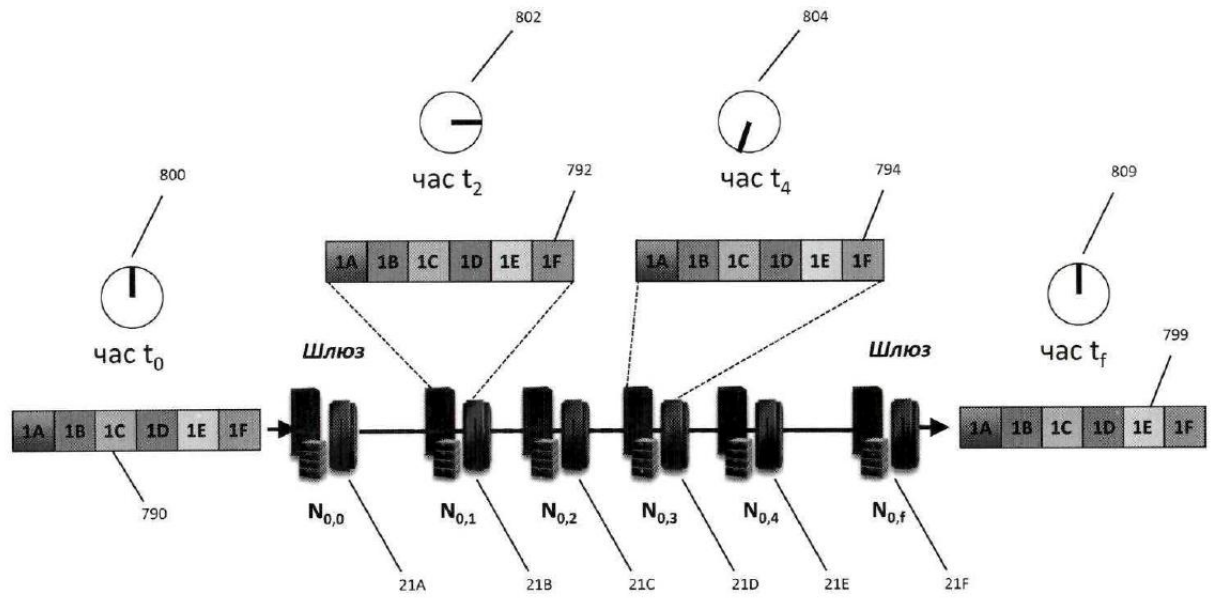


Рисунок 50

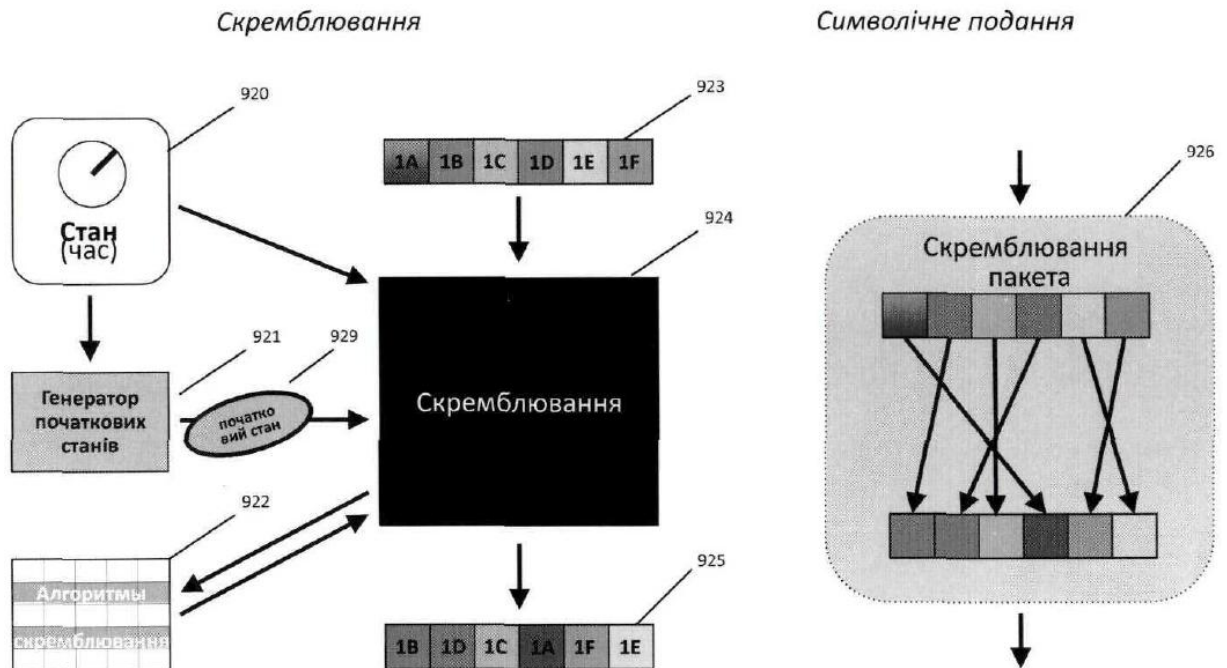


Рисунок 51А

Дескремблювання пакета

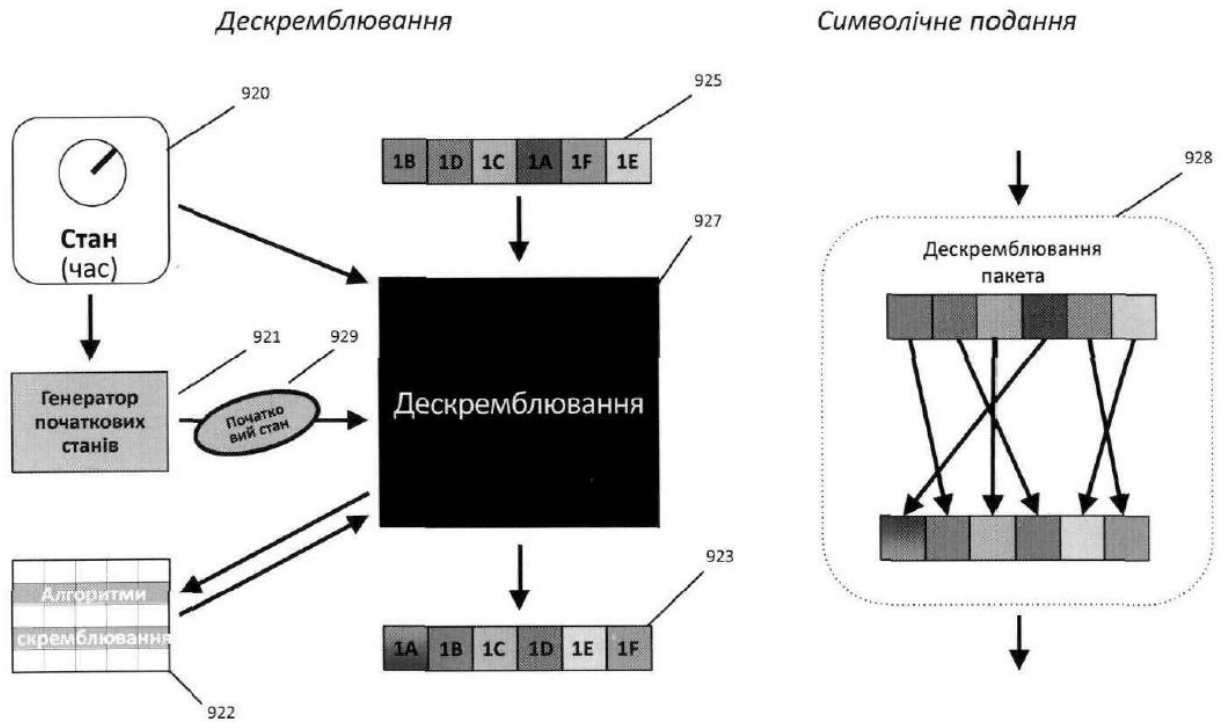


Рисунок 51В

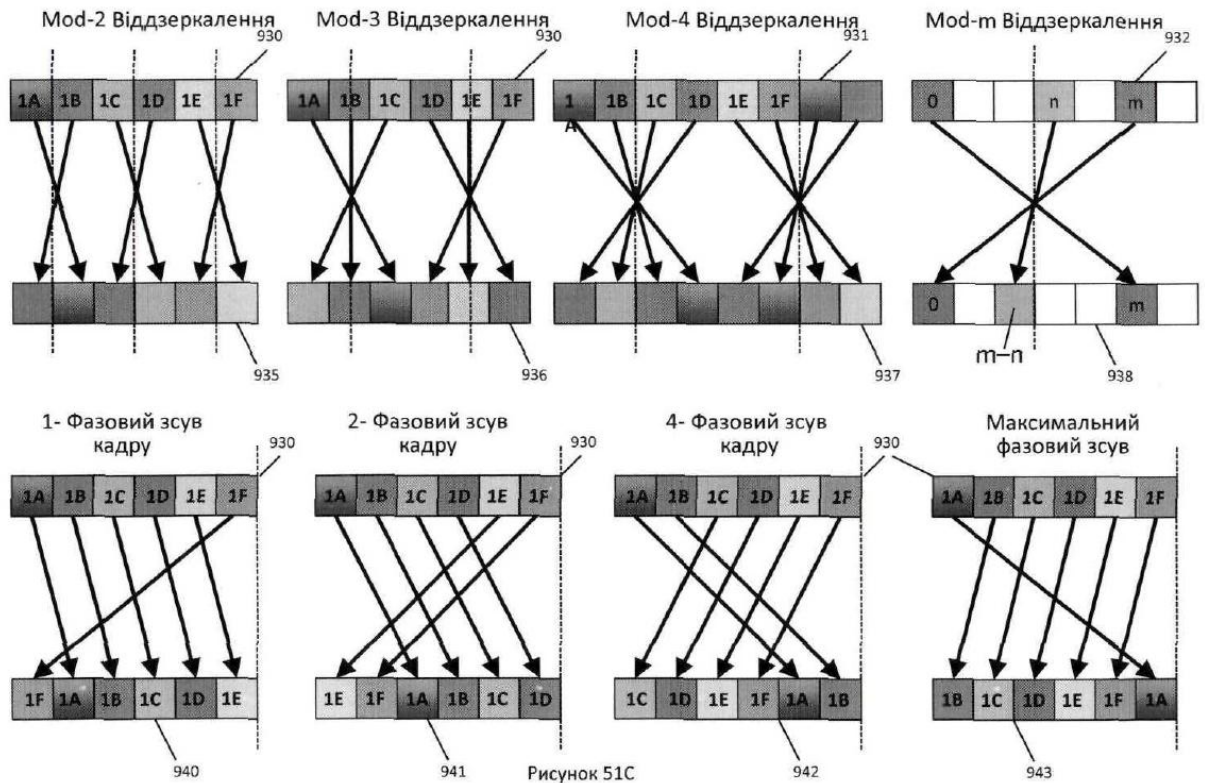


Рисунок 51С

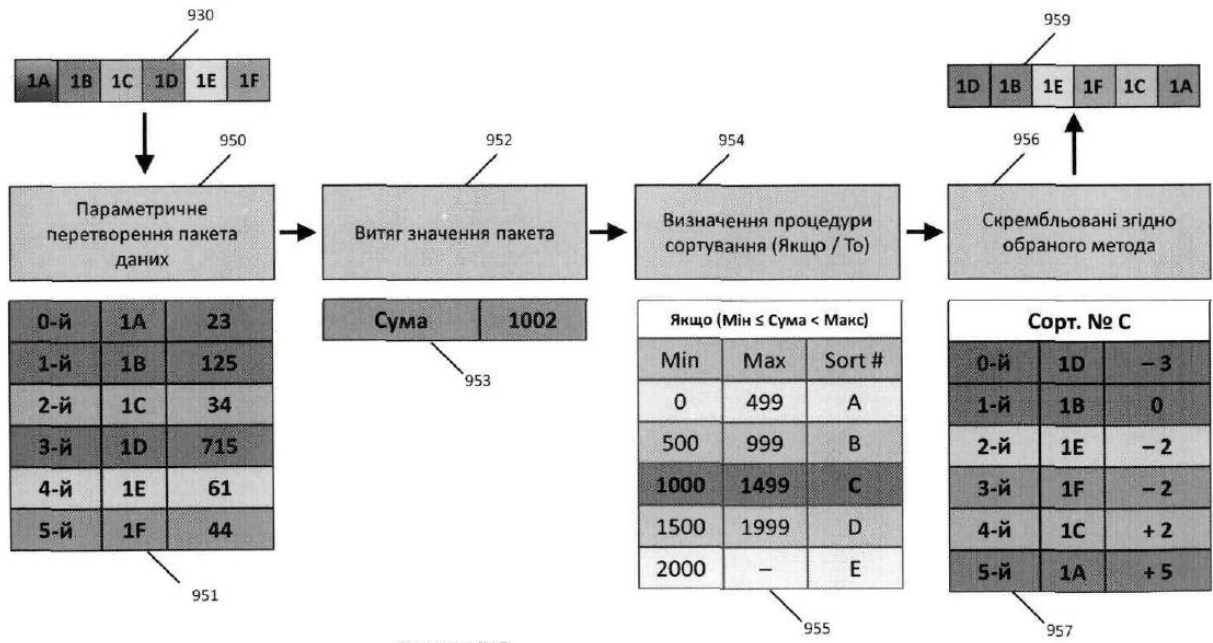


Рисунок 51D

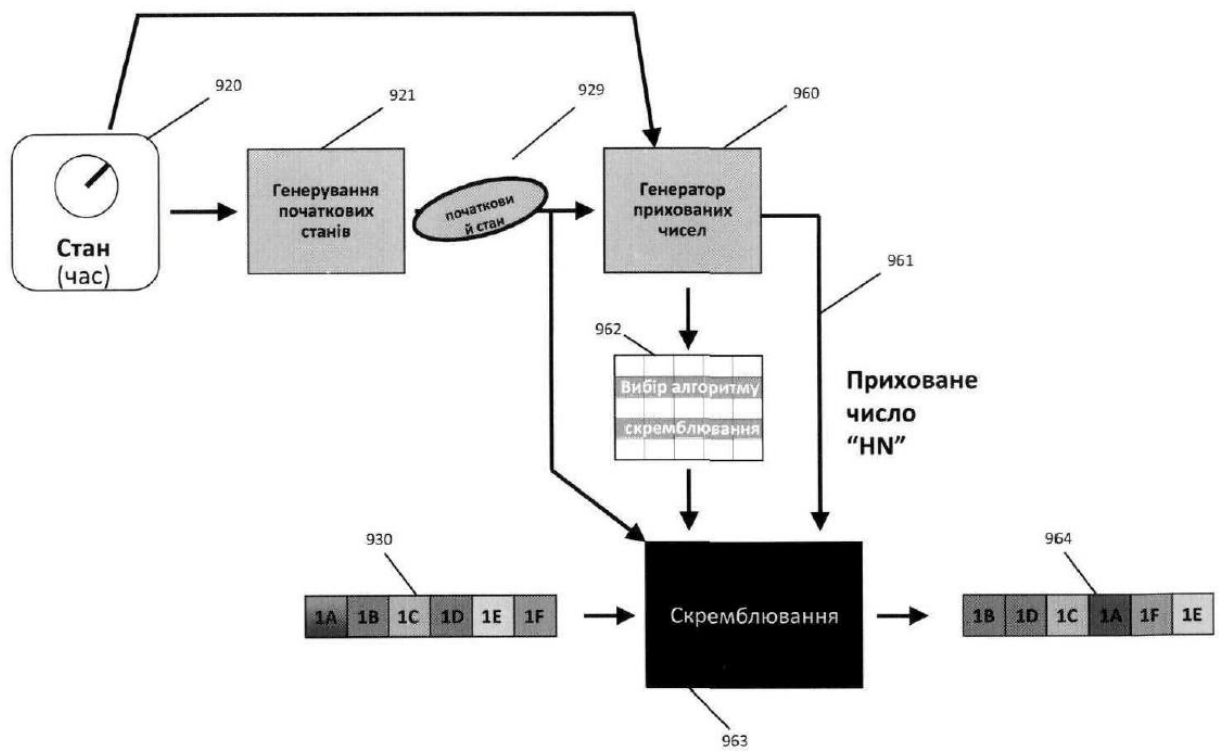


Рисунок 51E

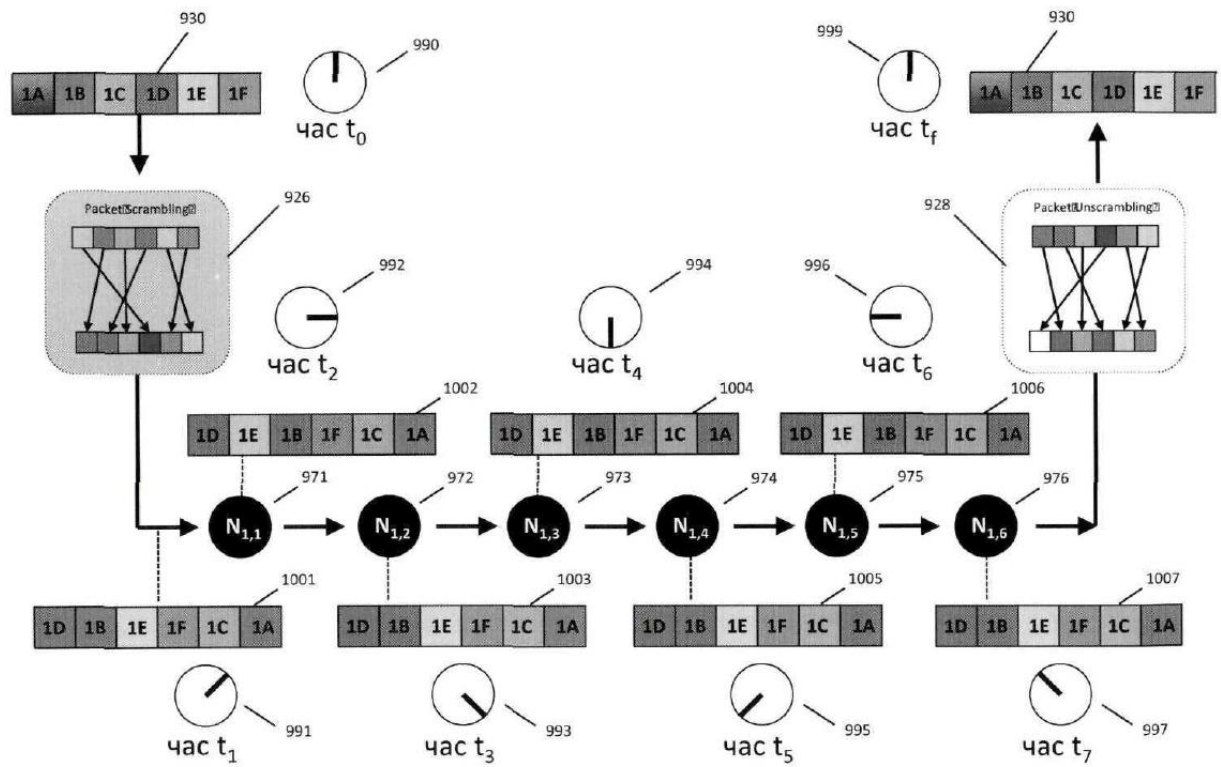


Рисунок 51F

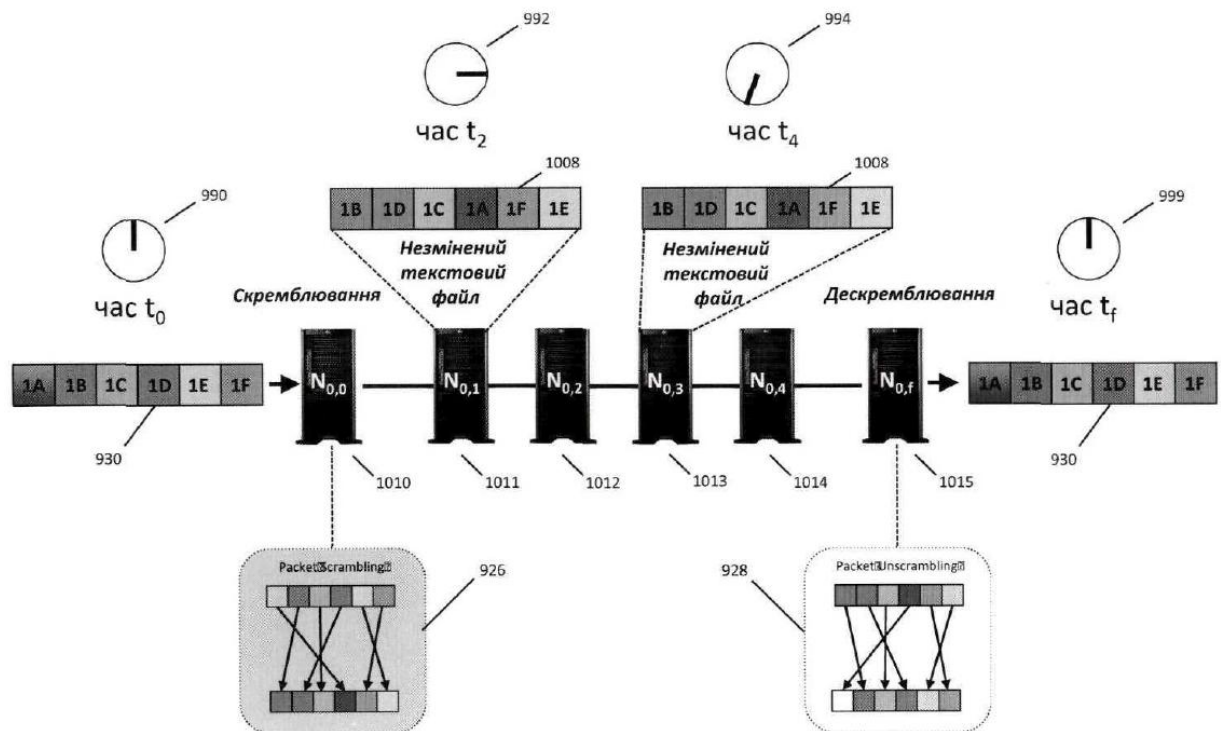


Рисунок 52

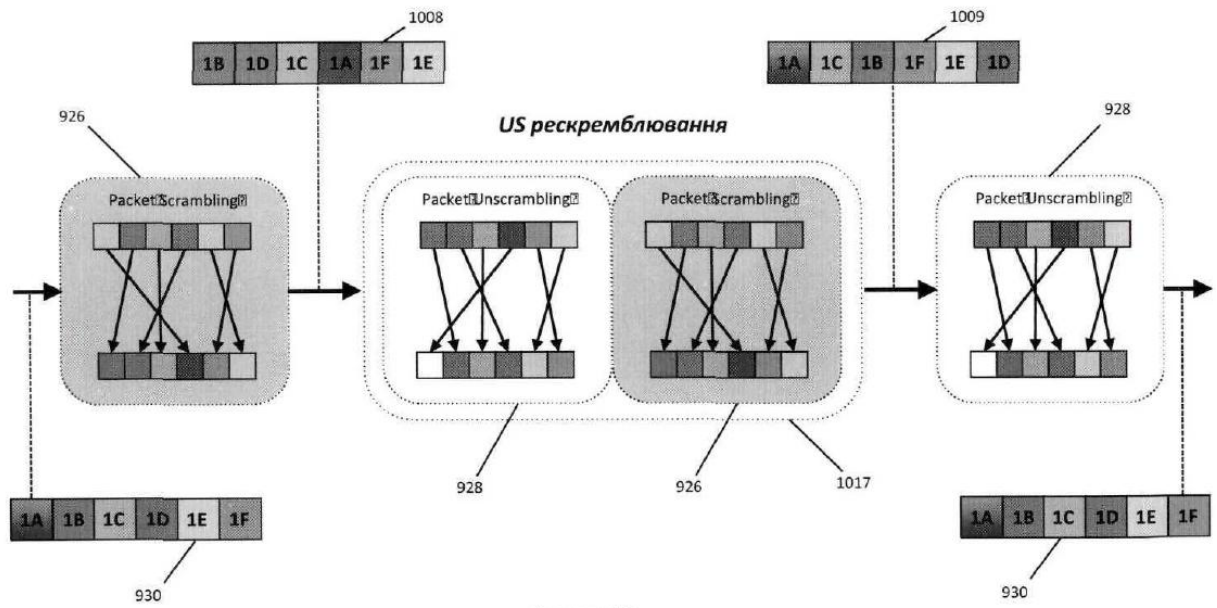


Рисунок 53

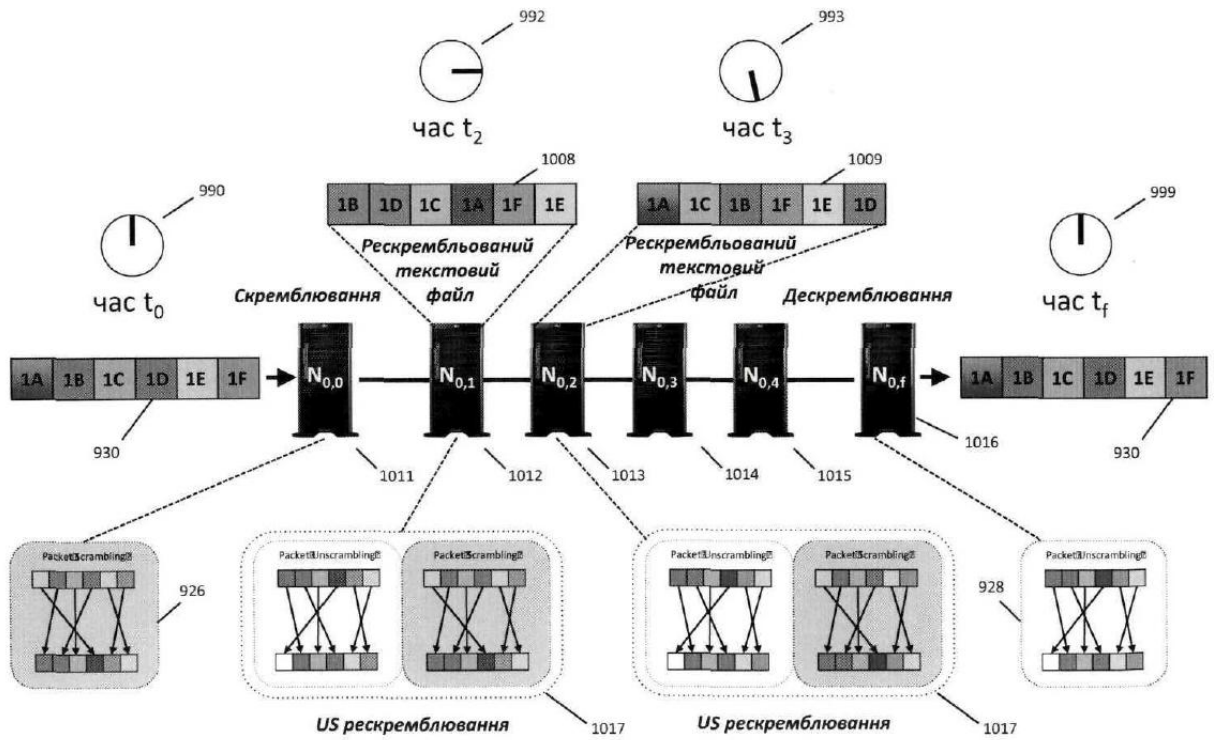


Рисунок 54

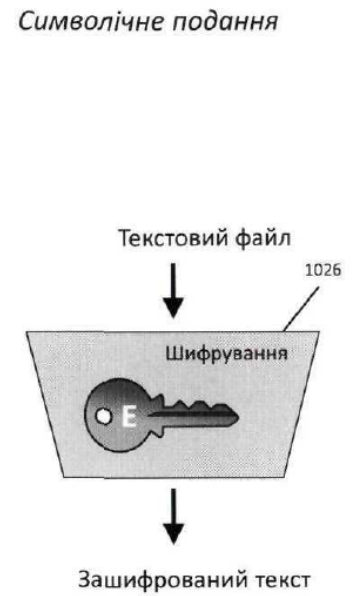
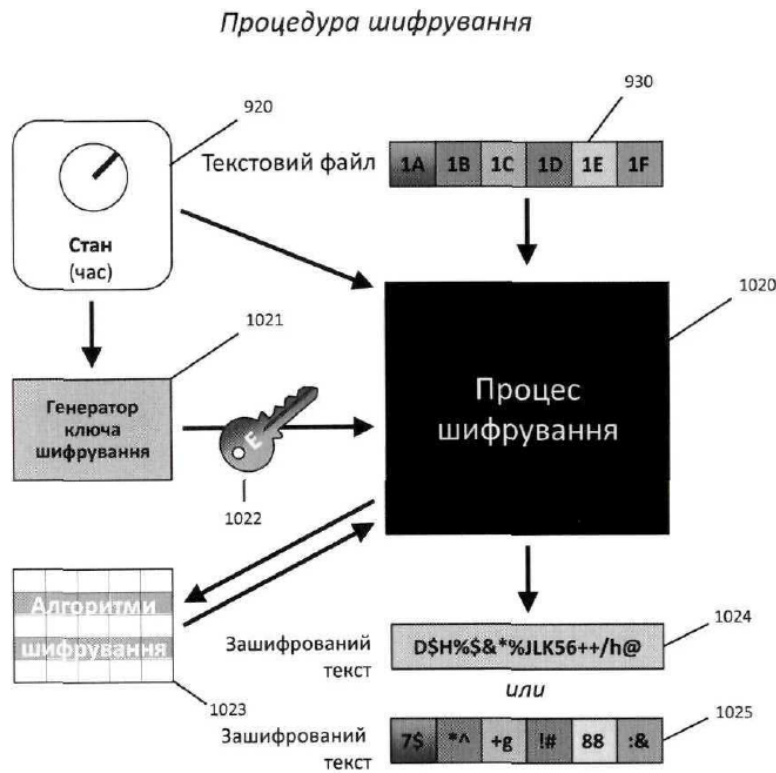


Рисунок 55A

Дешифрування пакетних даних

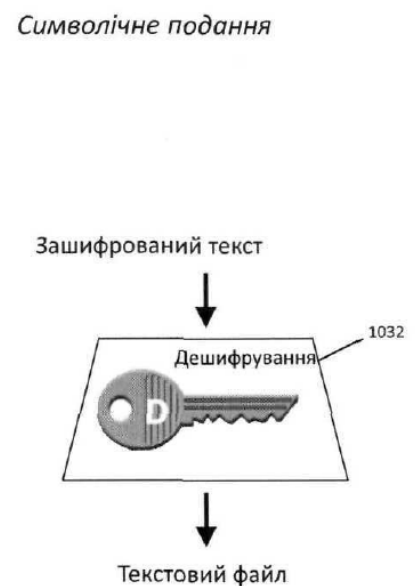
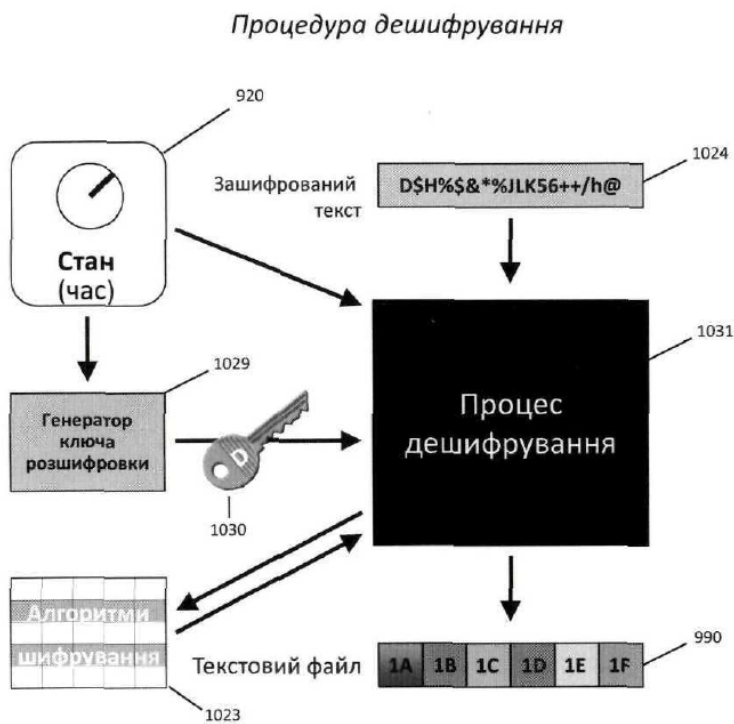


Рисунок 55B

Шифрування скрембльованого пакета даних

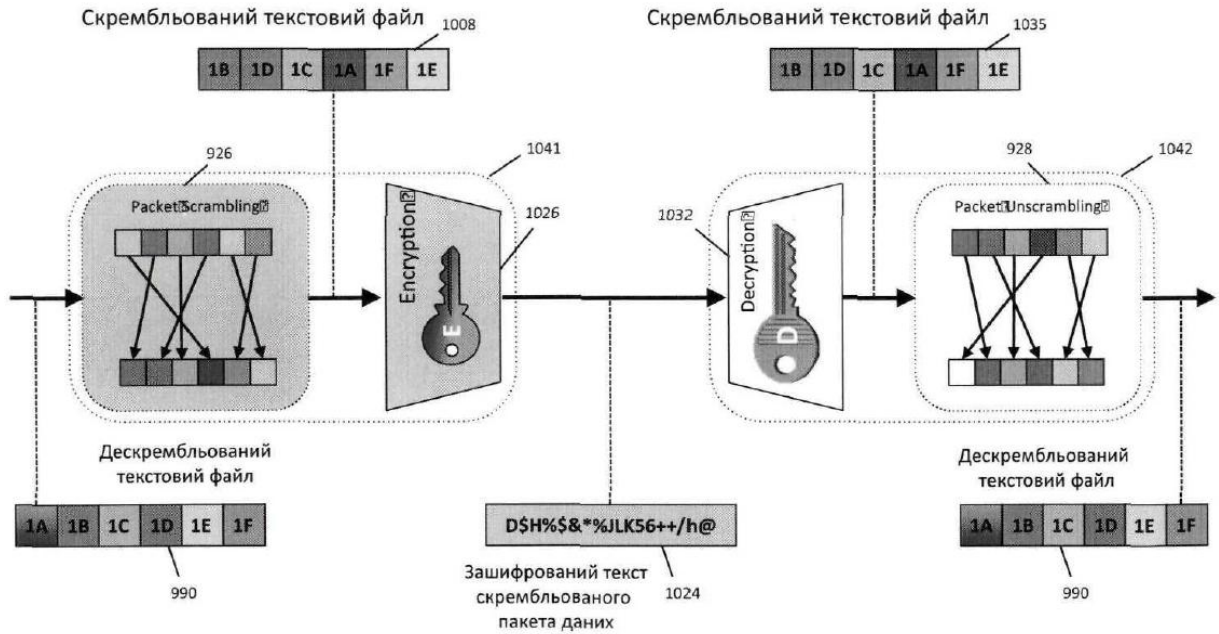


Рисунок 56

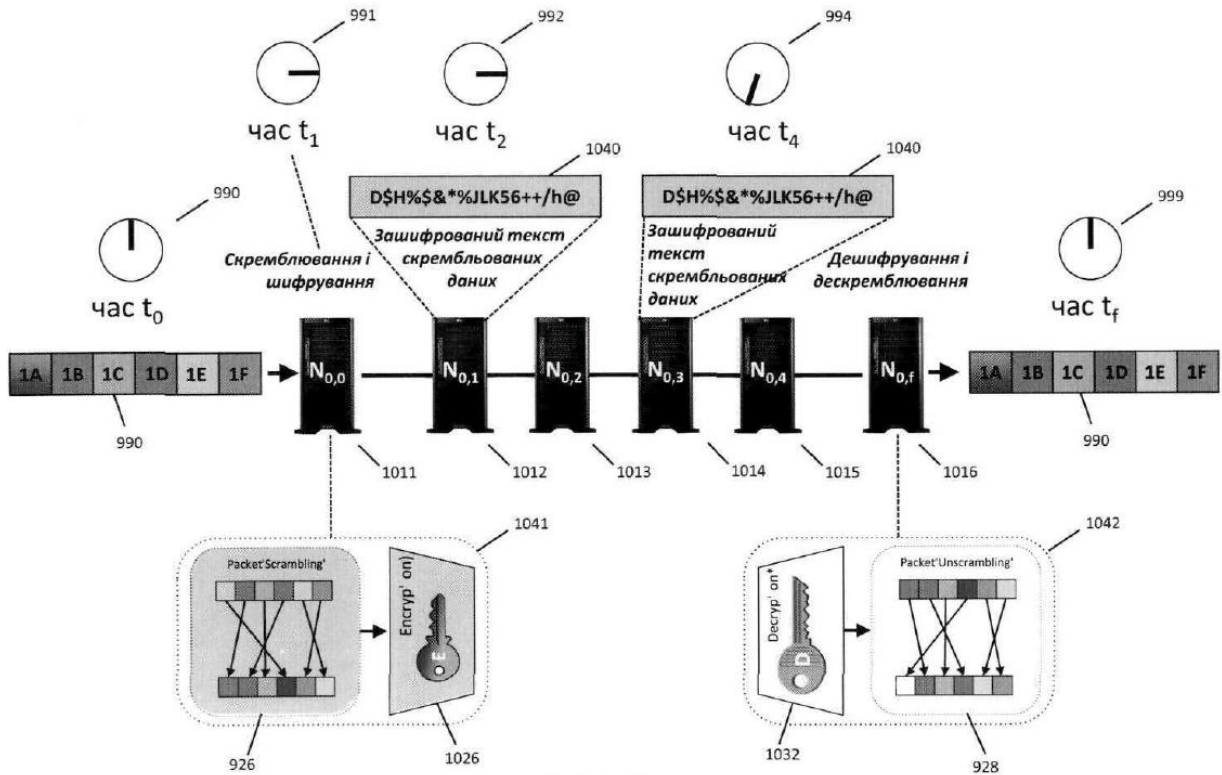


Рисунок 57

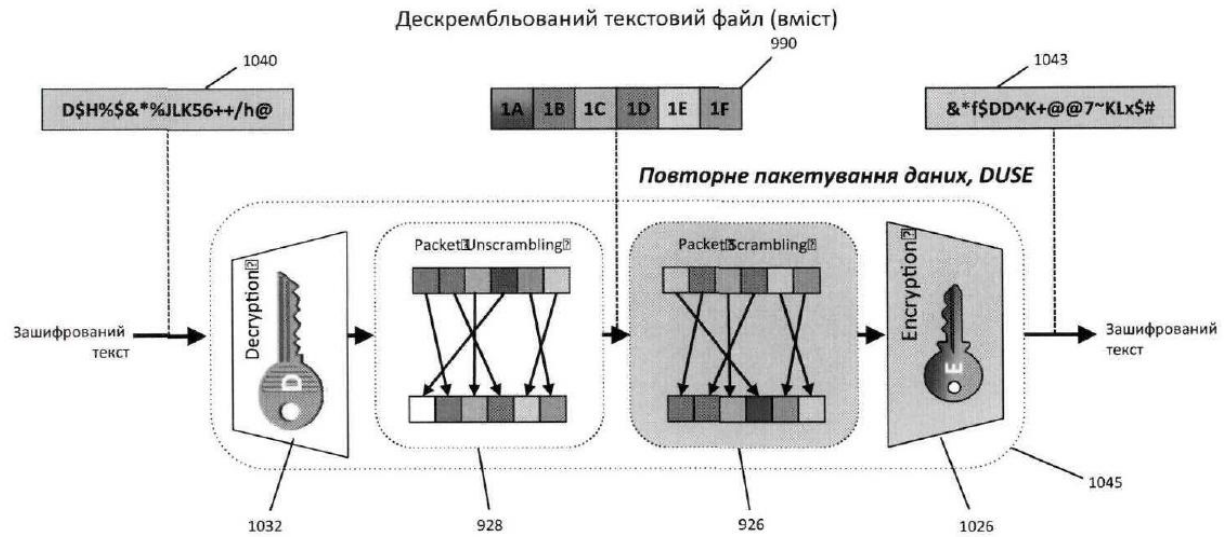


Рисунок 58

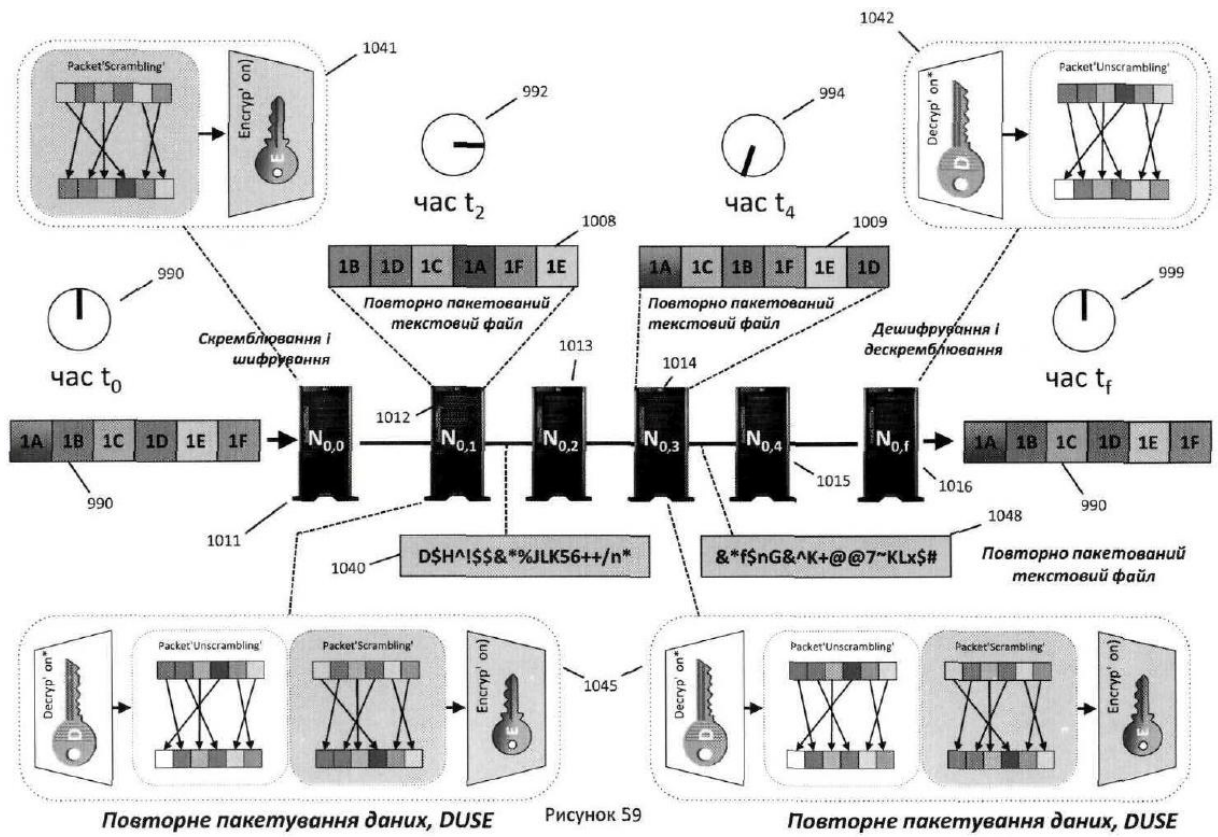


Рисунок 59

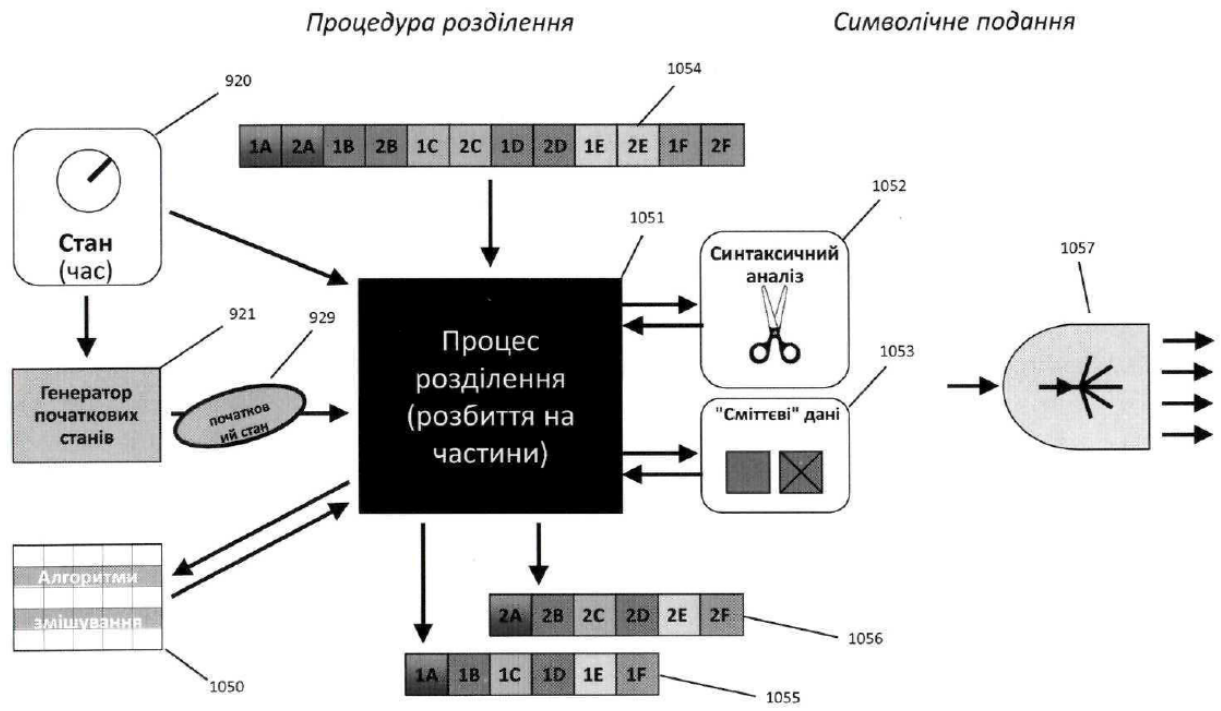


Рисунок 60А

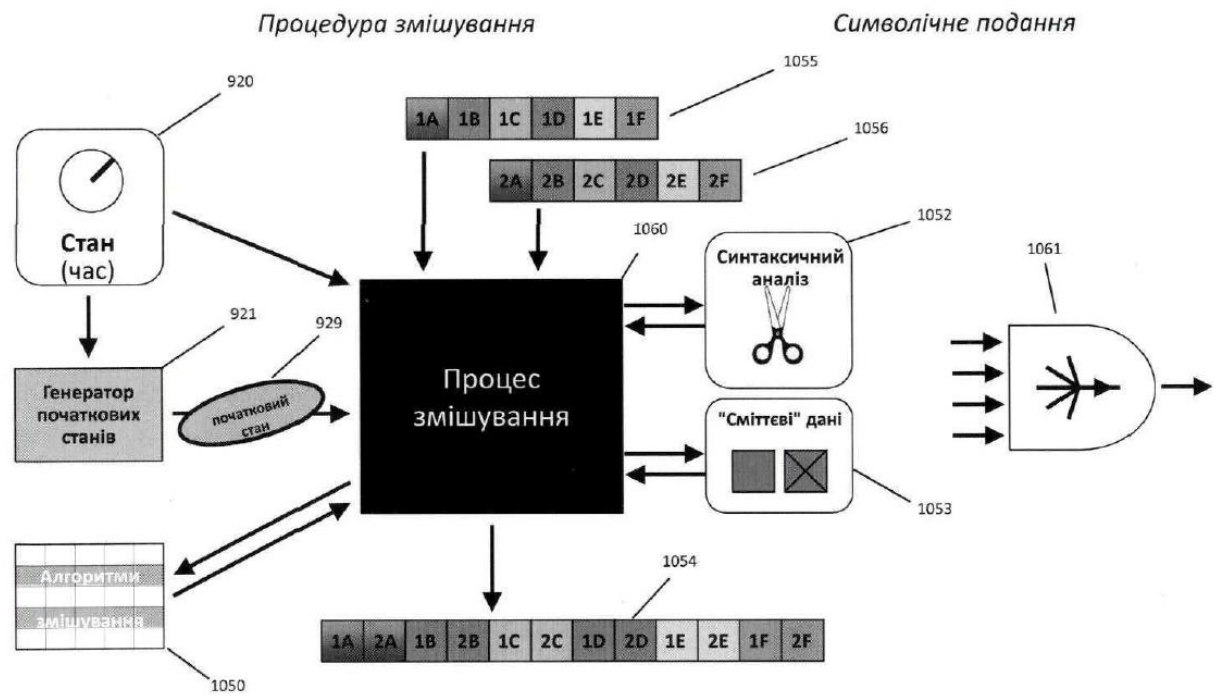


Рисунок 60В

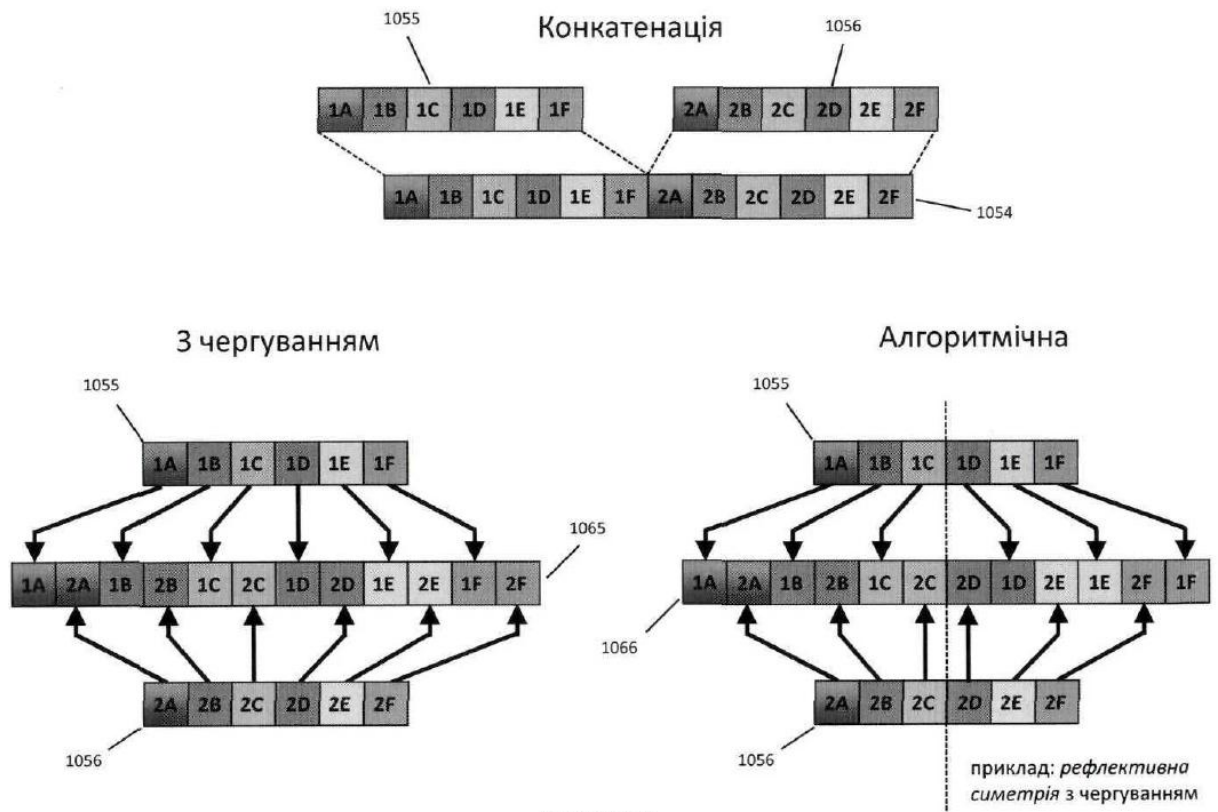


Рисунок 61А

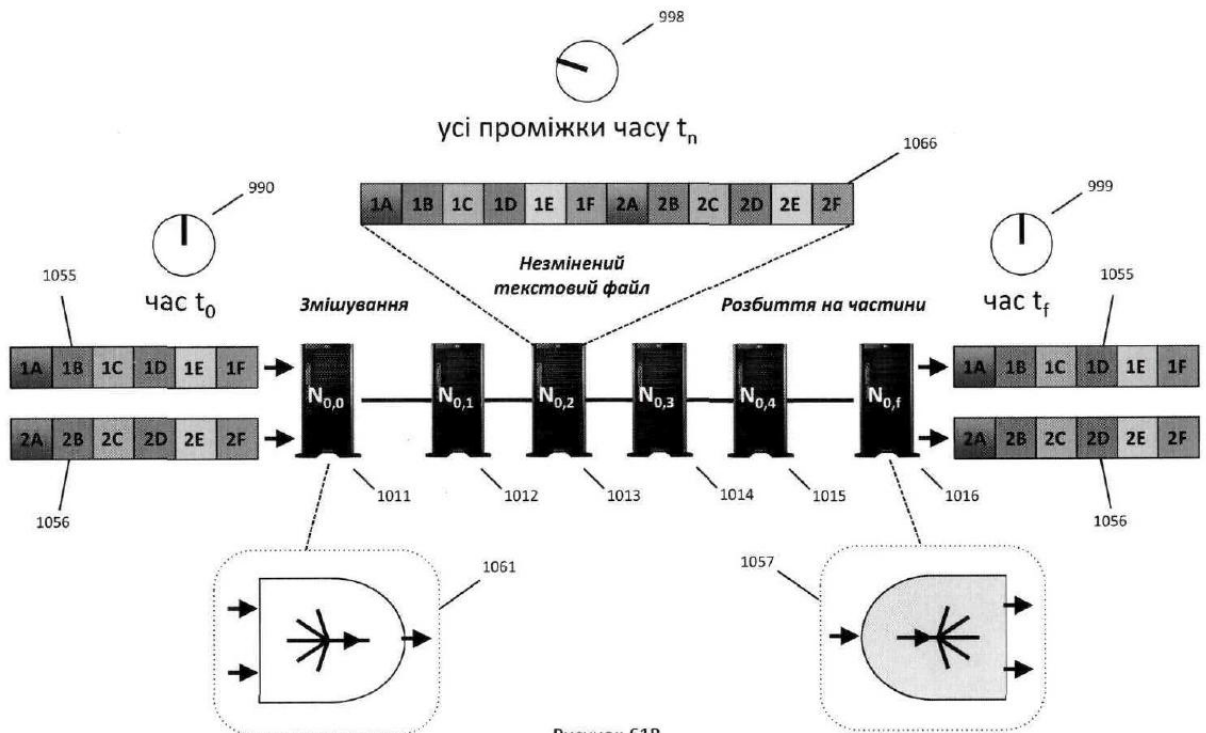


Рисунок 61В

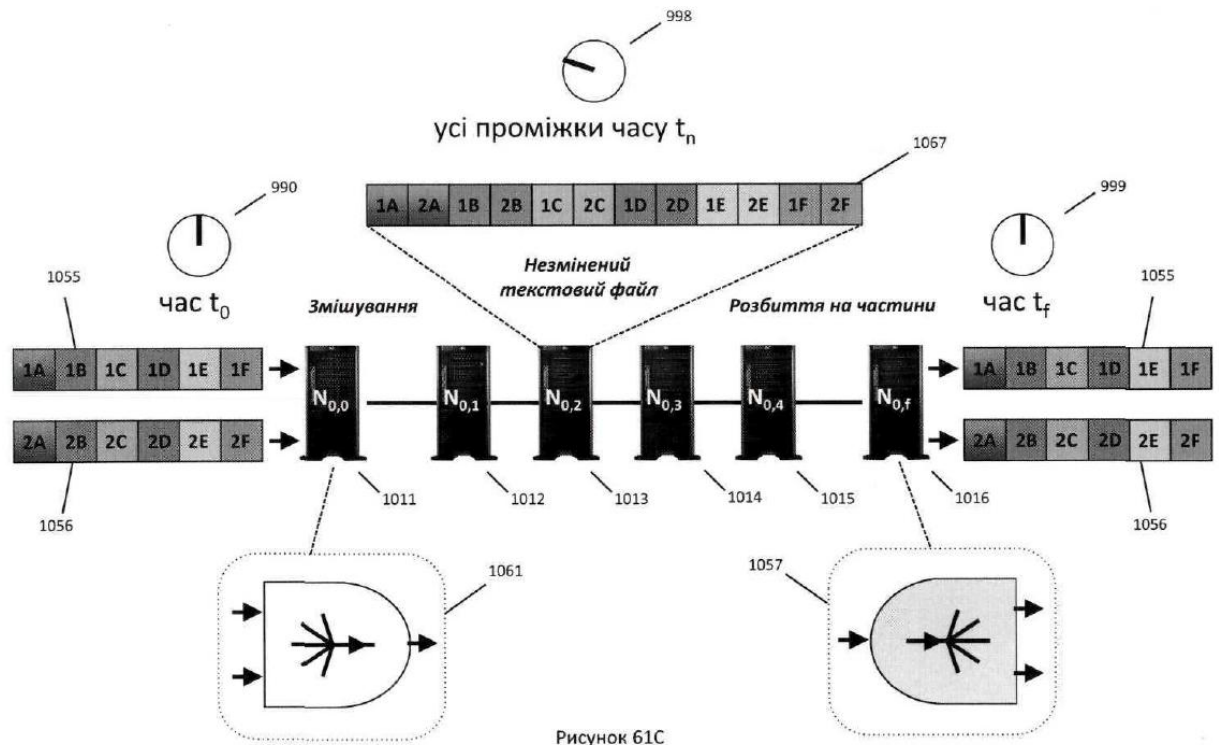


Рисунок 61C

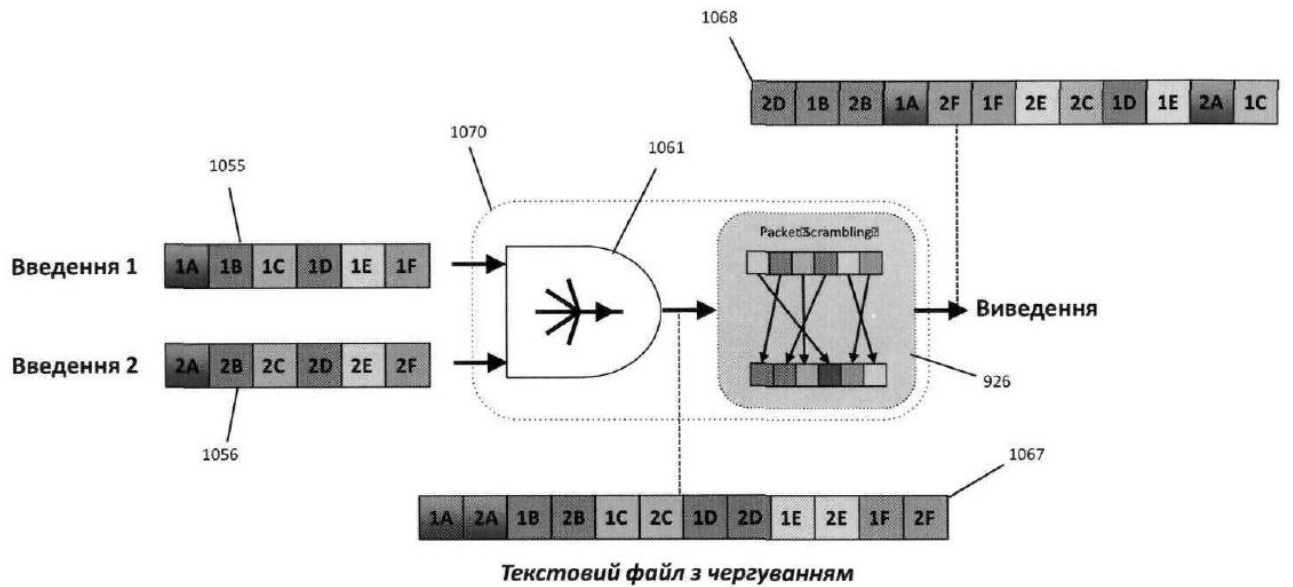


Рисунок 62A

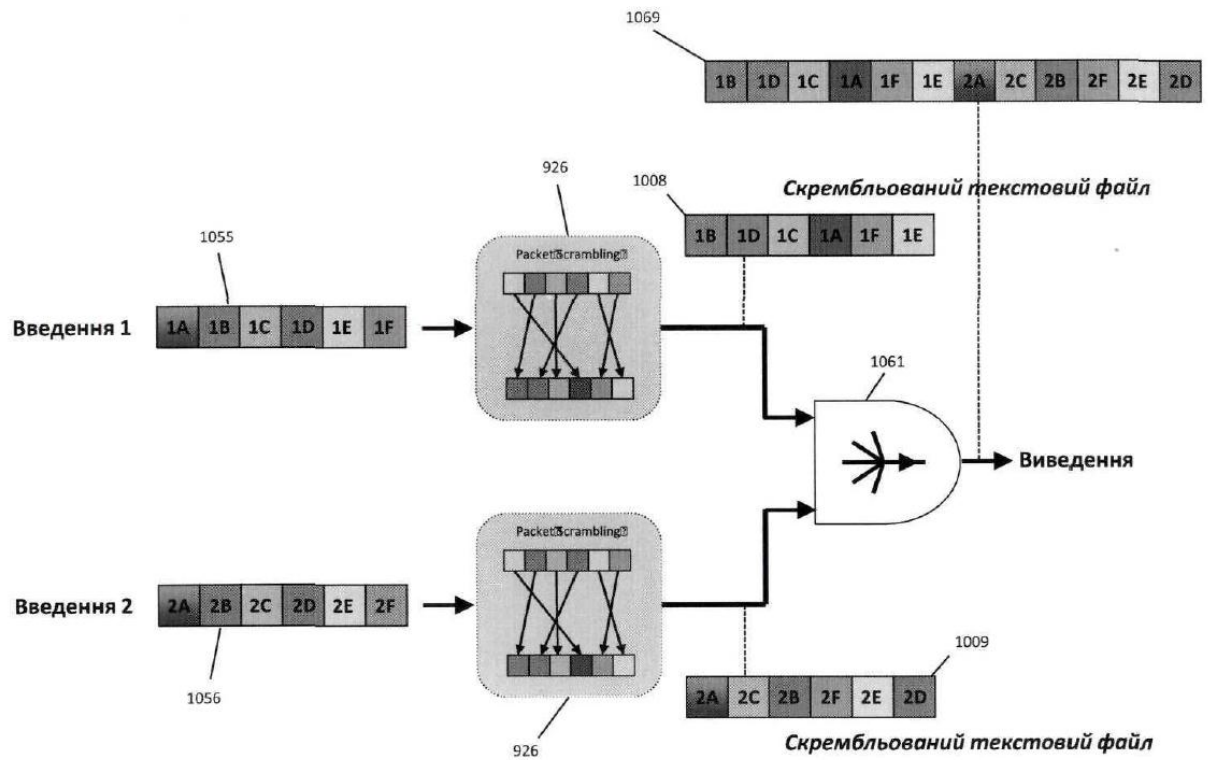


Рисунок 62В

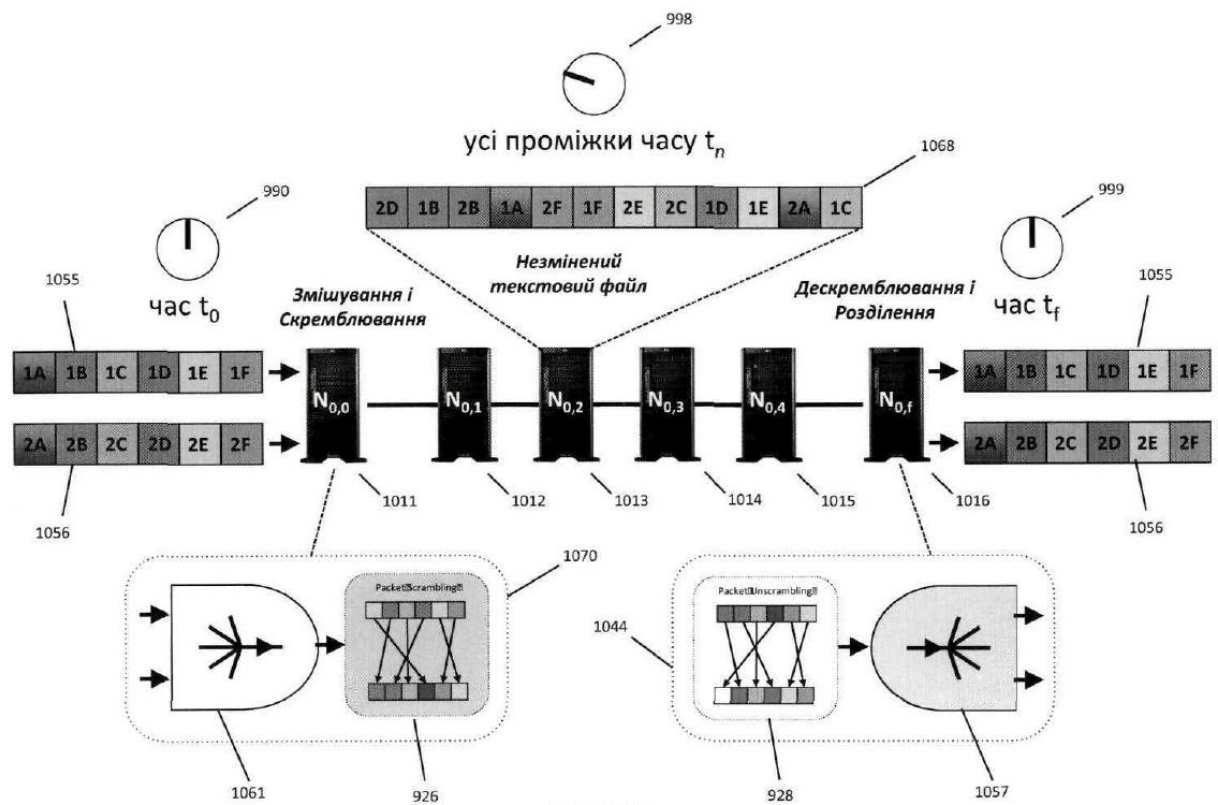


Рисунок 63

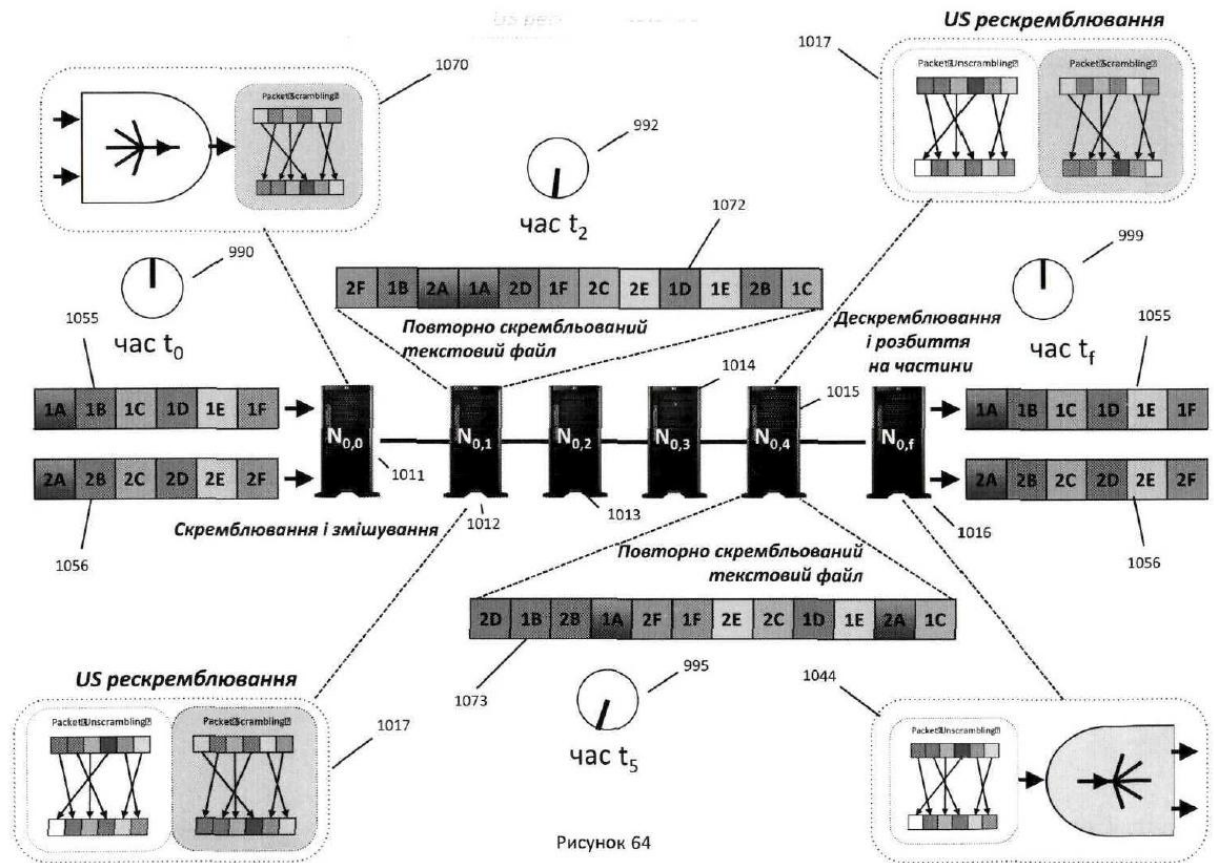


Рисунок 64

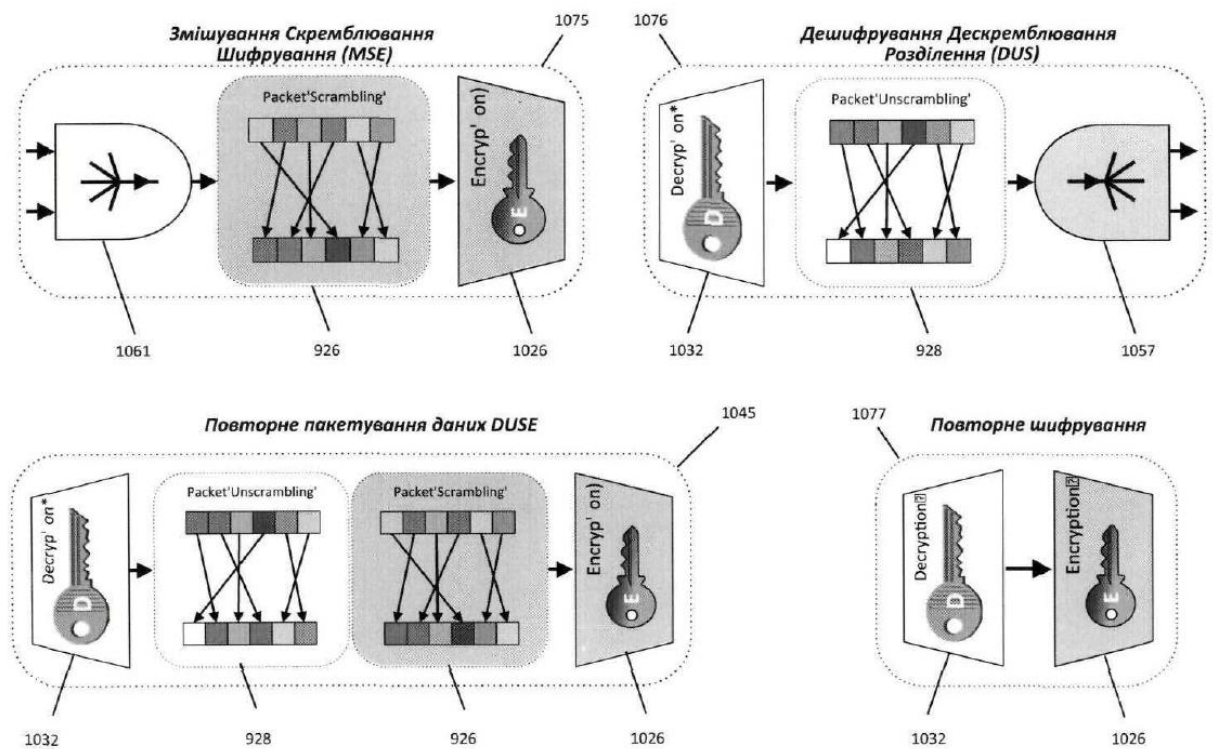
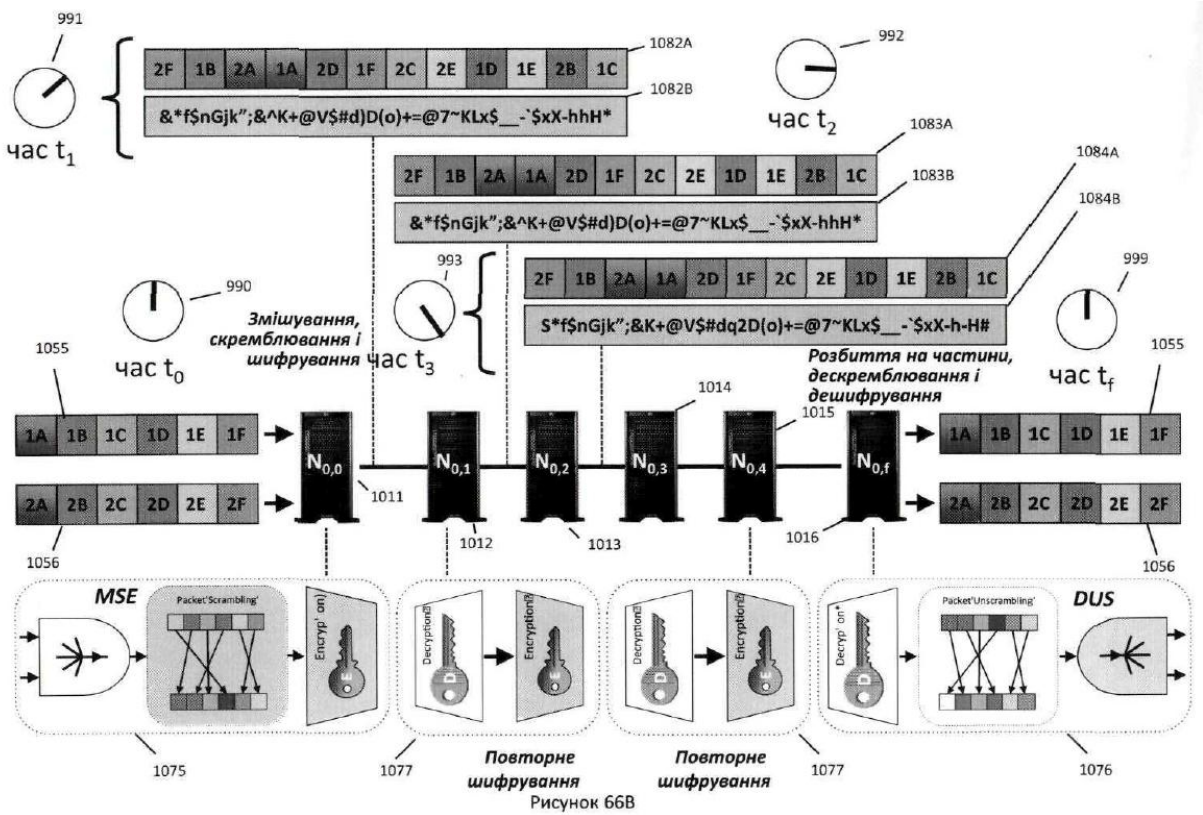
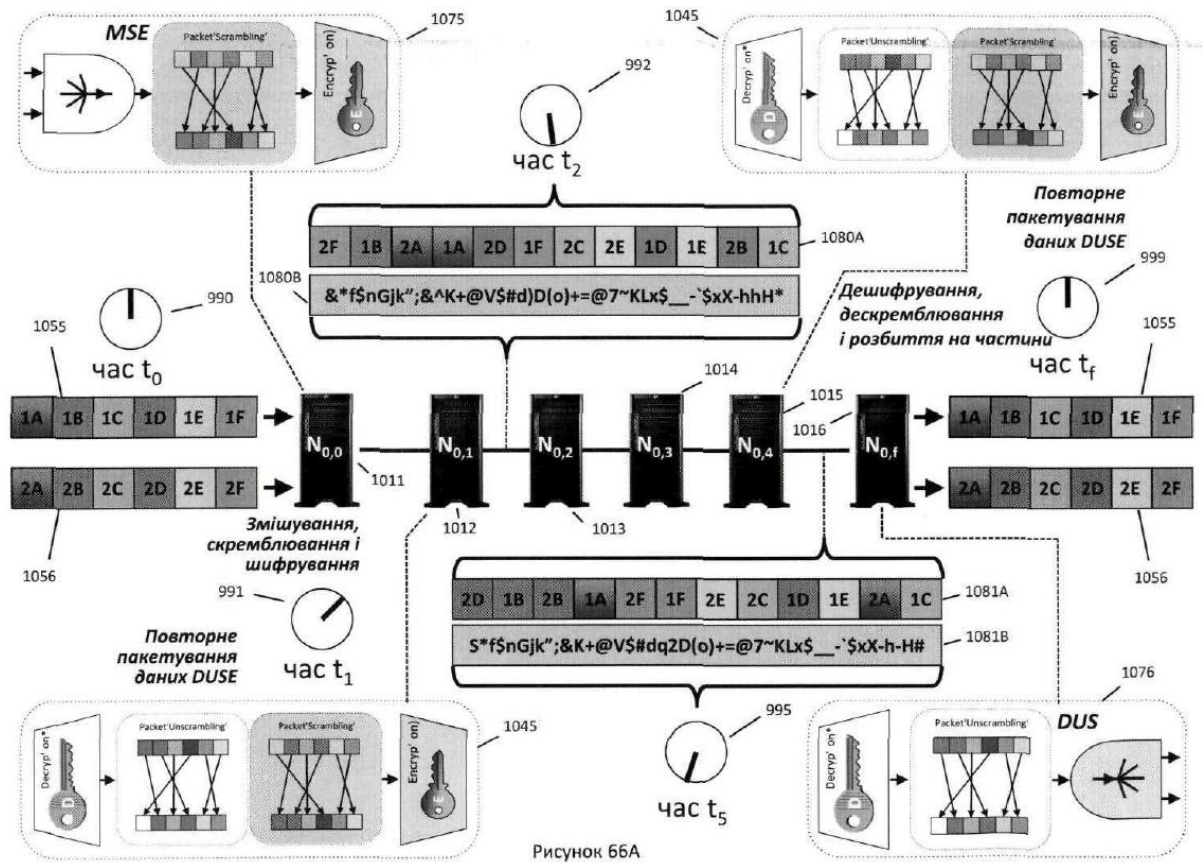


Рисунок 65



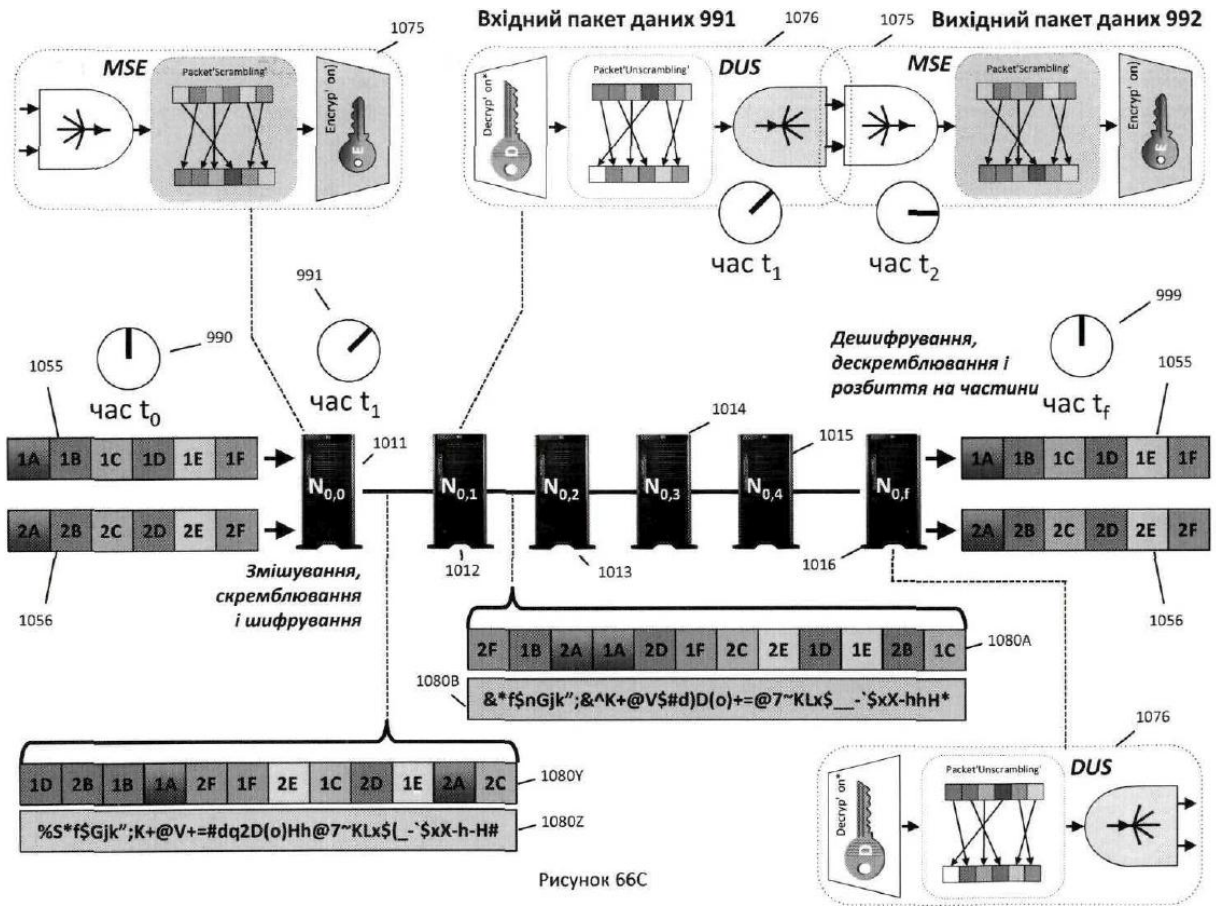


Рисунок 66C

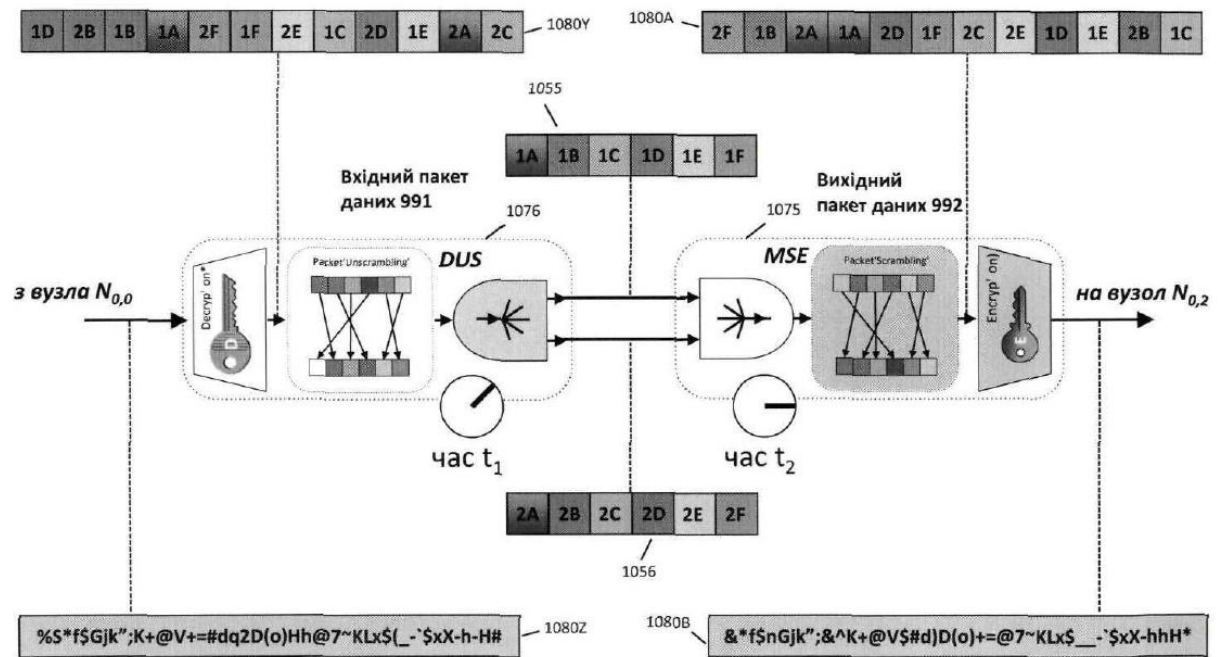
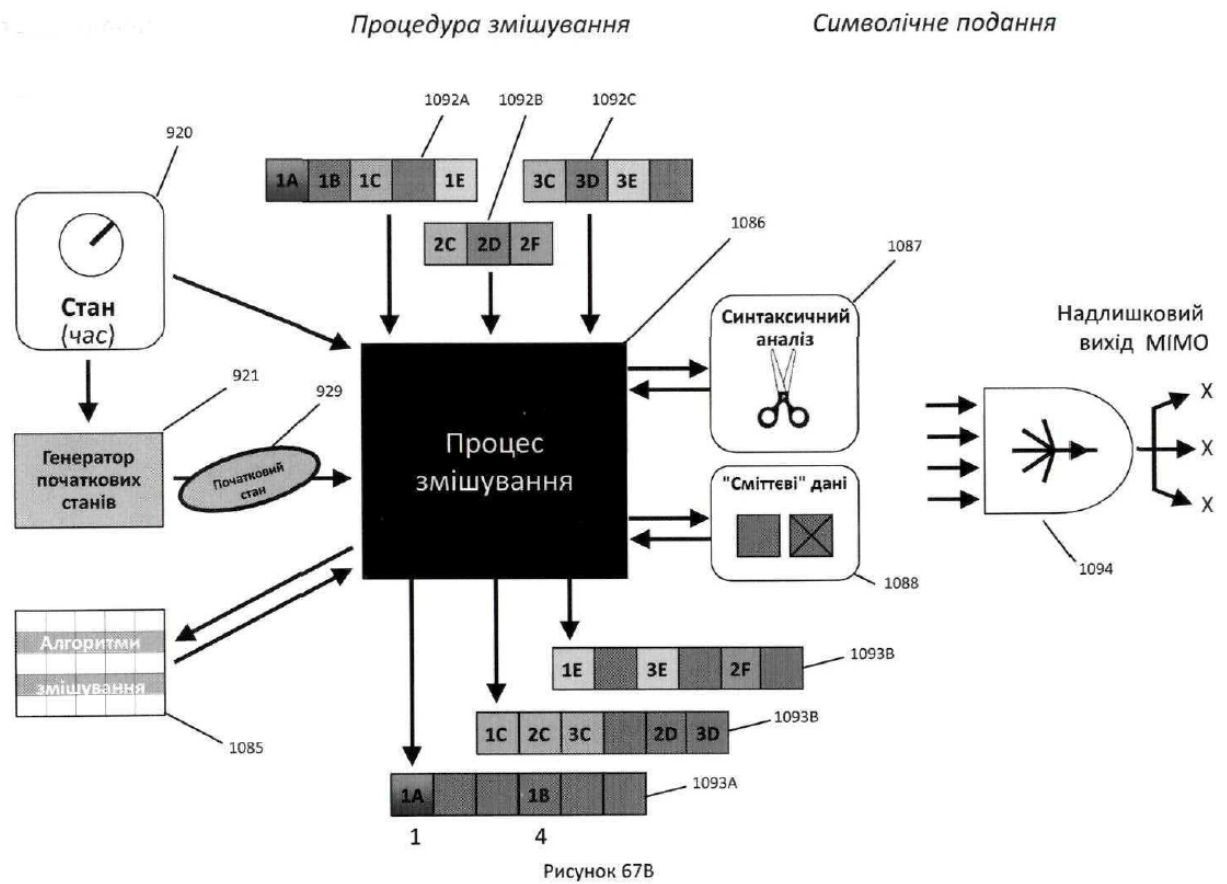
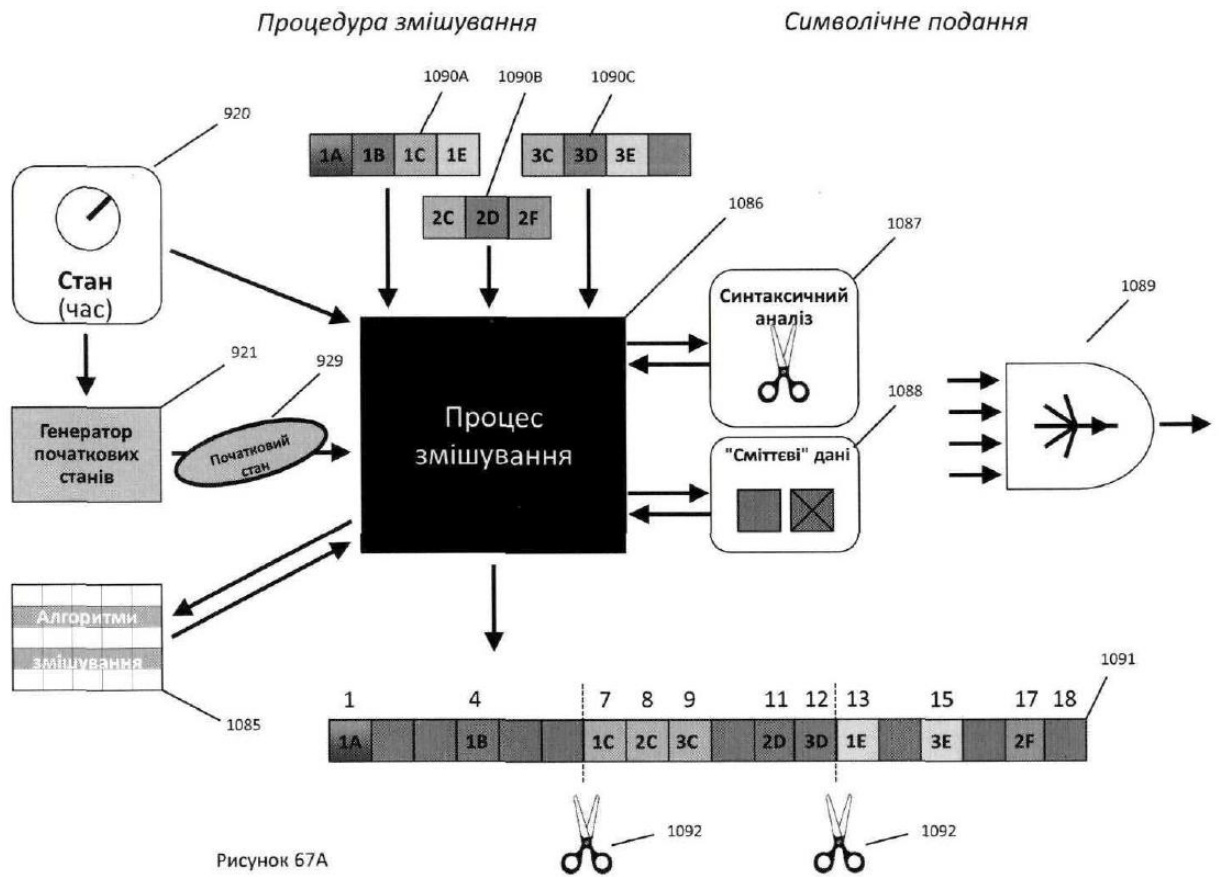


Рисунок 66D



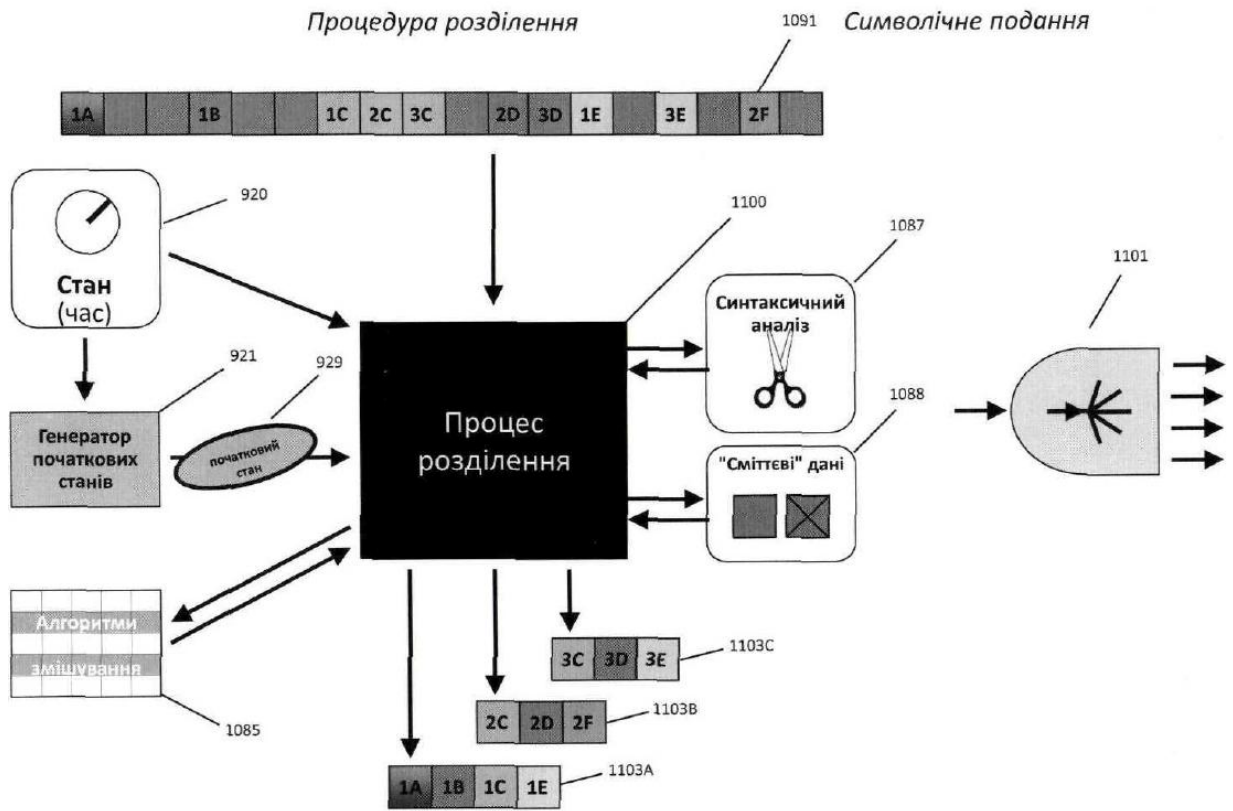


Рисунок 67C

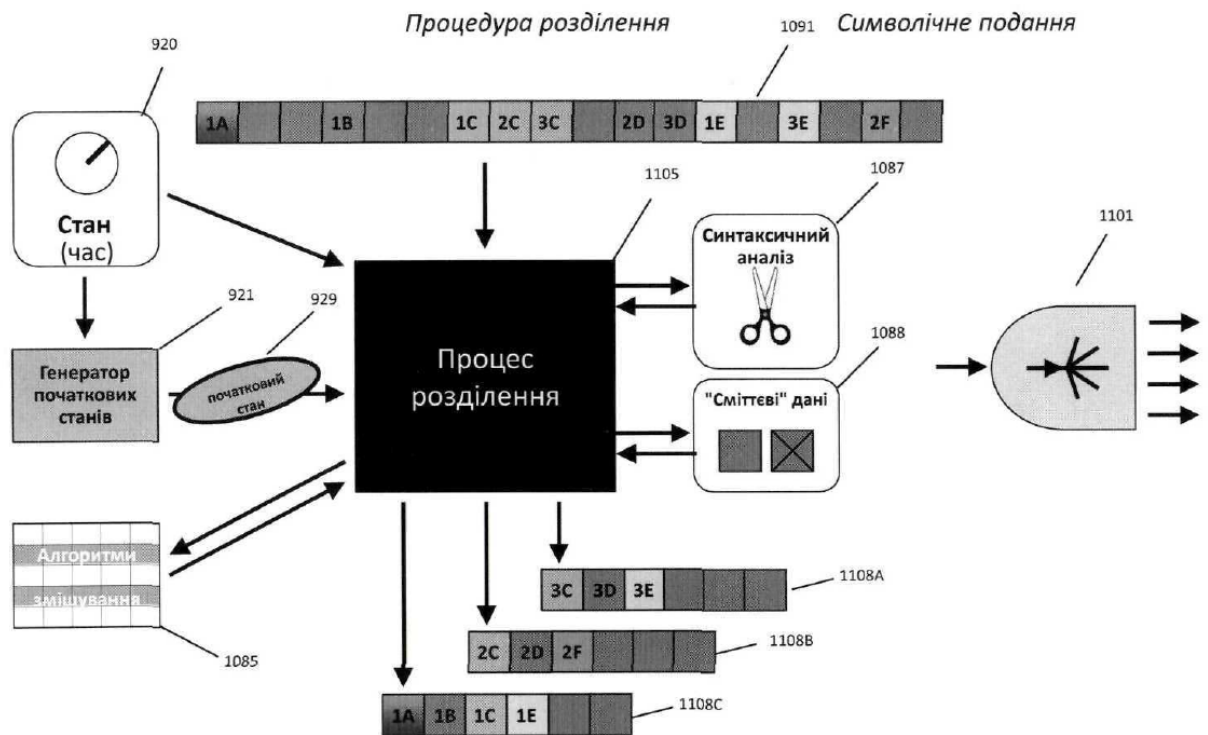
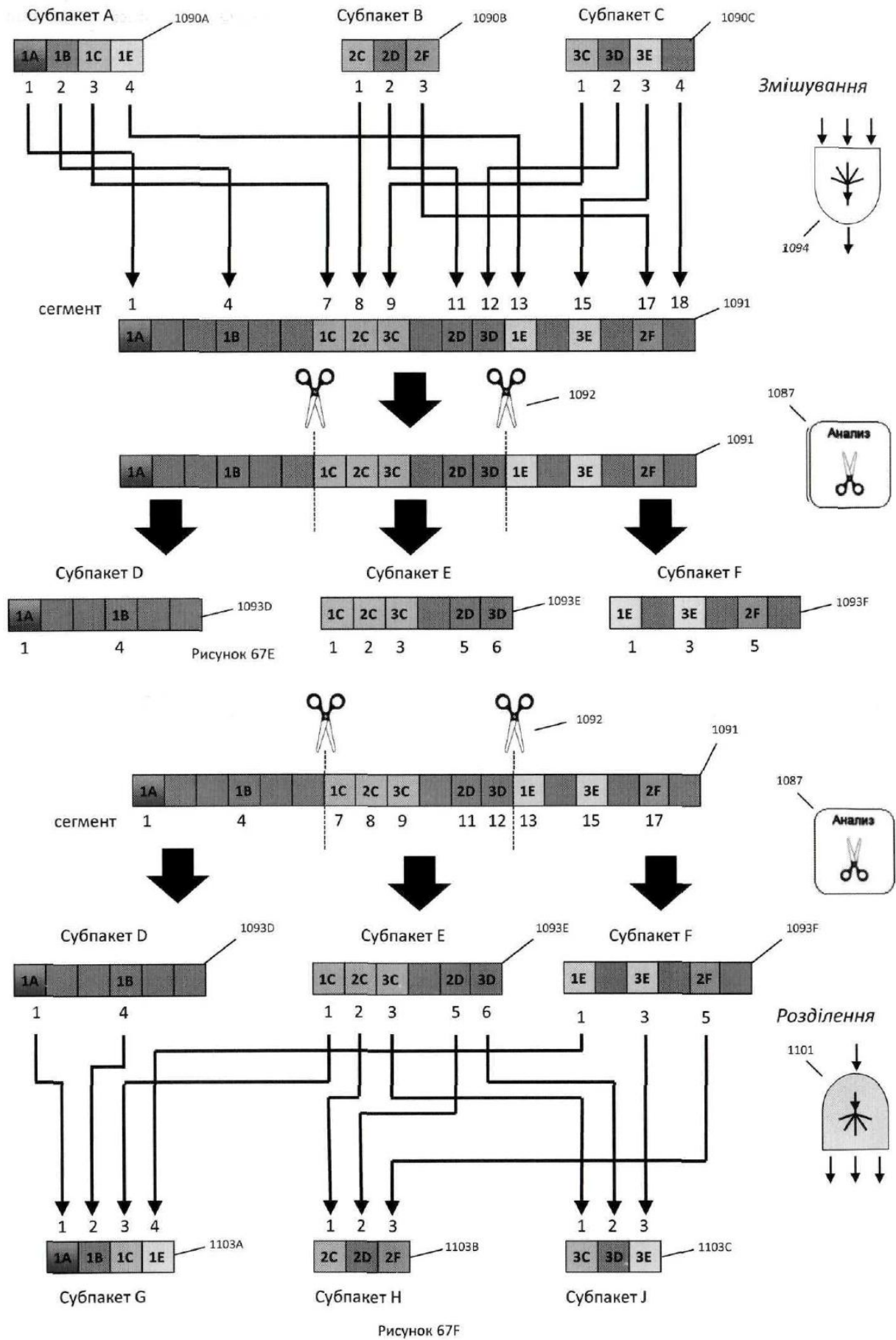


Рисунок 67D



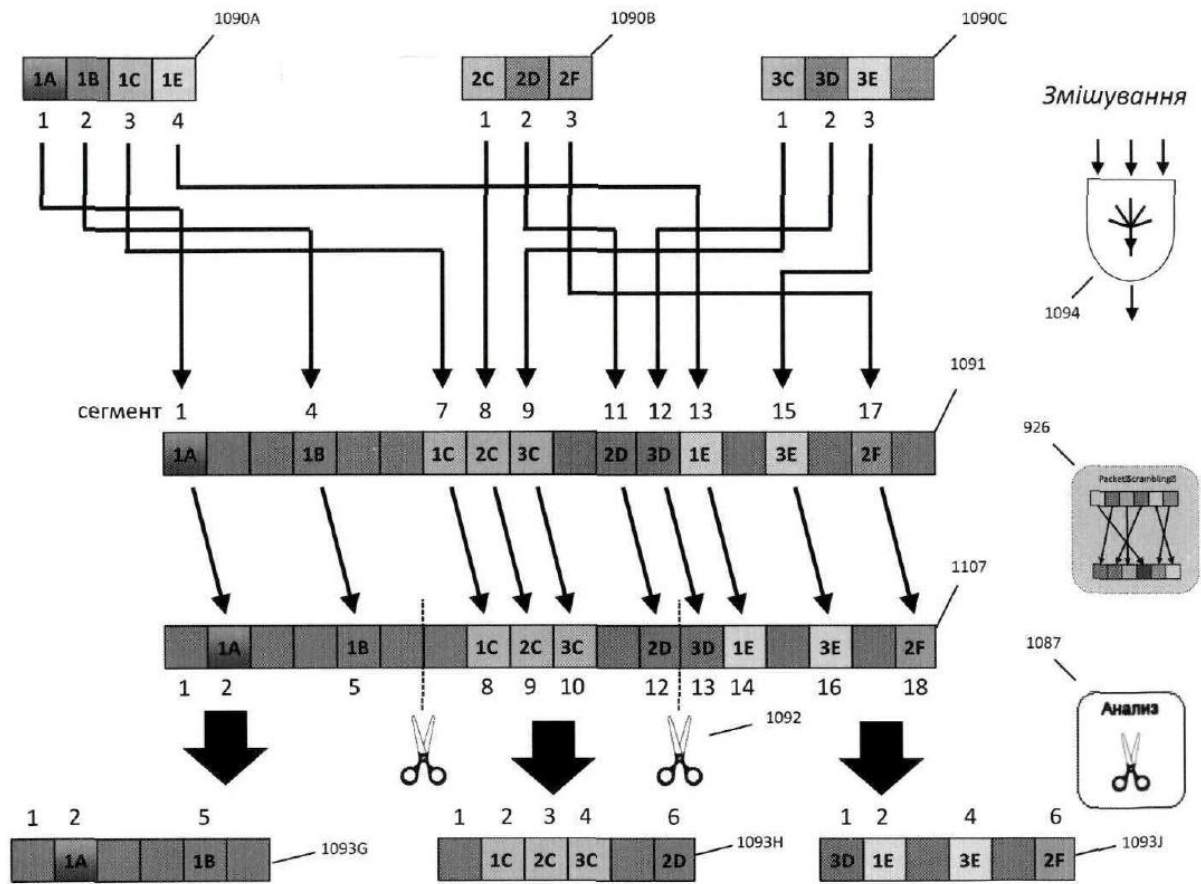


Рисунок 67G

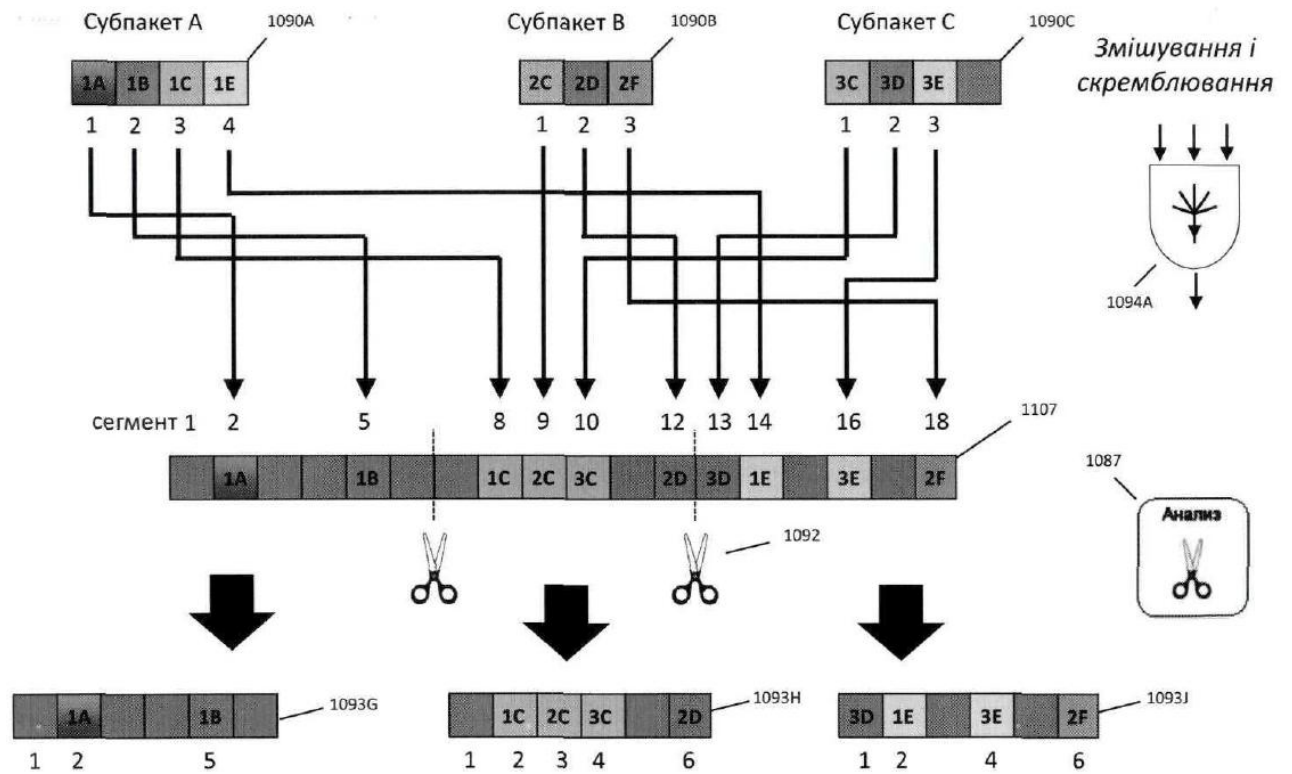


Рисунок 67H

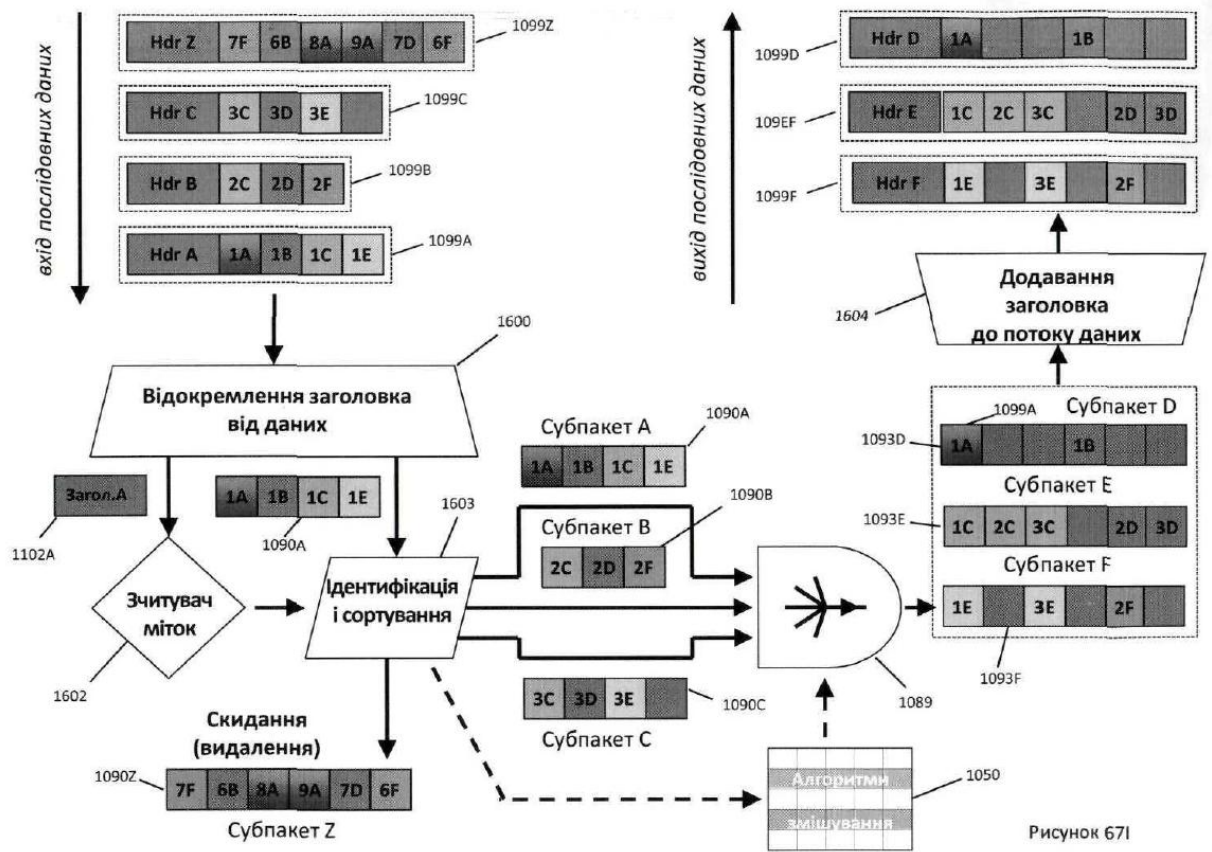


Рисунок 67I

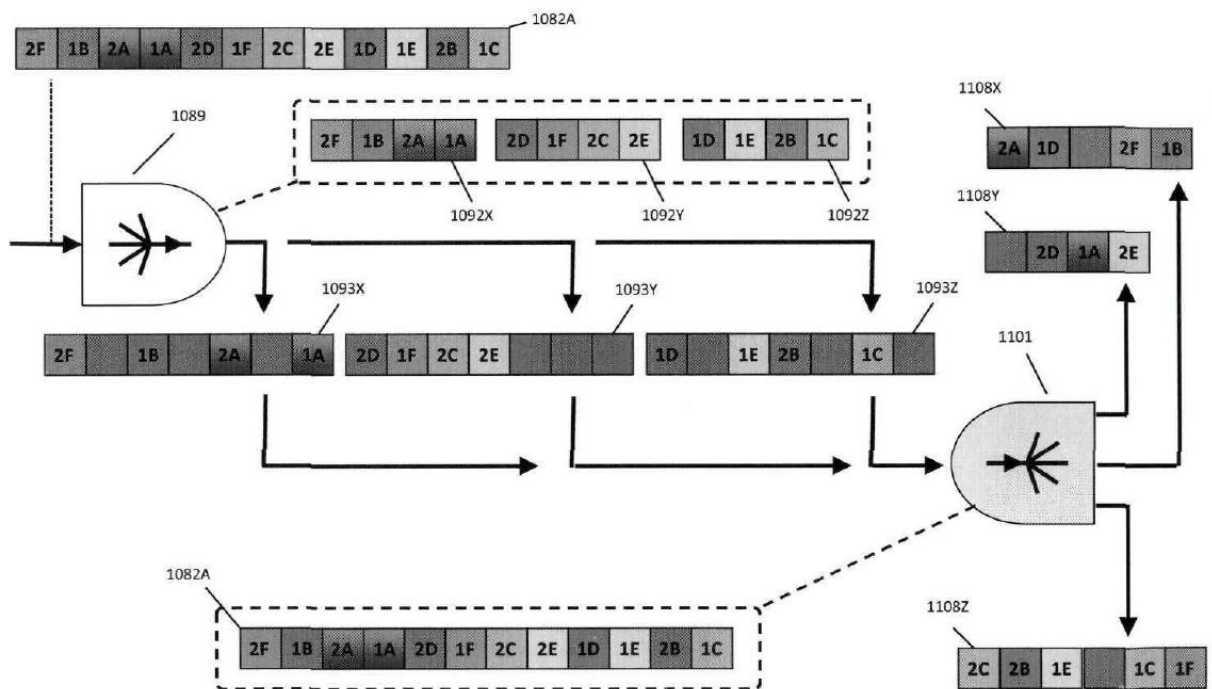
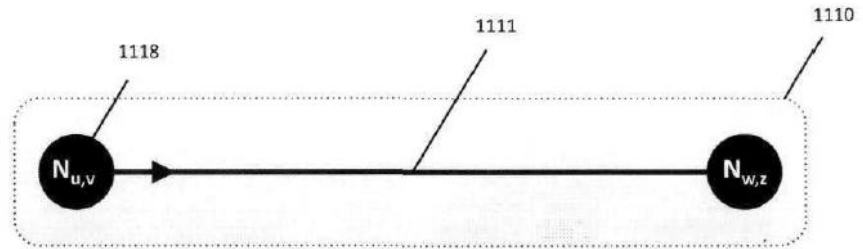
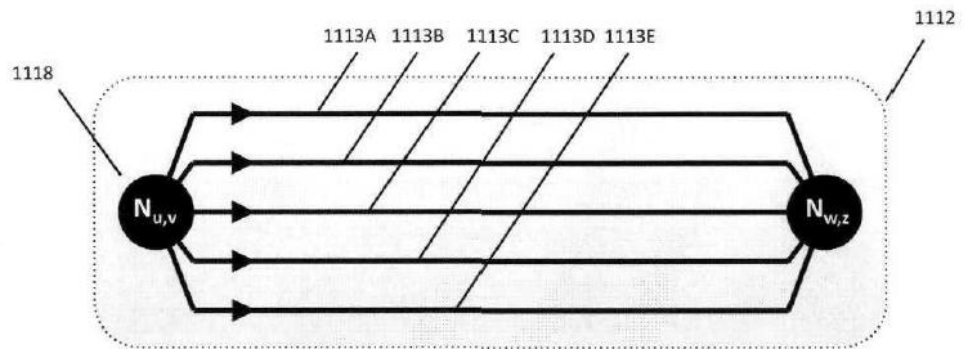


Рисунок 67J

Одношляхова
маршрутизація



Багатошляхова
маршрутизація



Решітчаста
маршрутизація

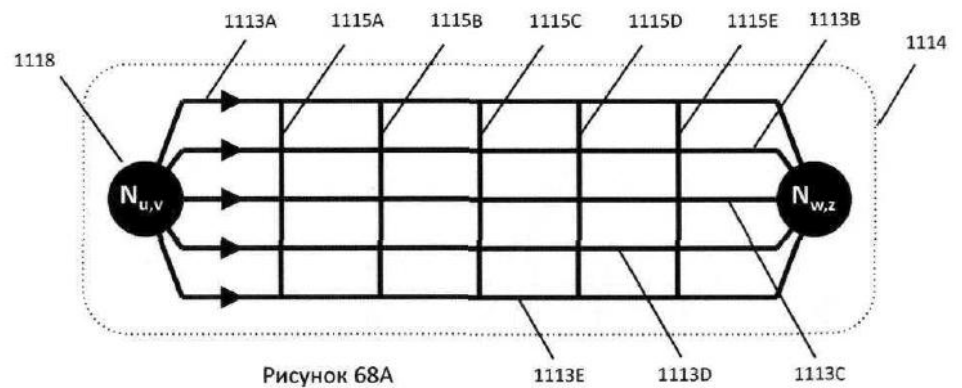


Рисунок 68А

1110

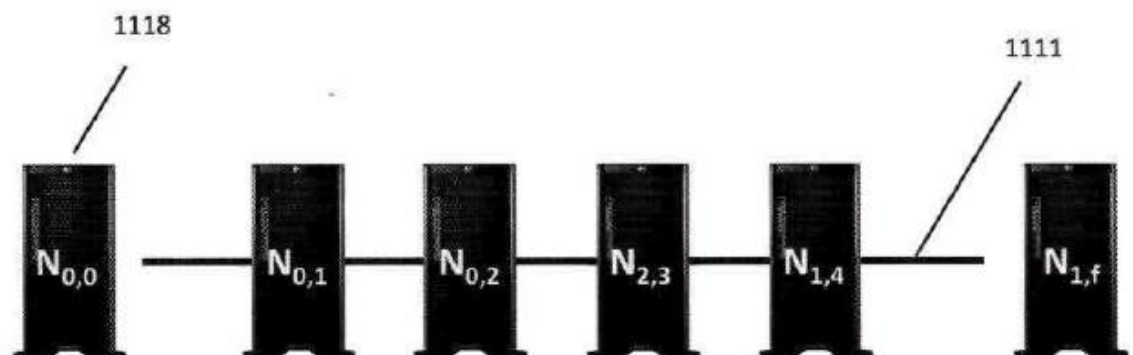


Рисунок 68В

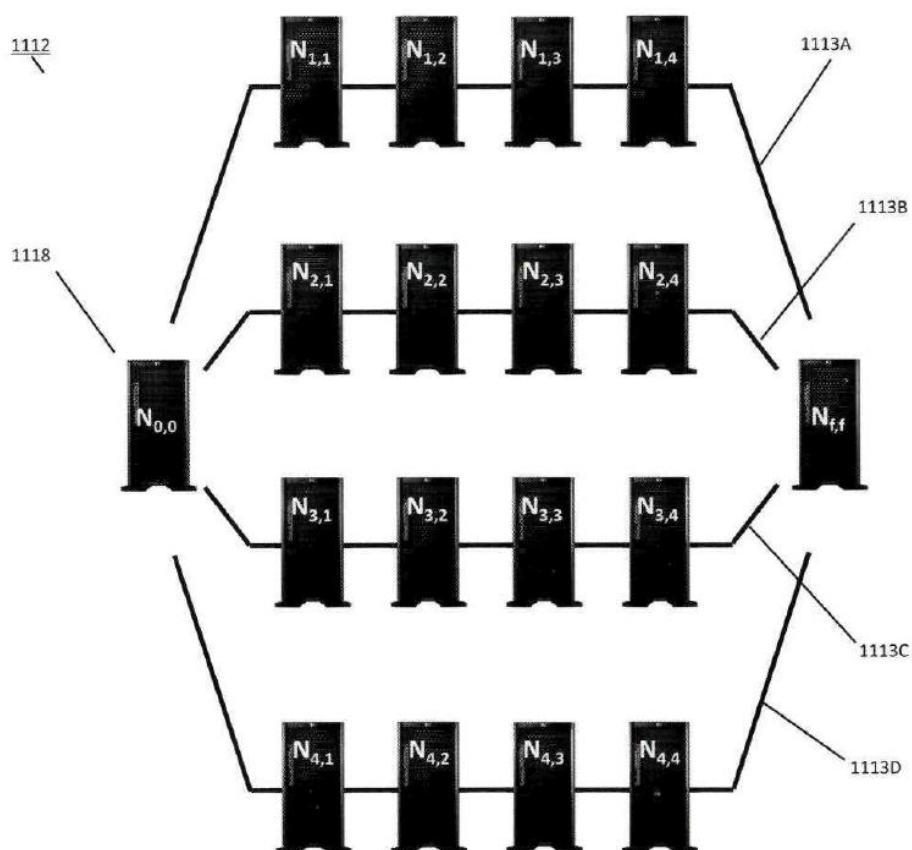


Рисунок 68С

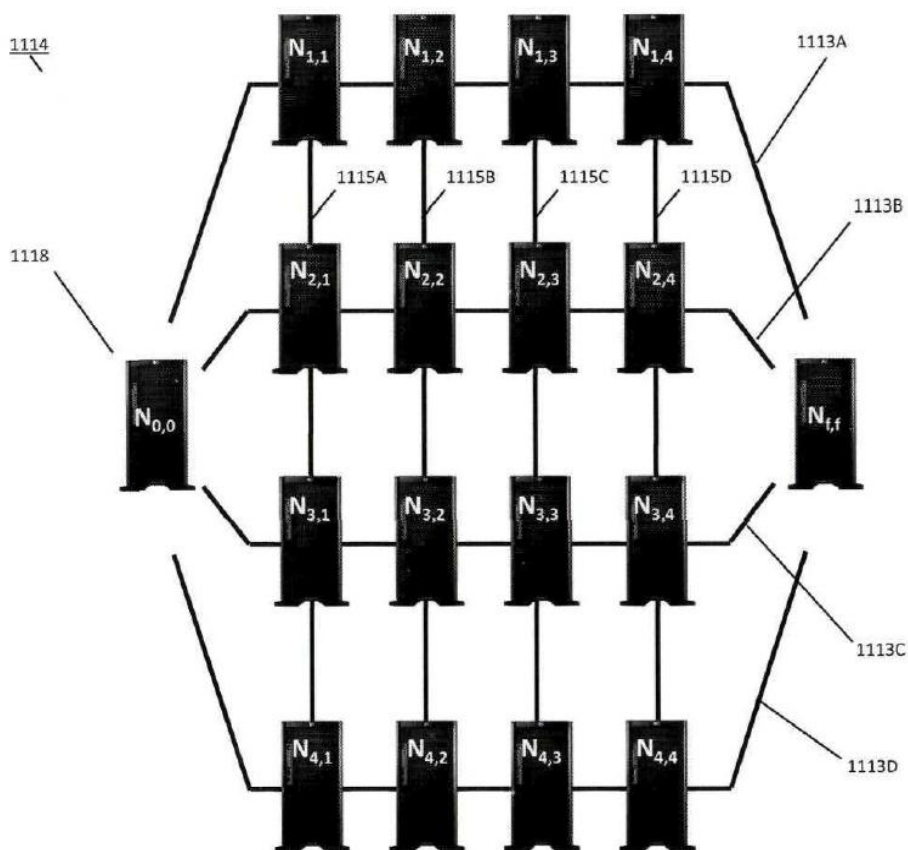


Рисунок 68D

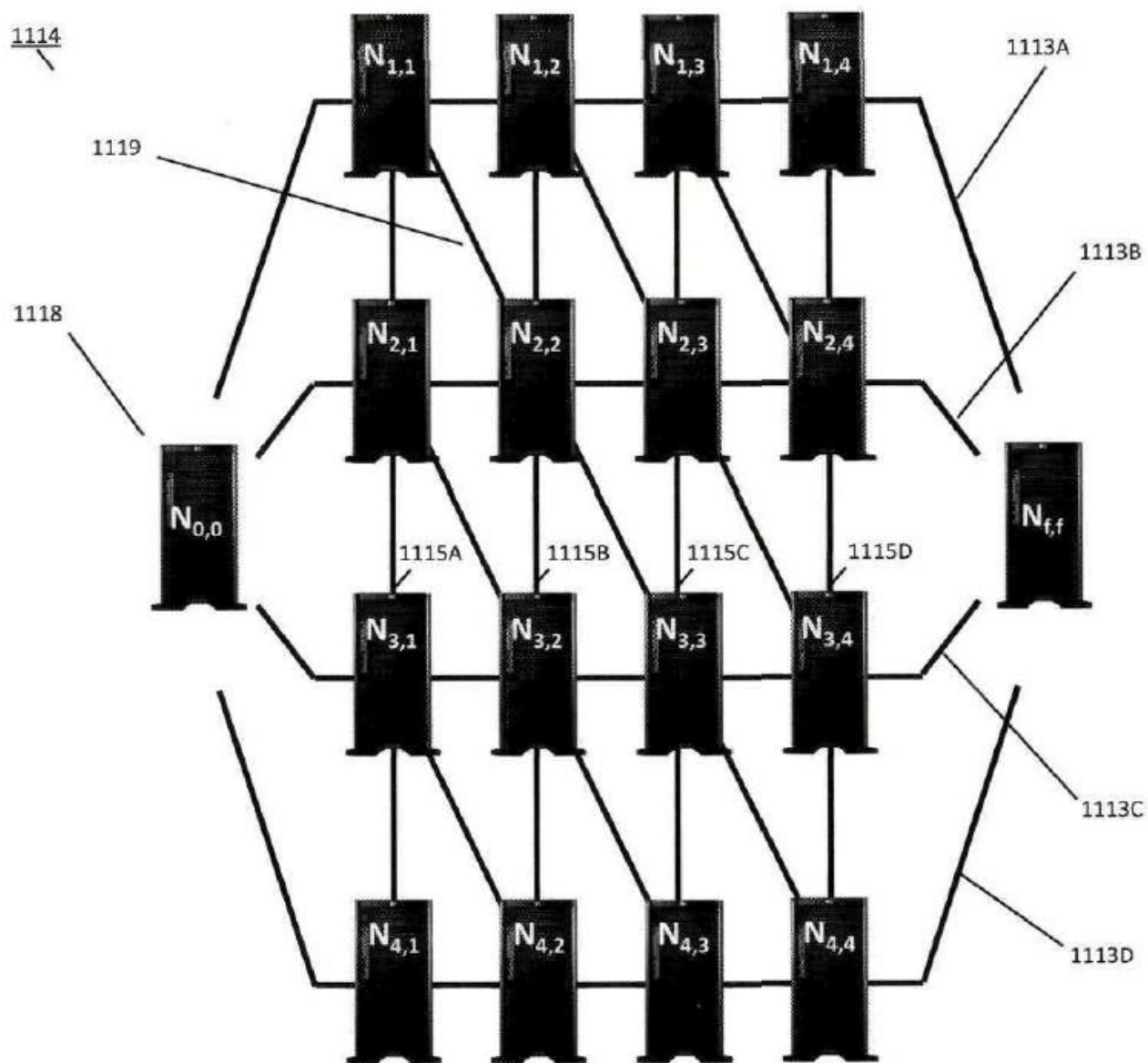


Рисунок 68Е

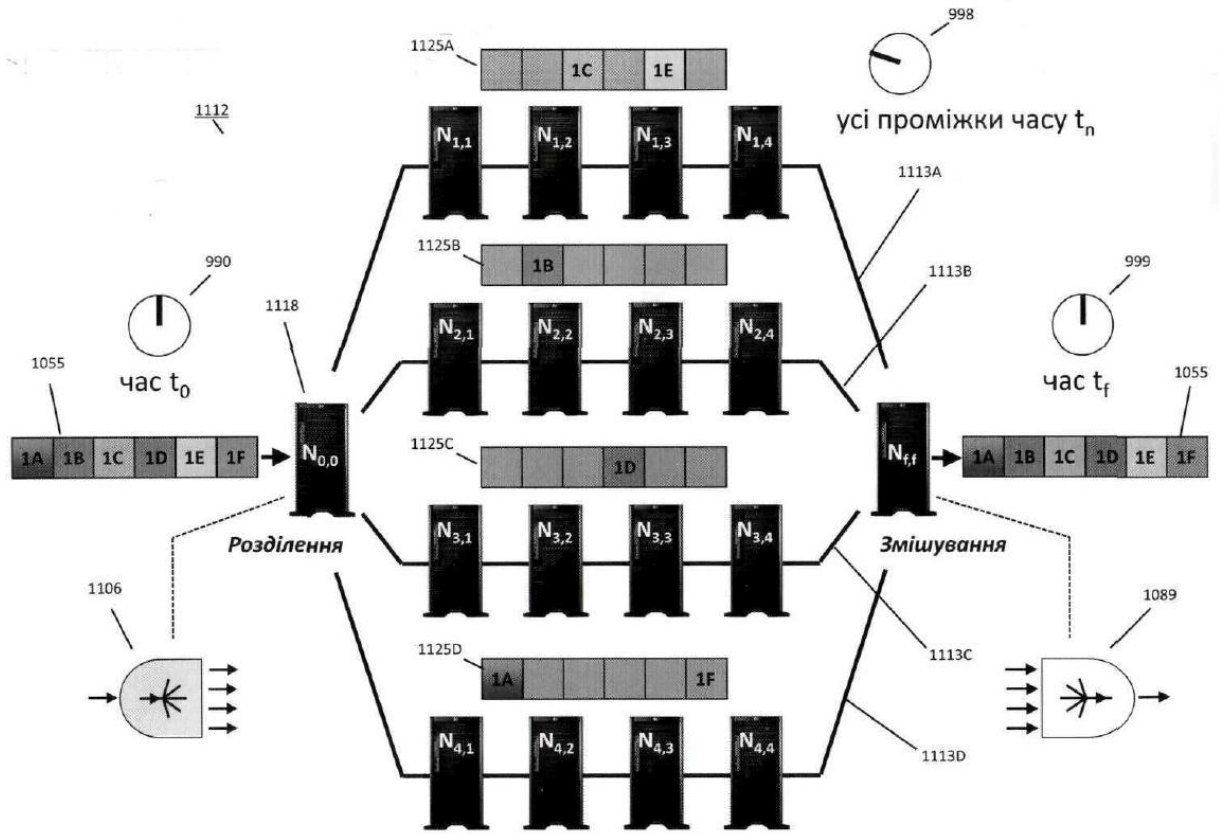


Рисунок 69

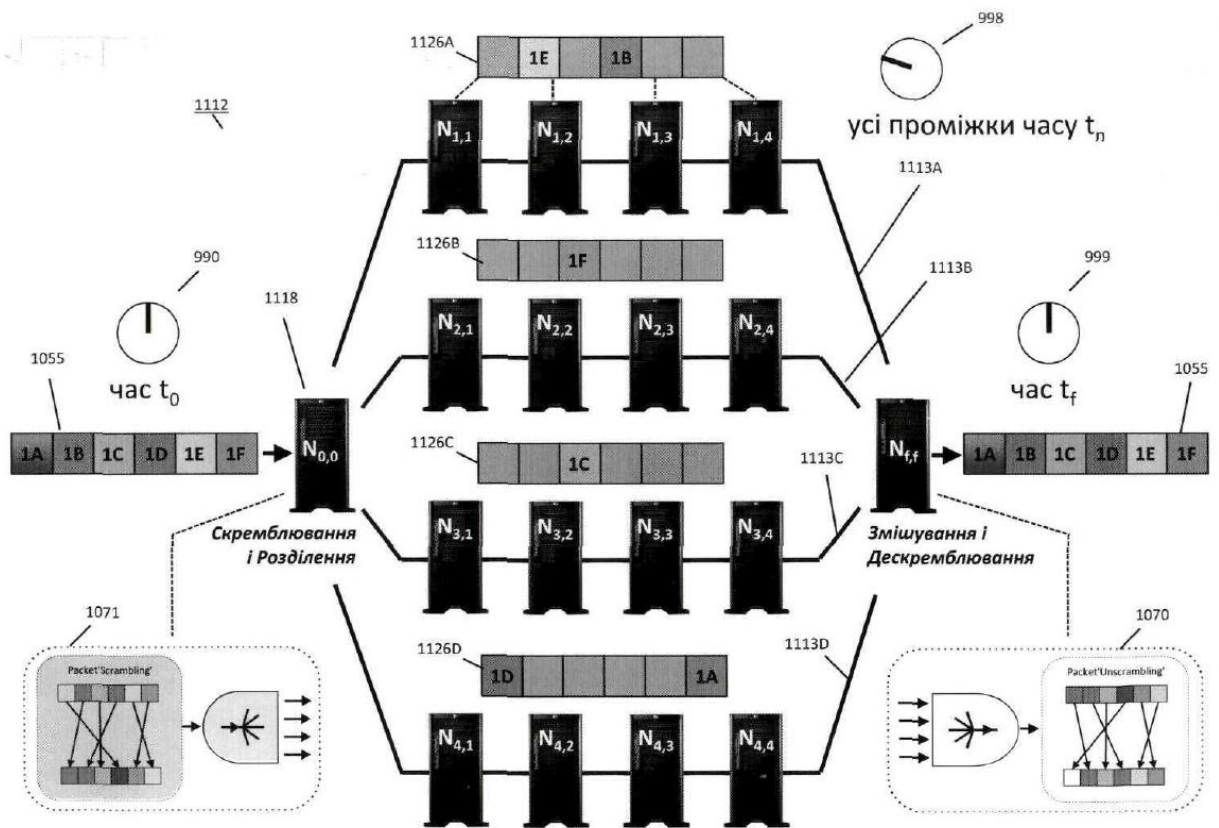


Рисунок 70

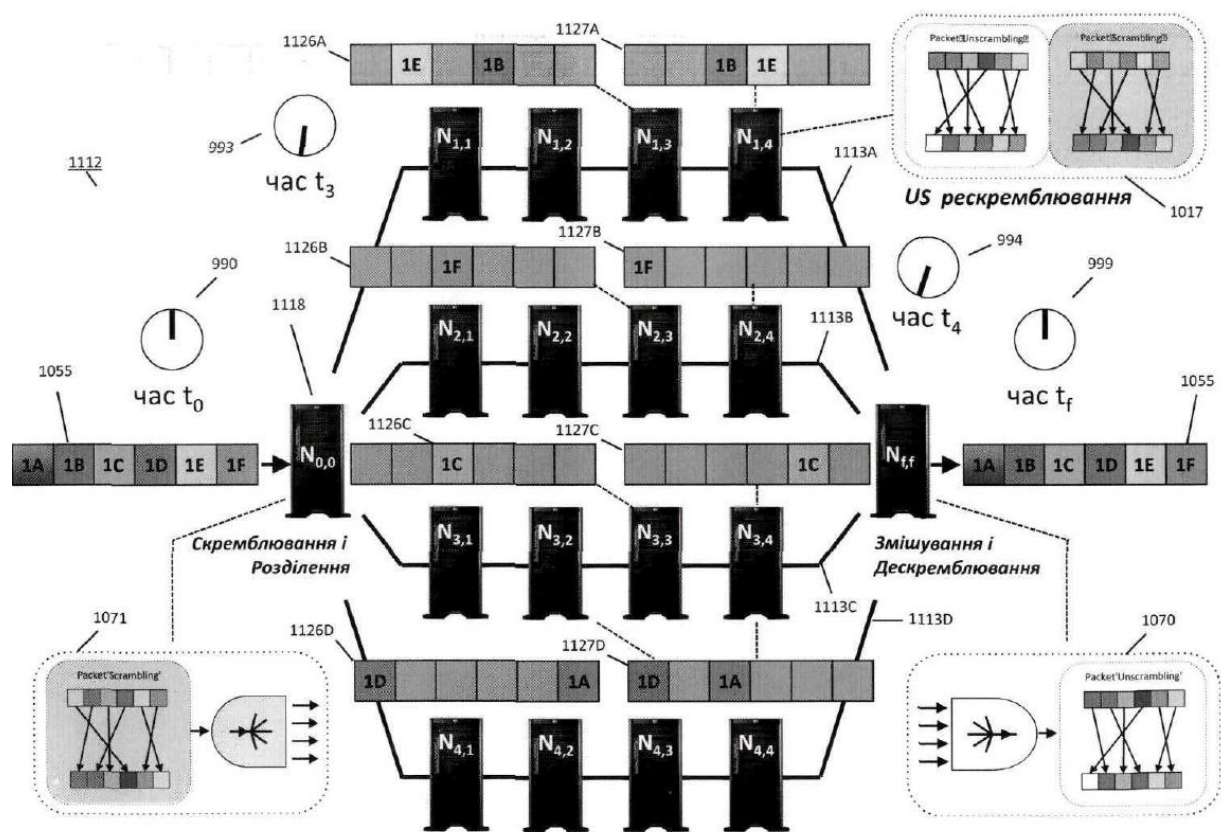


Рисунок 71А

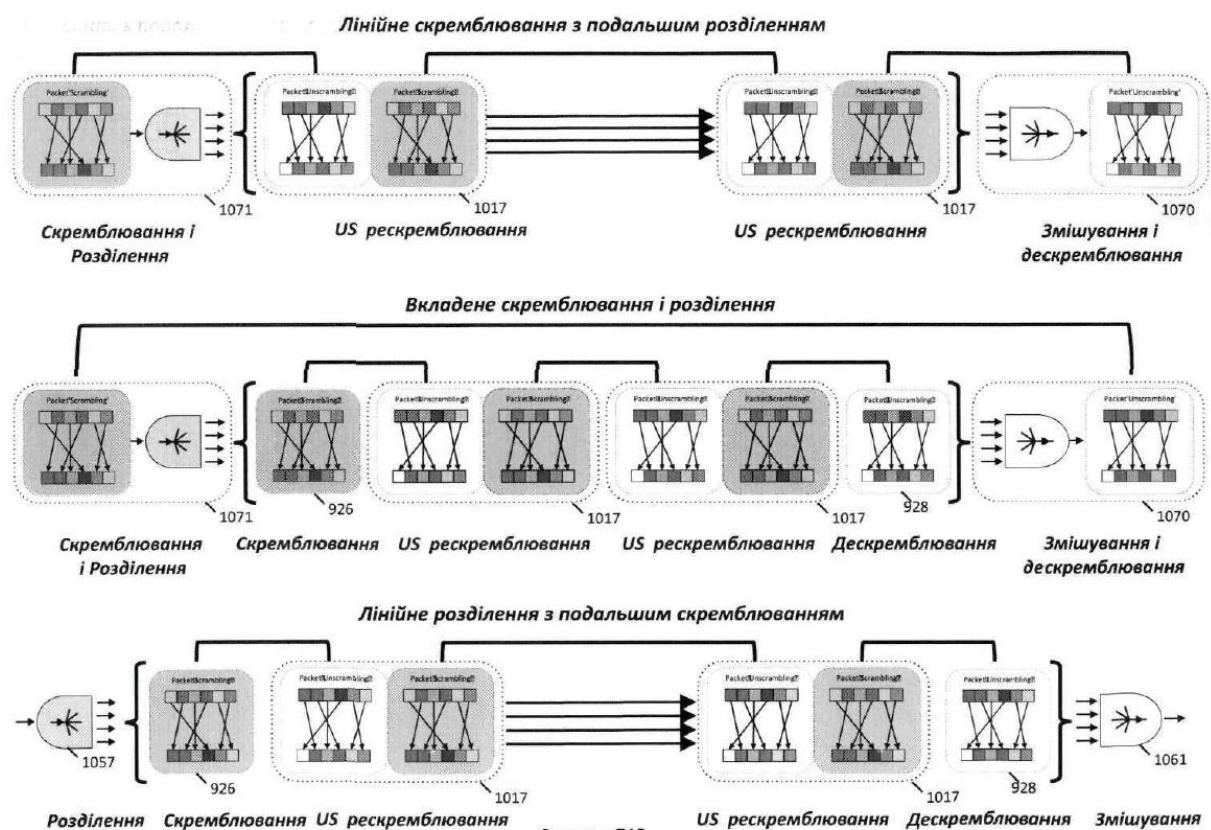


Рисунок 71В

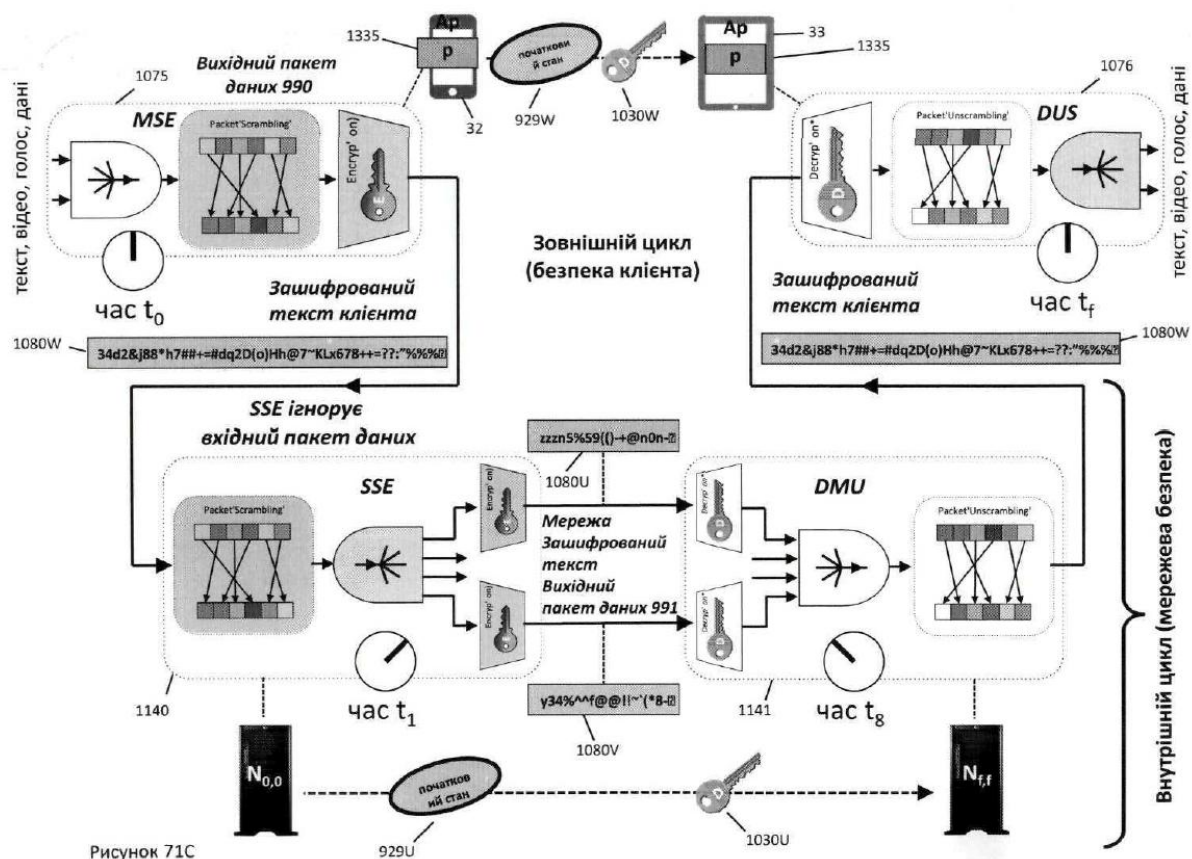


Рисунок 71С

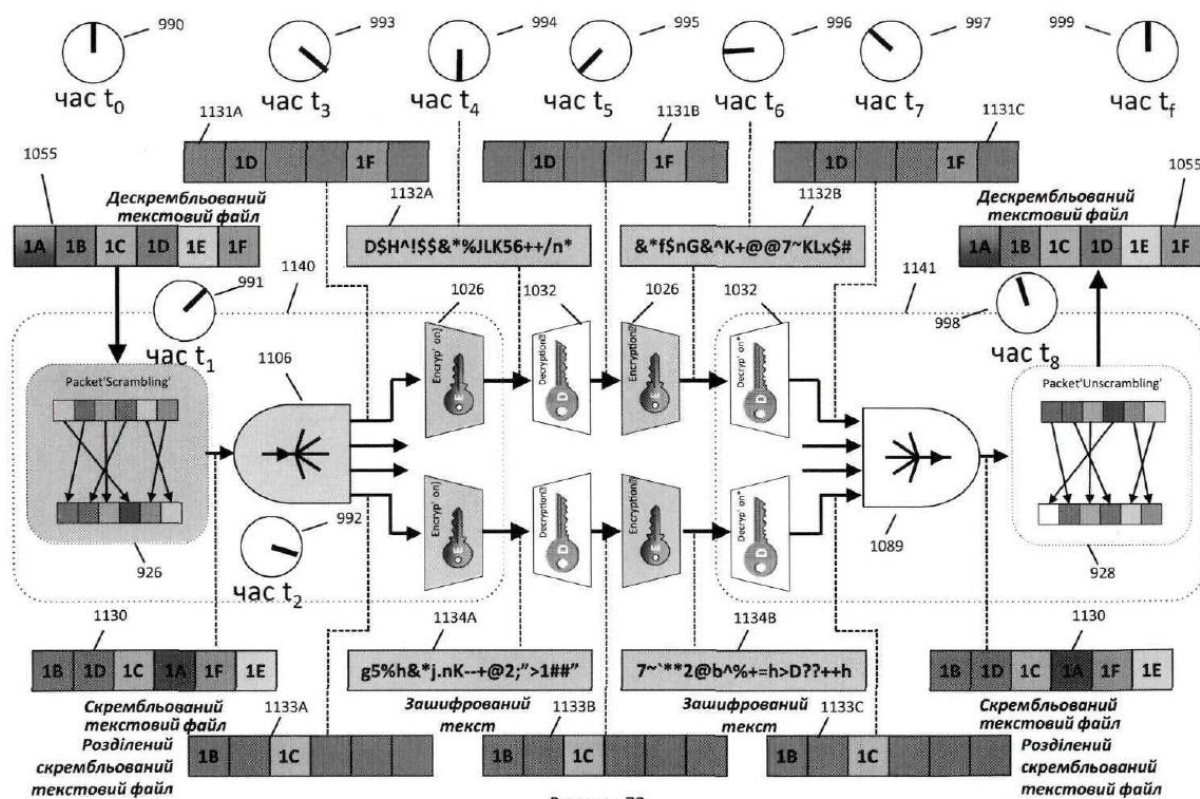


Рисунок 72

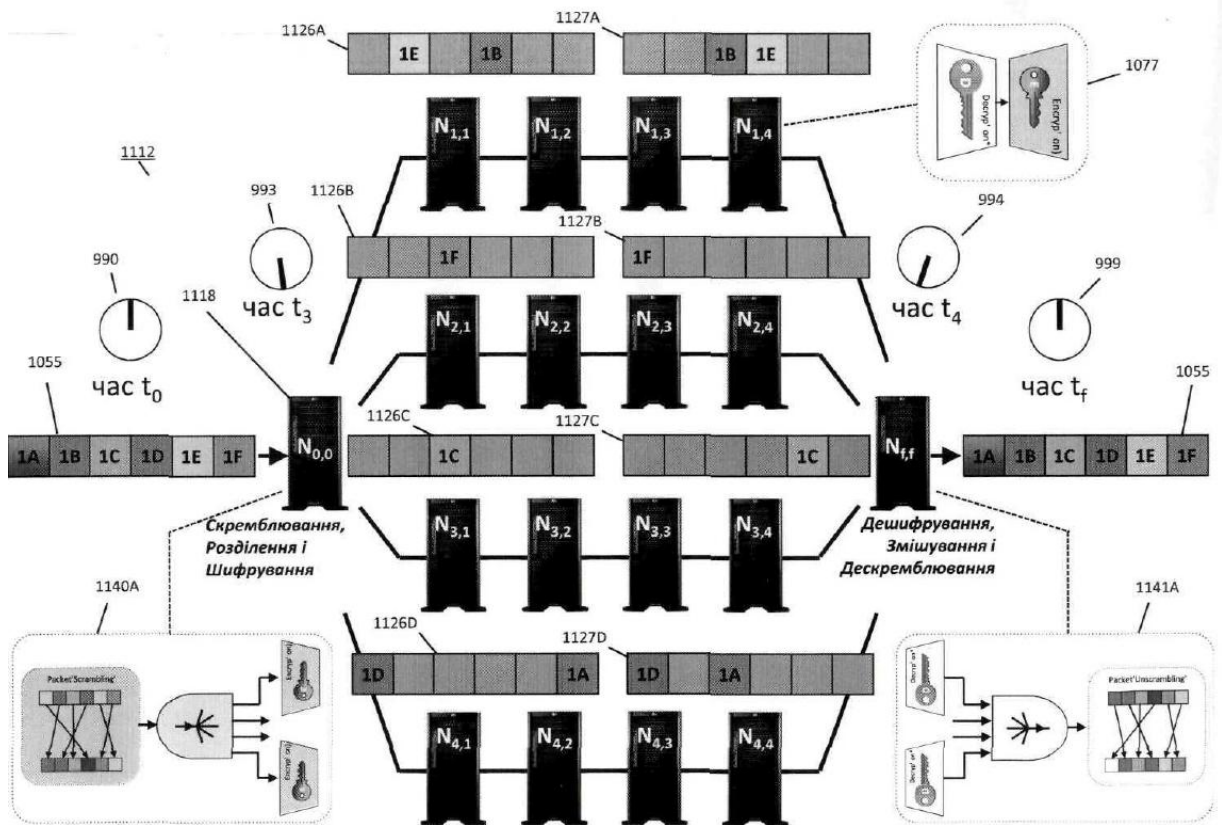


Рисунок 73

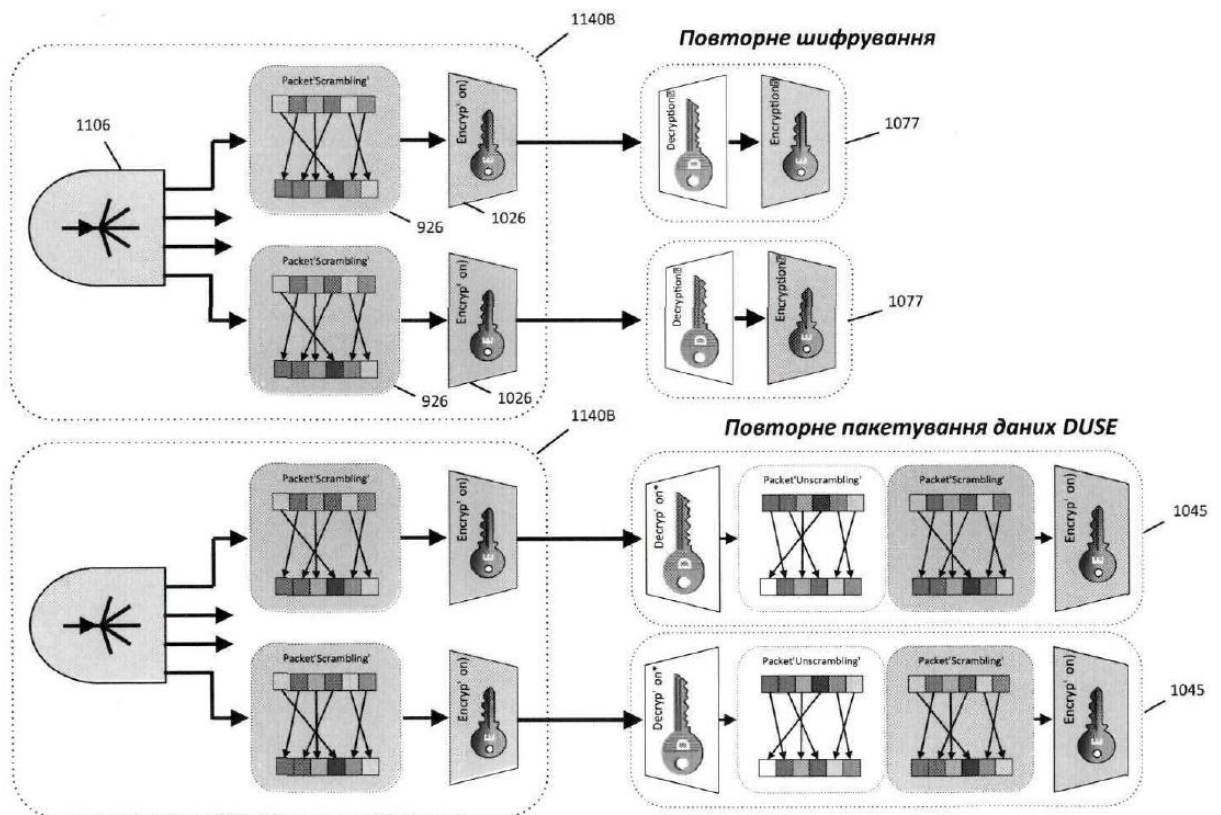


Рисунок 74

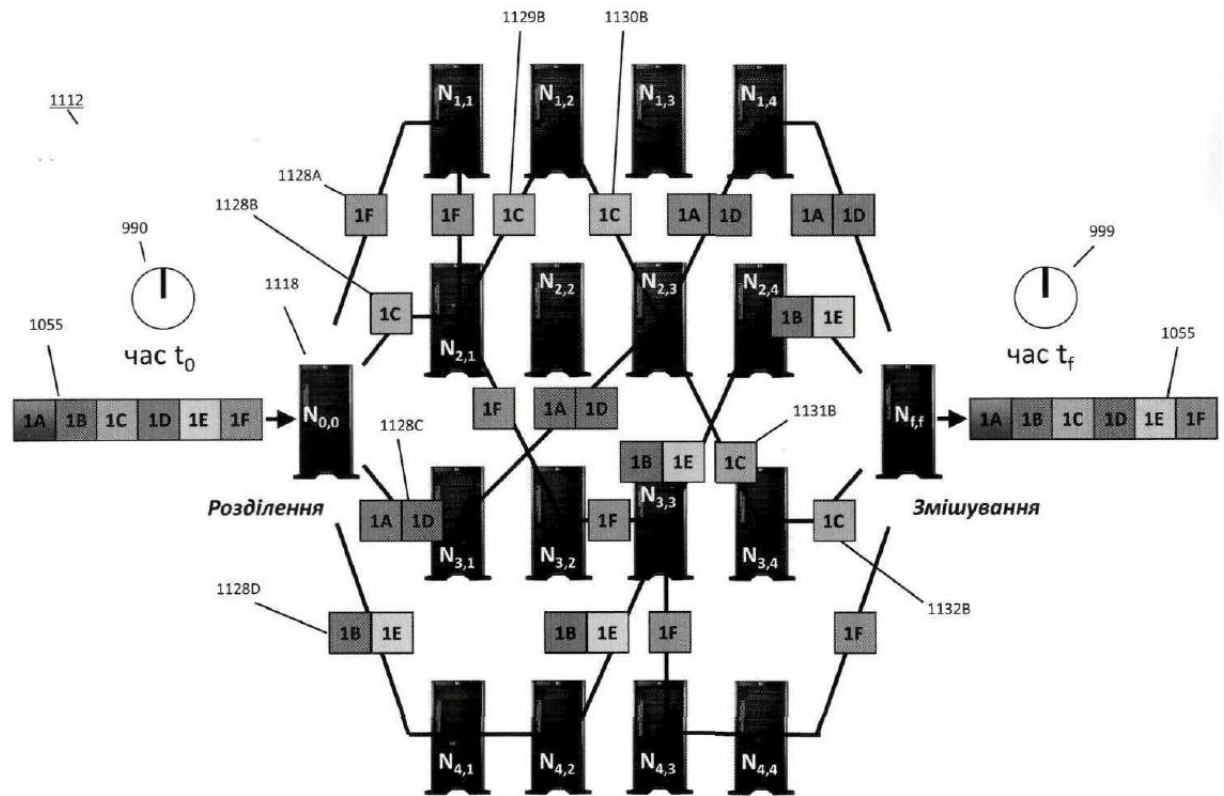


Рисунок 75

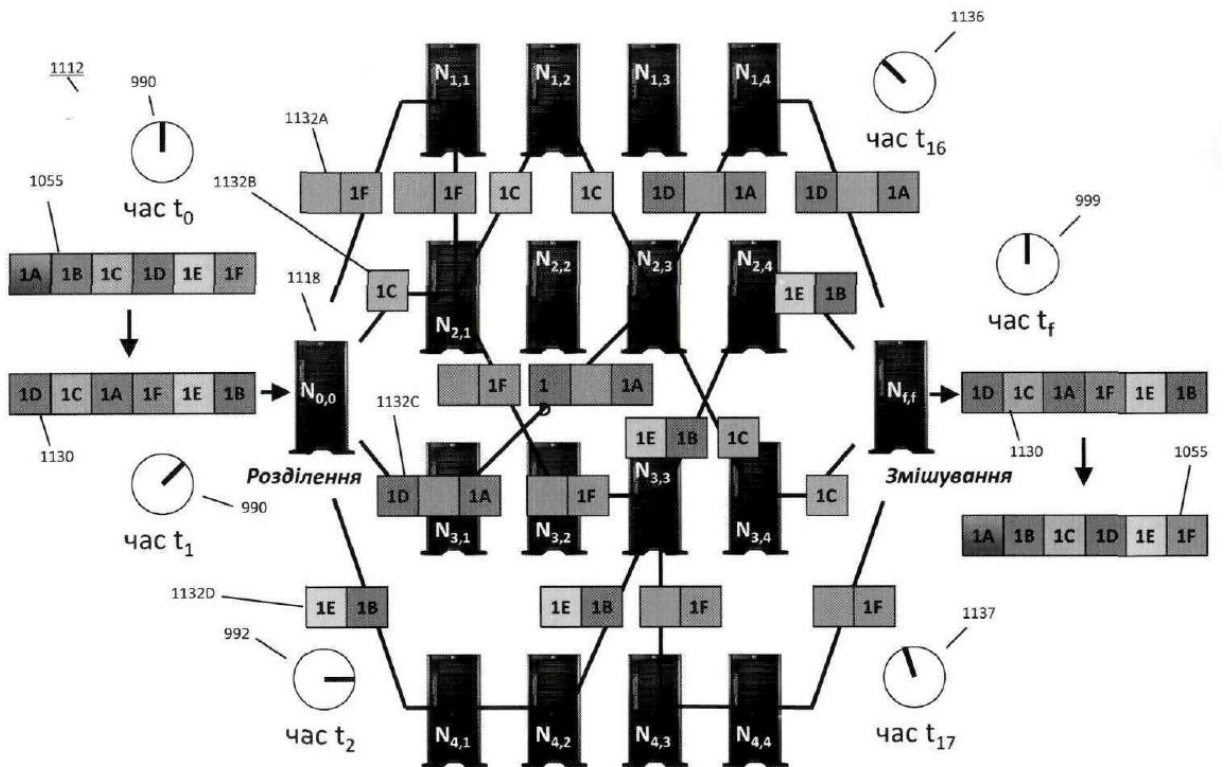


Рисунок 76

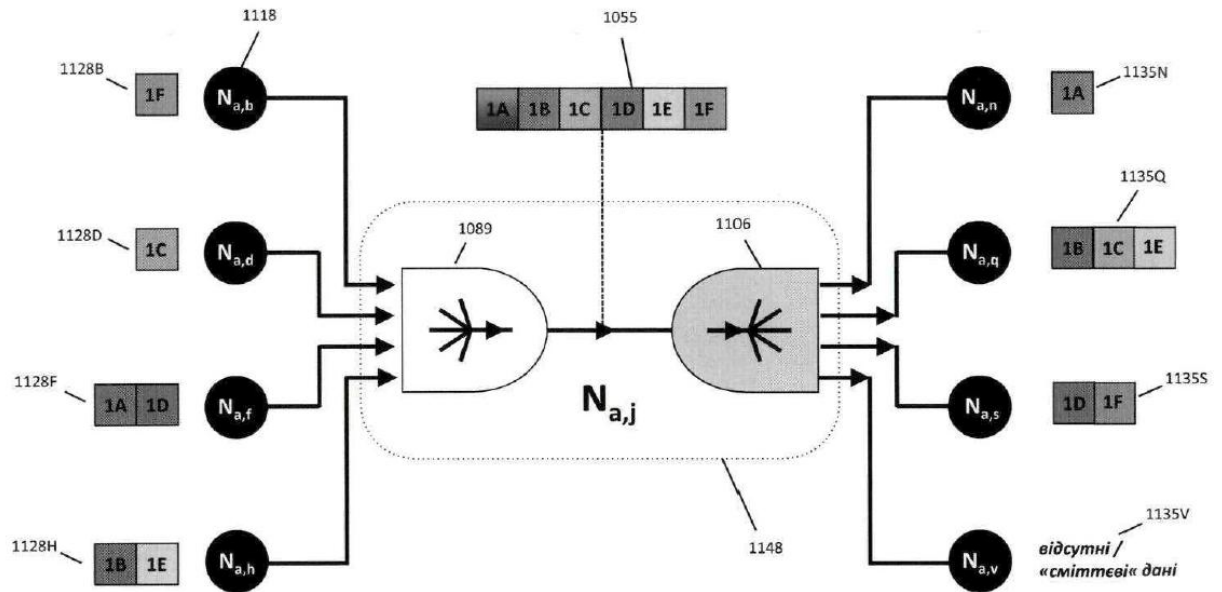


Рисунок 77А

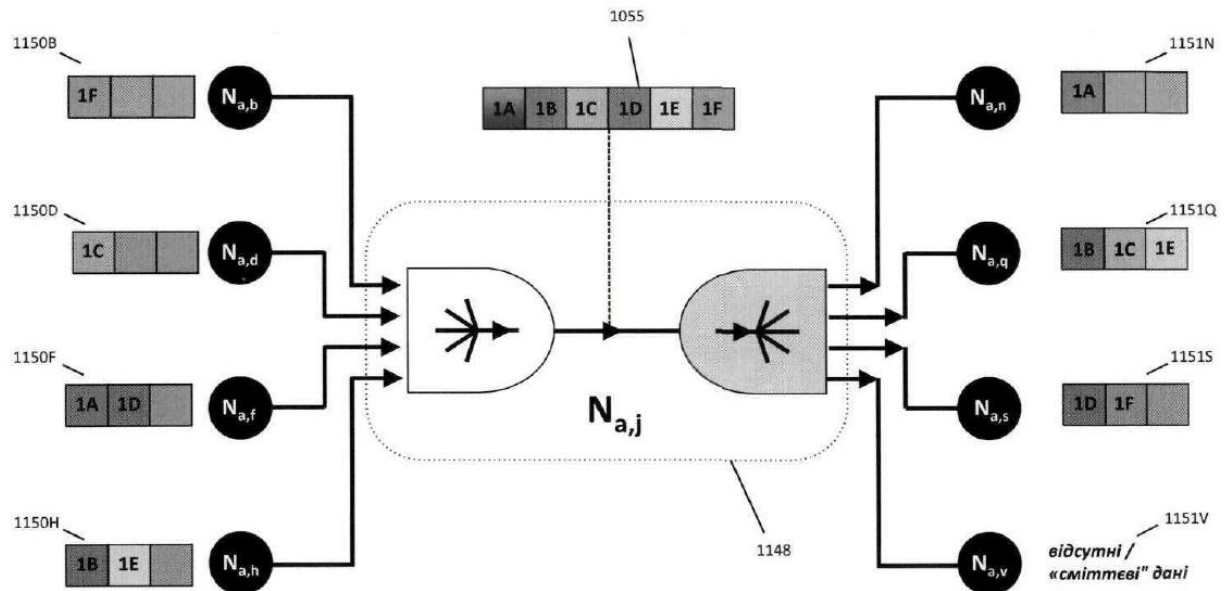


Рисунок 77В

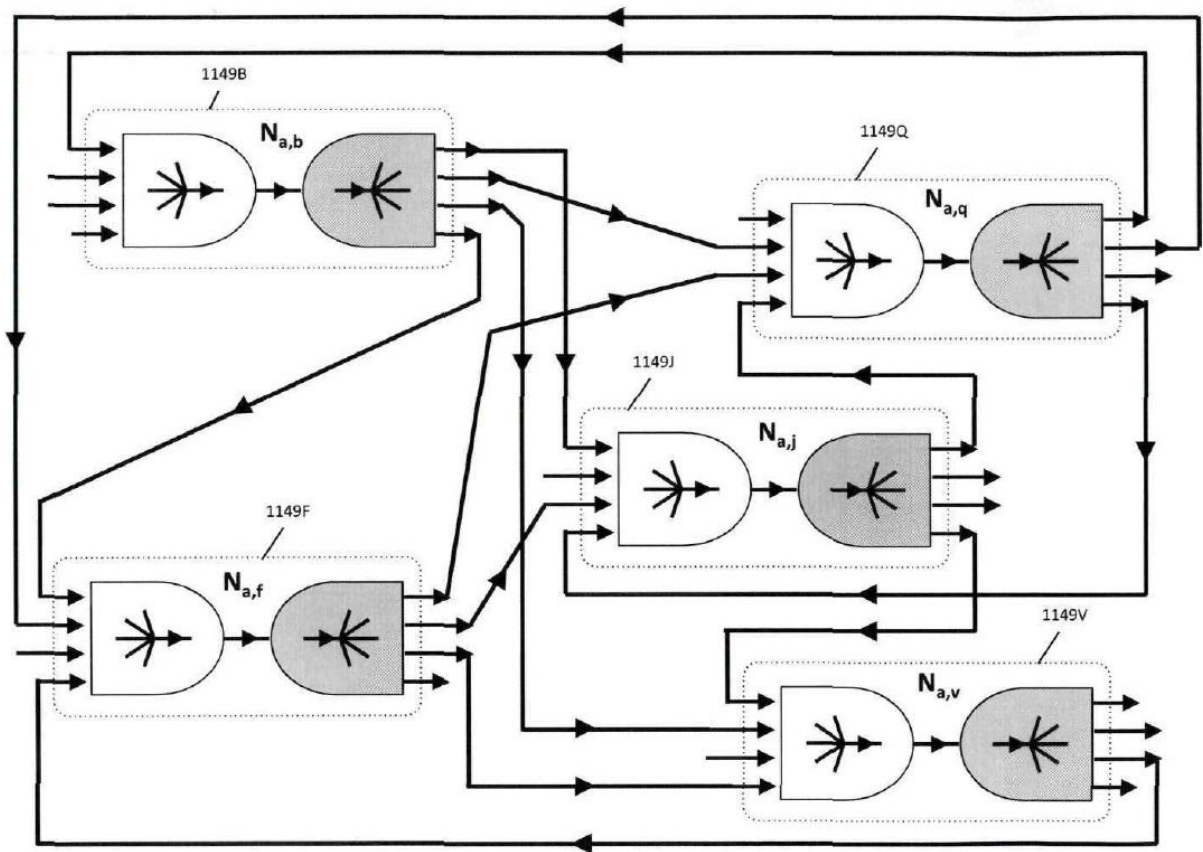


Рисунок 77С

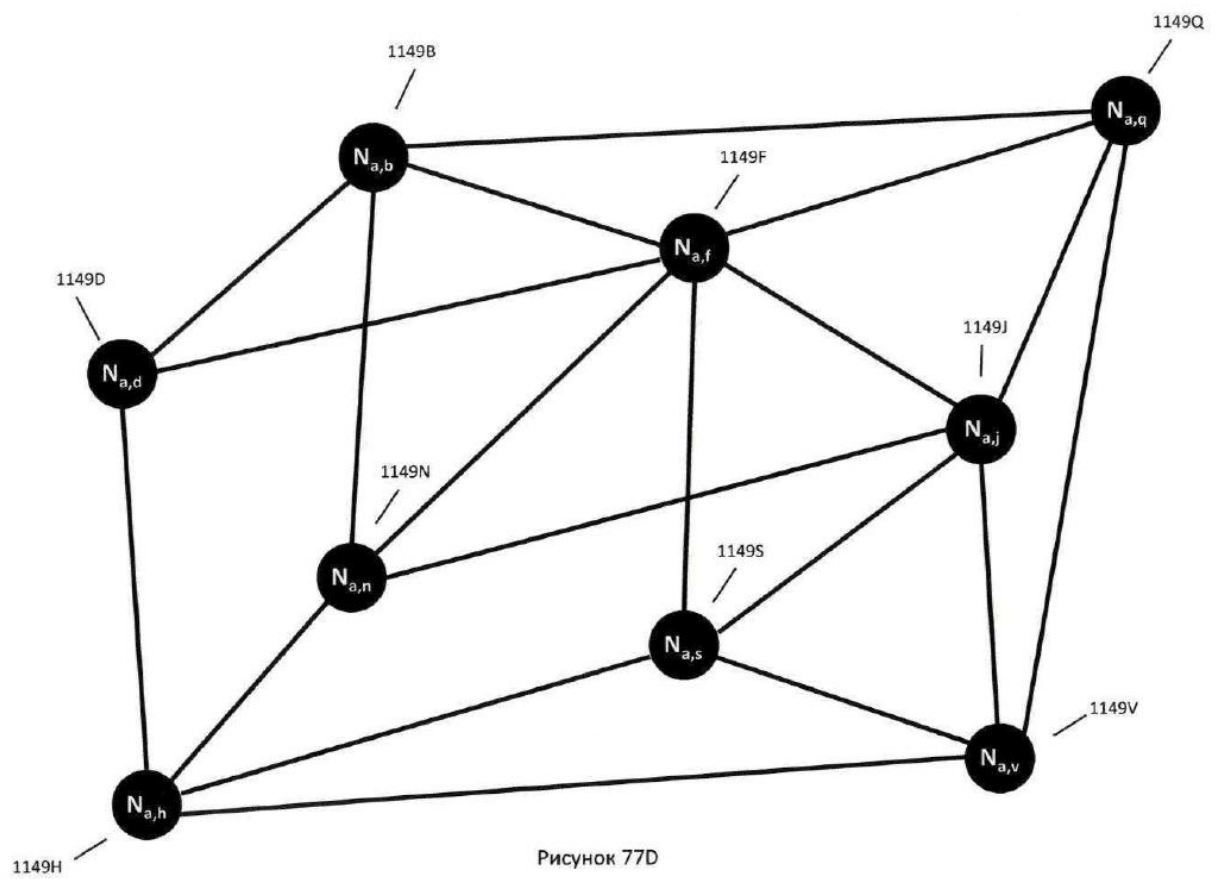


Рисунок 77D

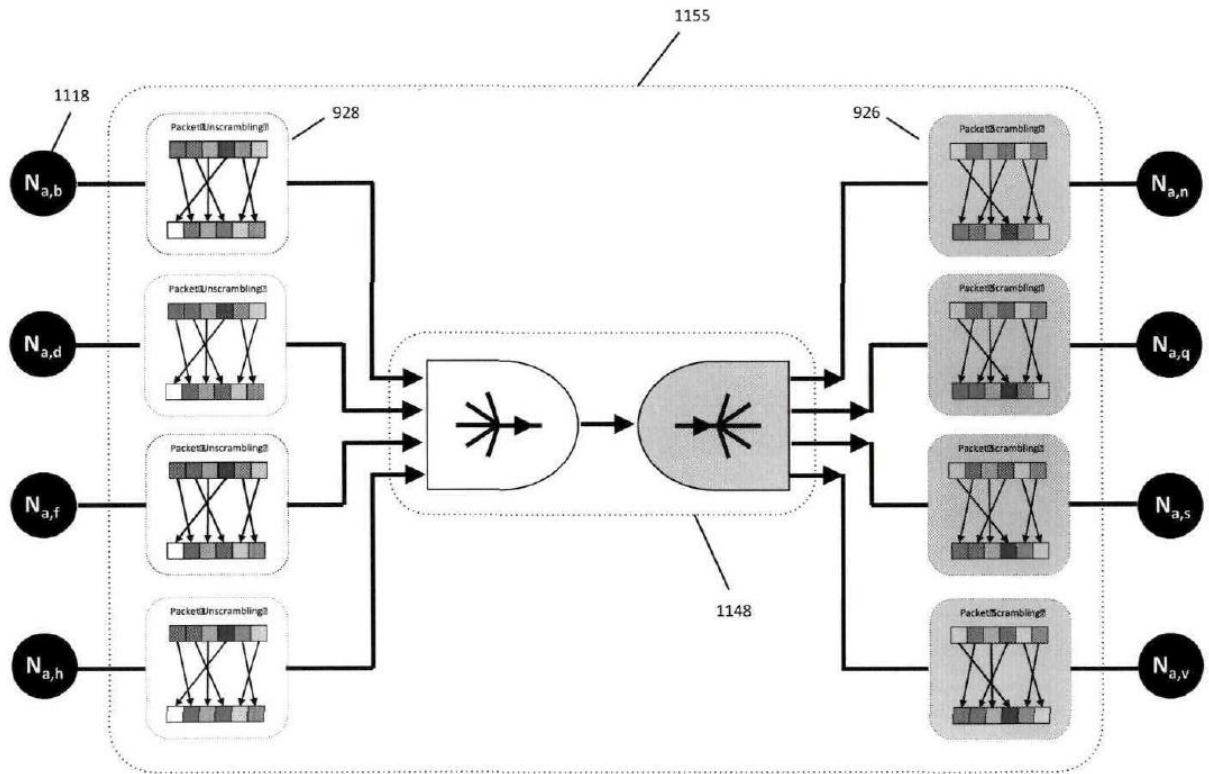


Рисунок 78А

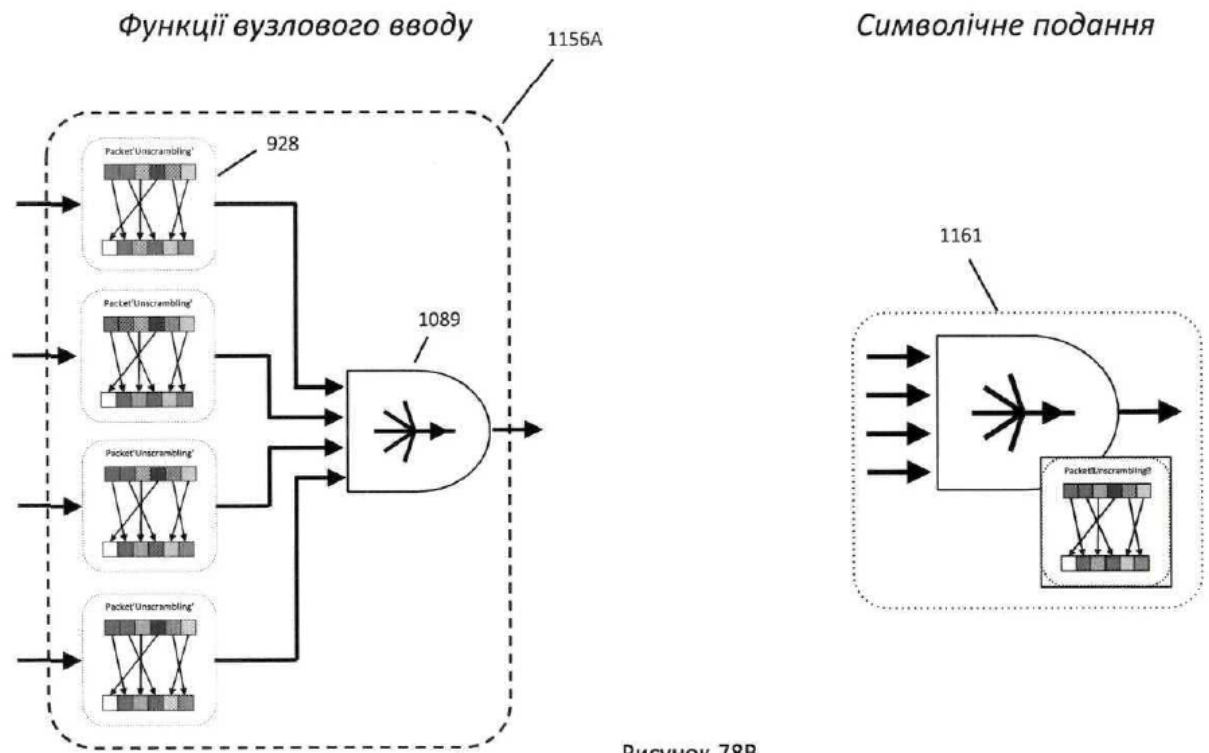
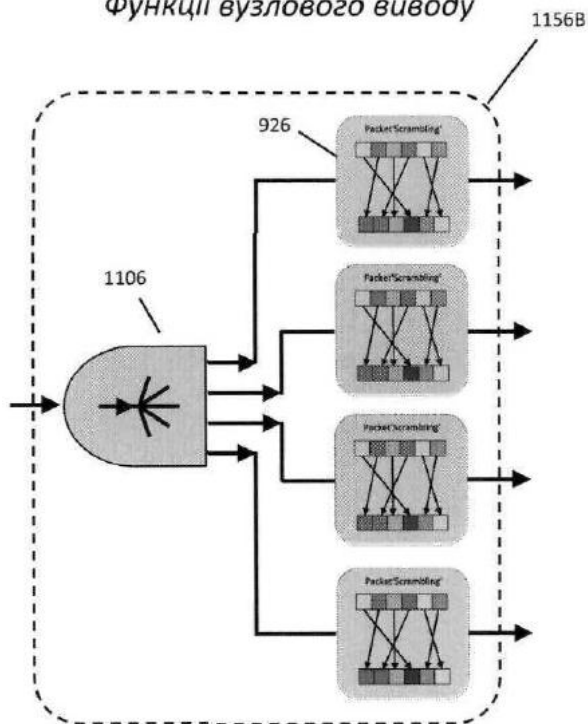


Рисунок 78В

Функції вузлового виводу



Символічне подання

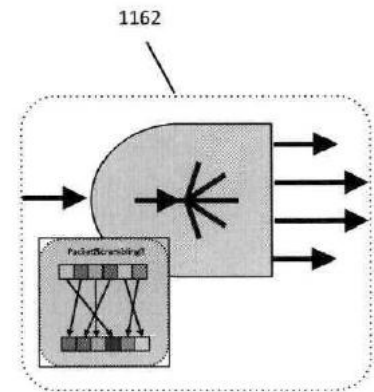
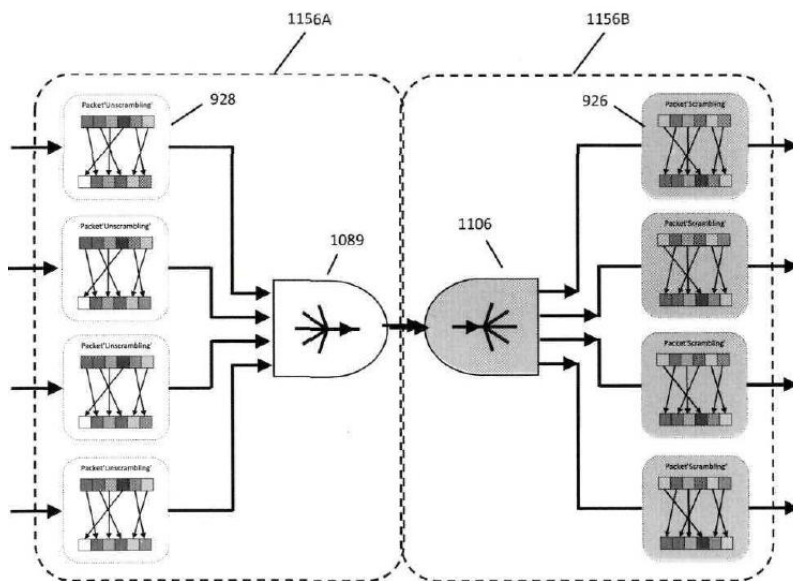


Рисунок 78С

Функції вузлового вводу



Символічне подання

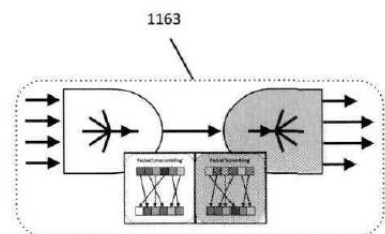


Рисунок 78D

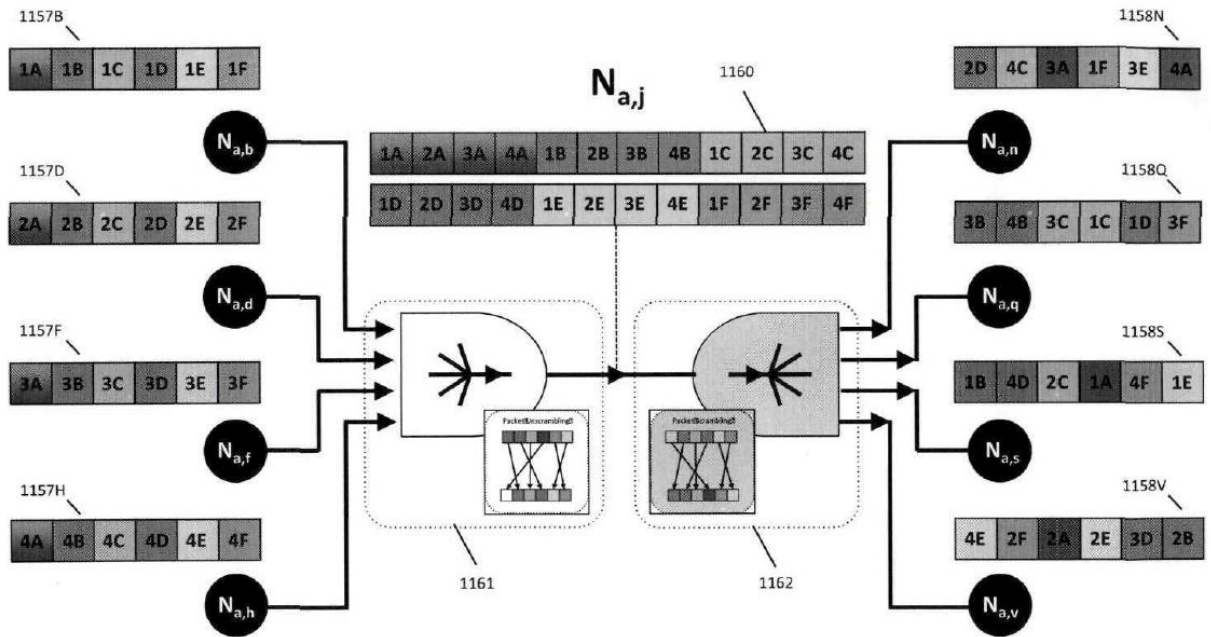


Рисунок 79А

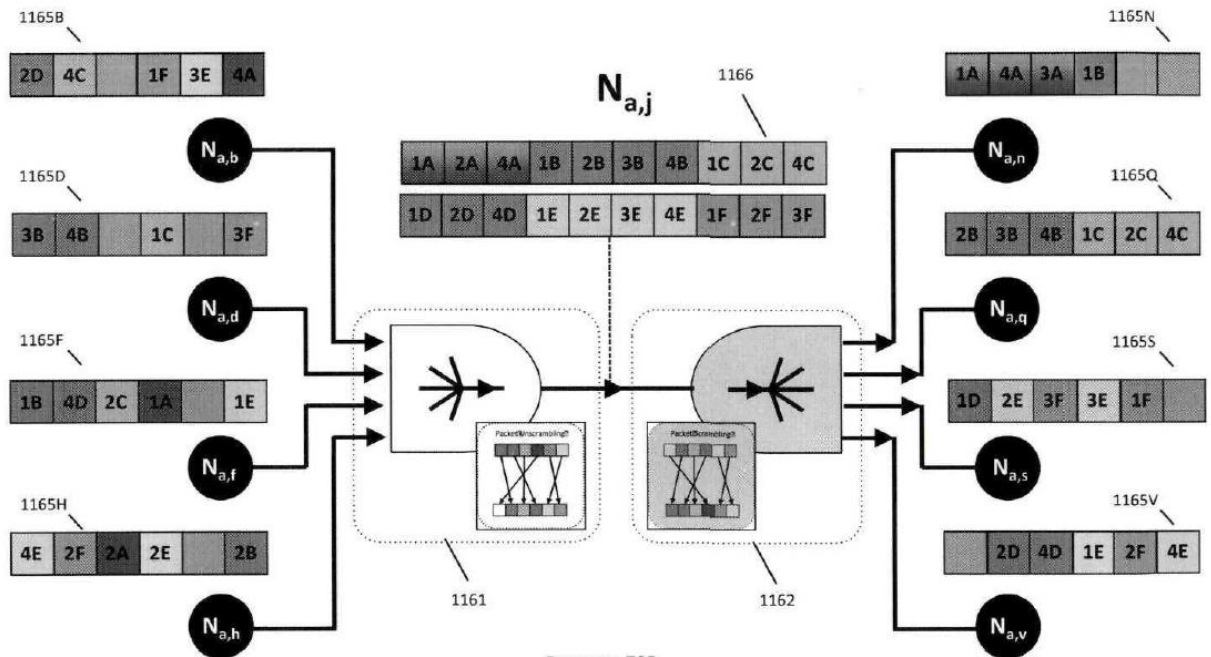


Рисунок 79В

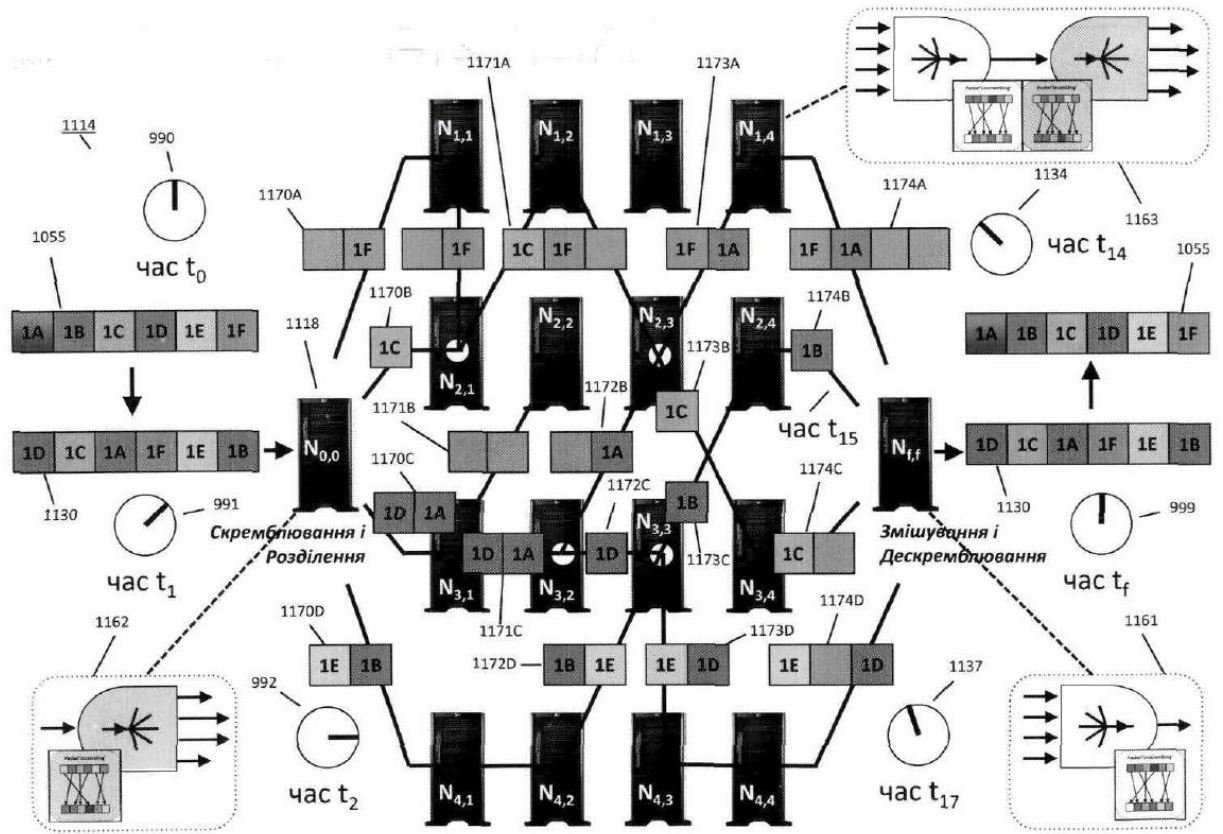


Рисунок 80

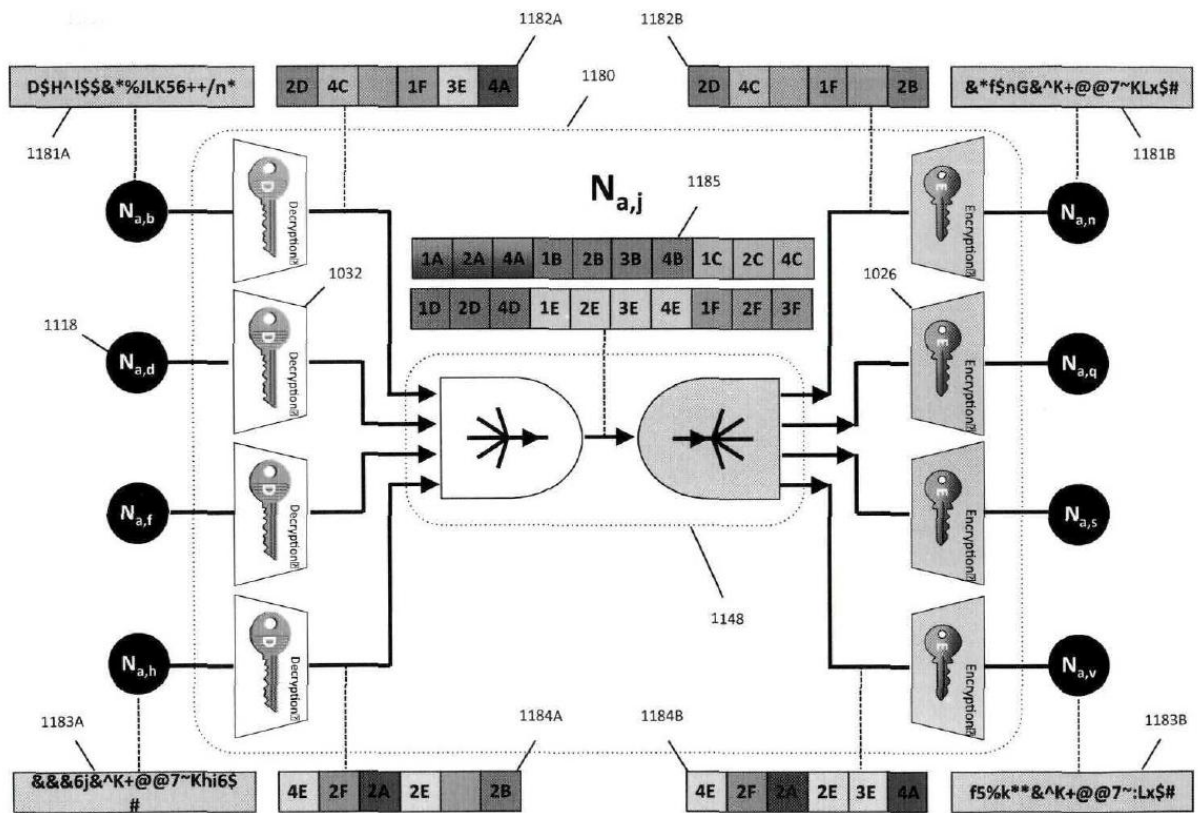
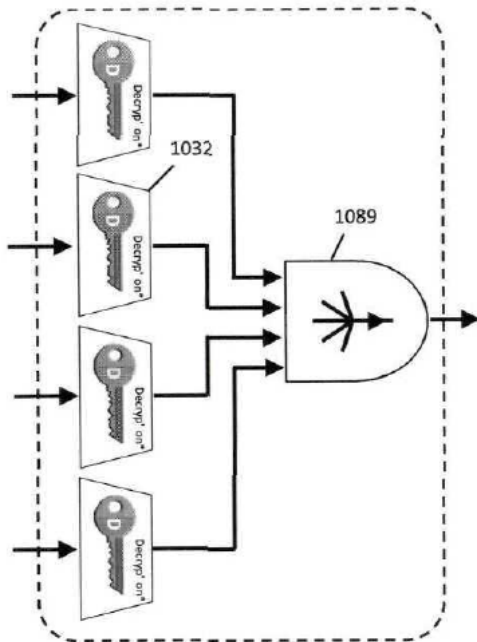


Рисунок 81А

Функції вузлового вводу



Символічне подання

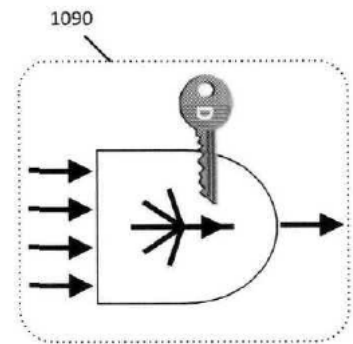
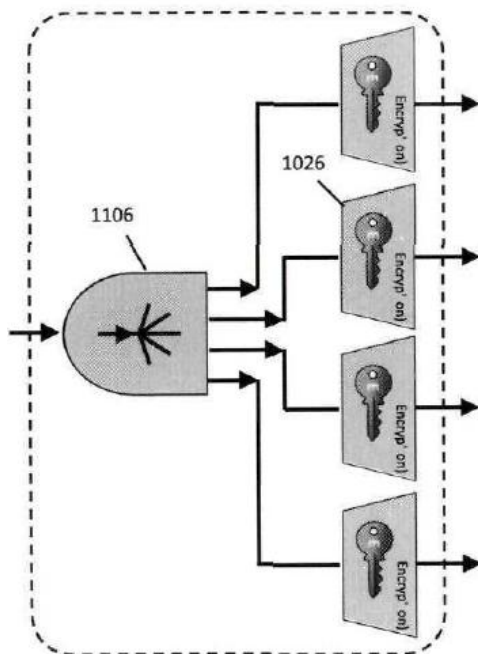


Рисунок 81В

Функції вузлового виводу



Символічне подання

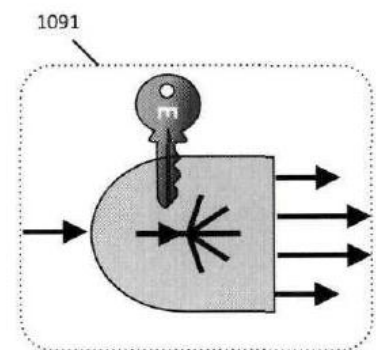
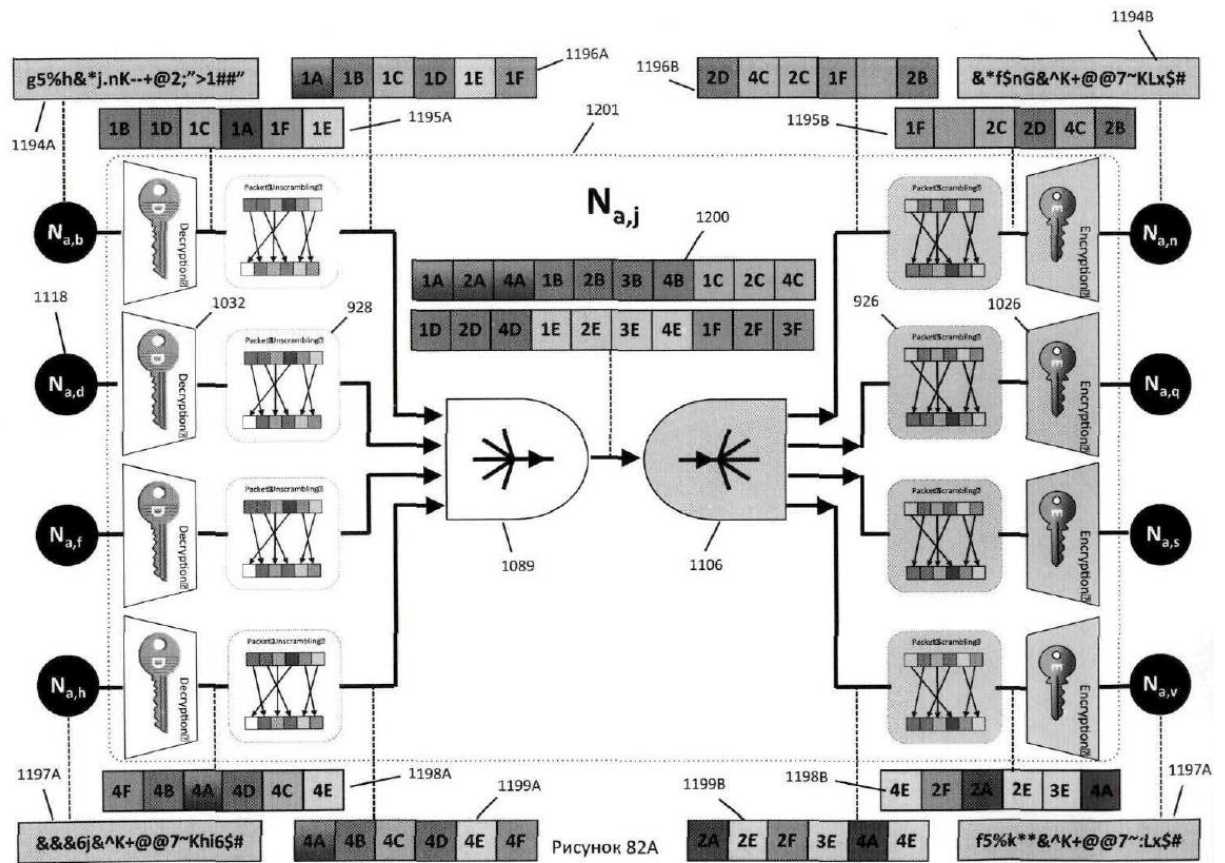


Рисунок 81С



Функції вузлового вводу DUM

Символічне подання DUM

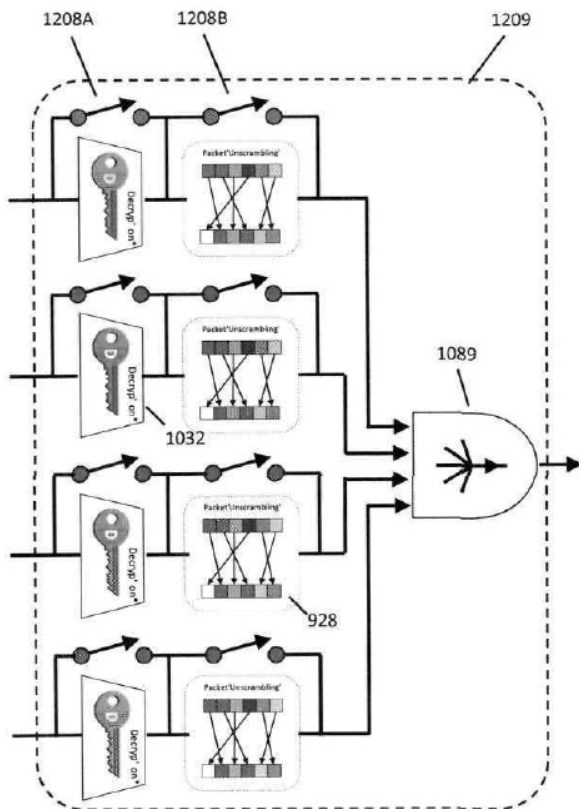
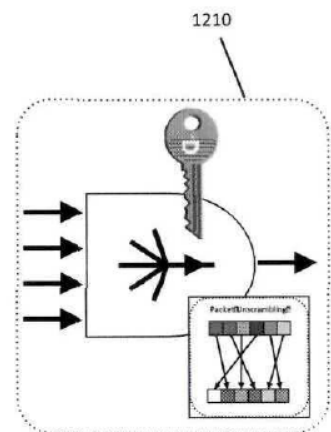
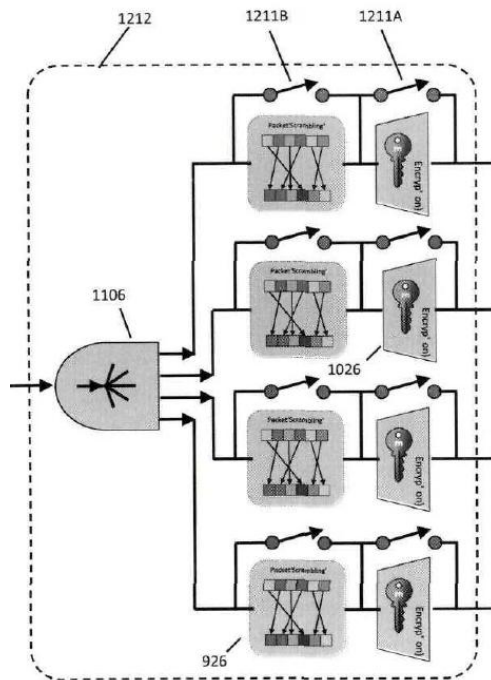


Рисунок 82В



Функції вузлового виводу SSE



Символічне подання

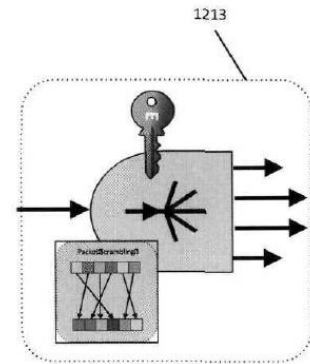


Рисунок 82С

Функція решітчастого повторного пакетування на вузлі SDNP

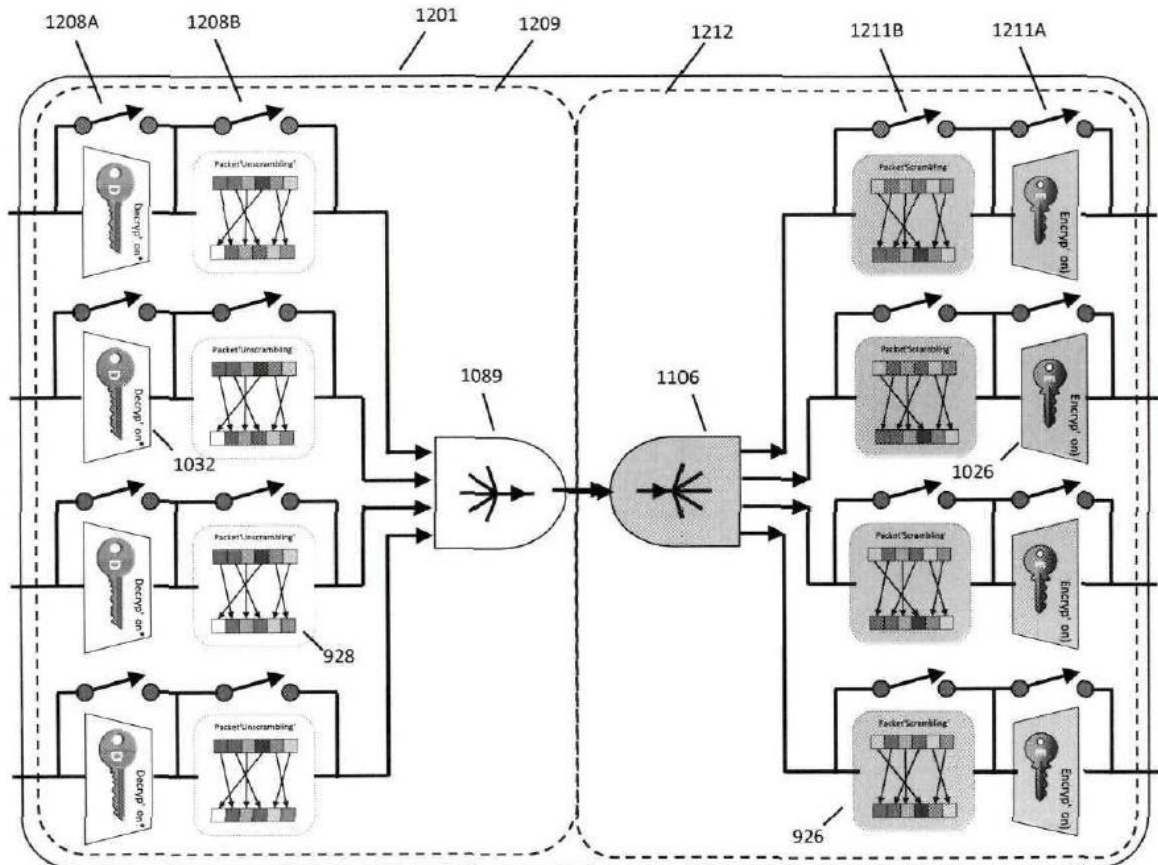


Рисунок 83А

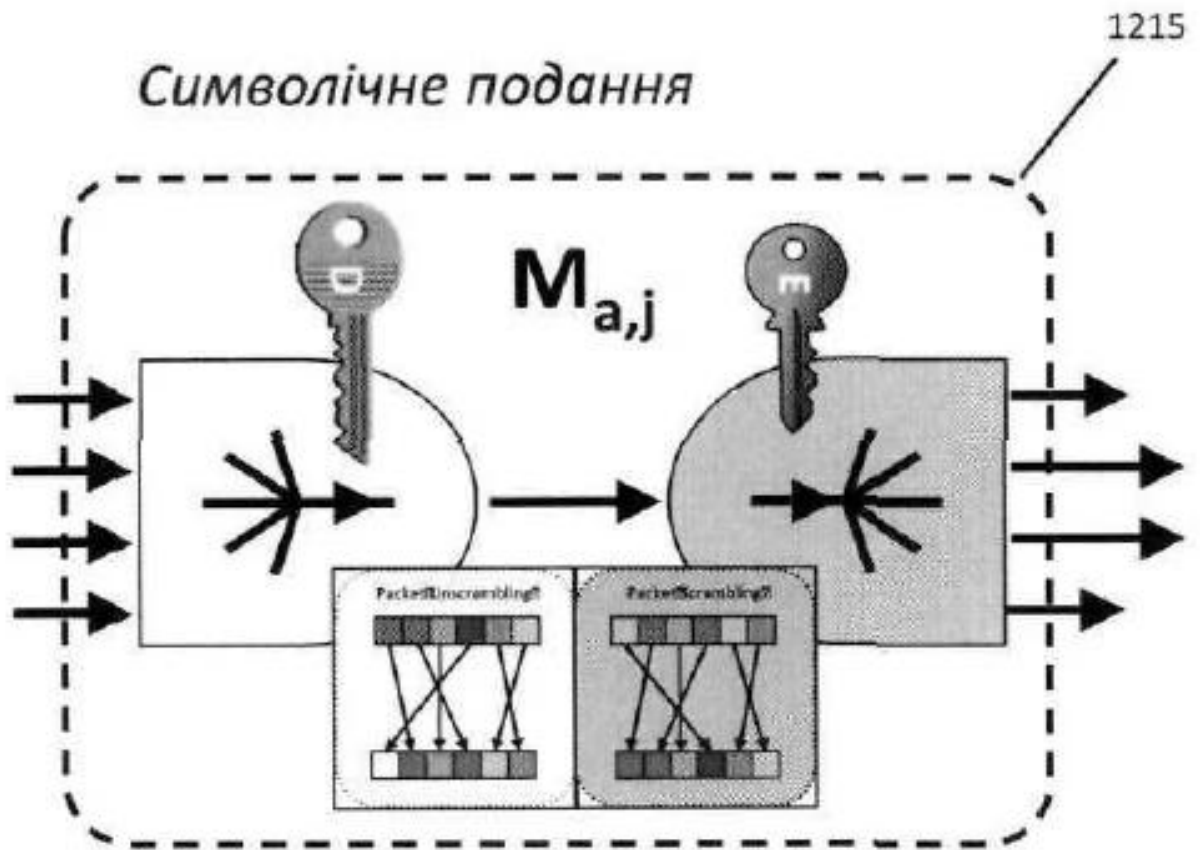


Рисунок 83В

Наскрізний прохід одношляхової маршрутизації

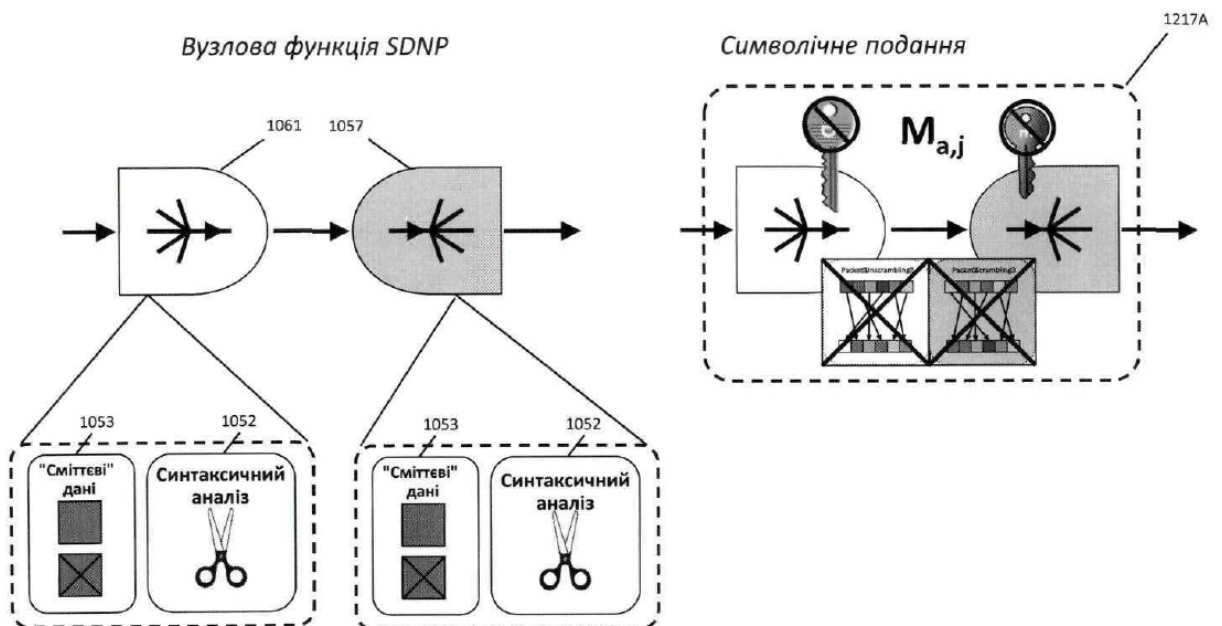


Рисунок 83С

Надлишкове дублювання маршруту

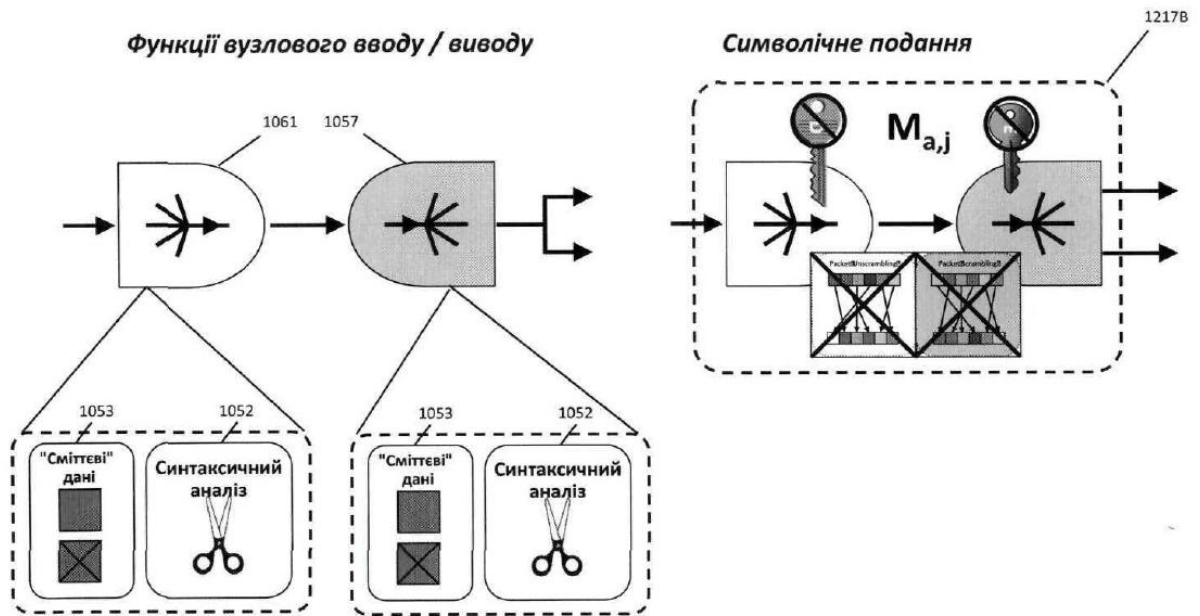


Рисунок 83D

Скремблювання при одношляховій маршрутизації

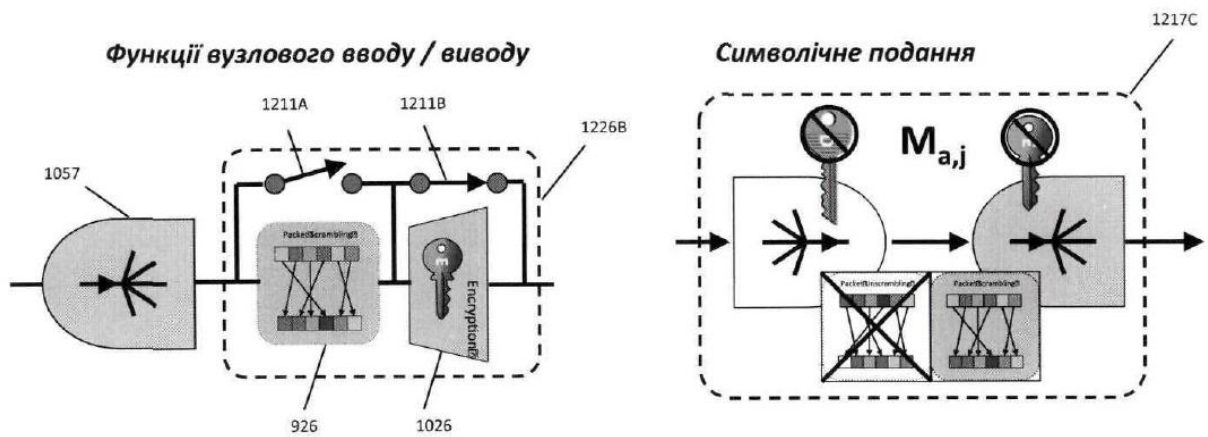
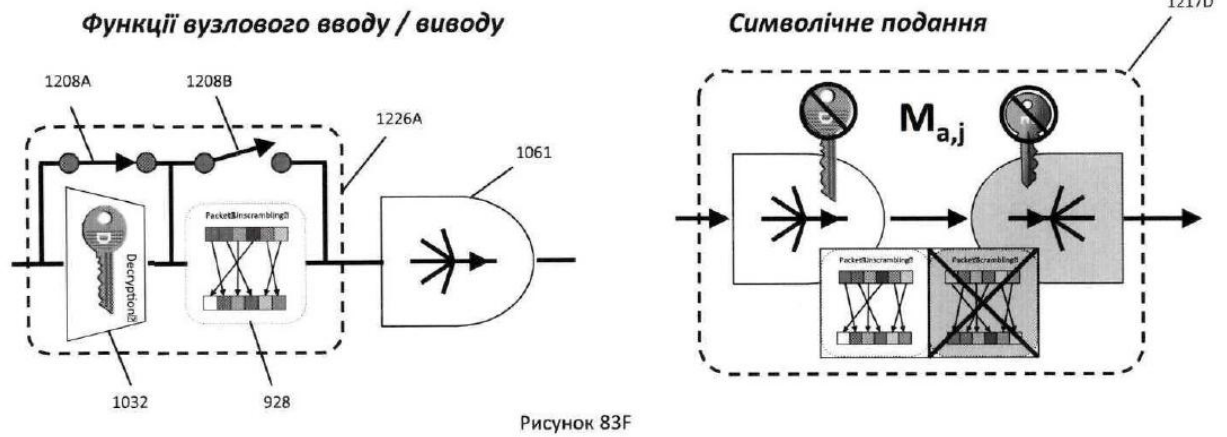
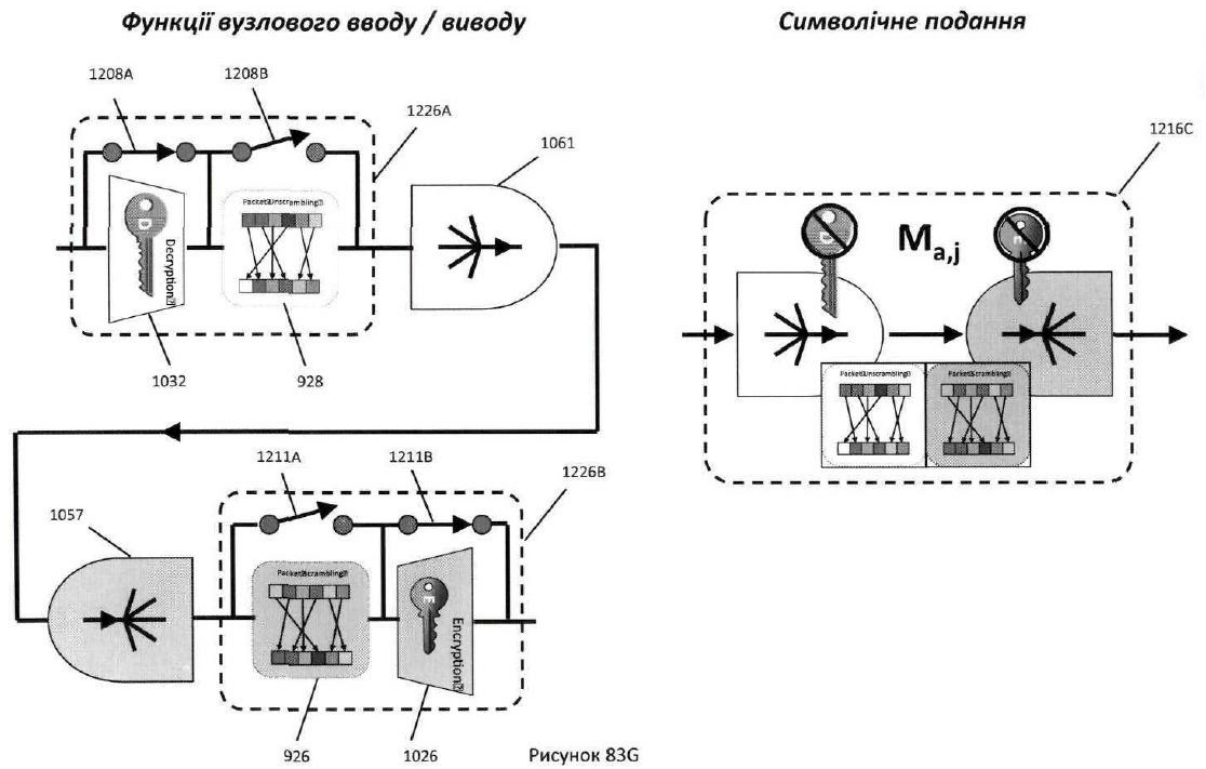


Рисунок 83E

Дескремблювання при одношляховій маршрутизації



Повторне скремблювання при одношляховій маршрутизації



Шифрування при одношляховій маршрутизації

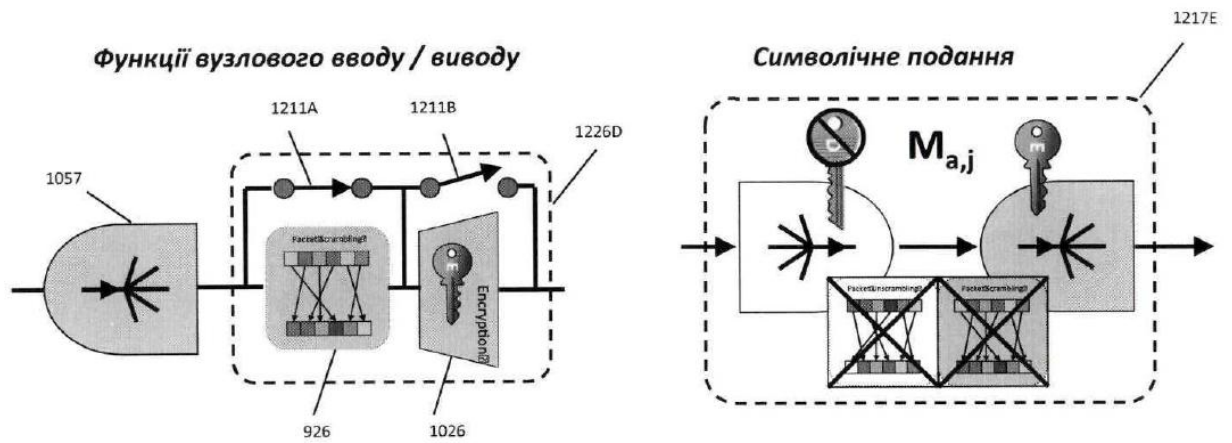


Рисунок 83H

Дешифрування при одношляховій маршрутизації

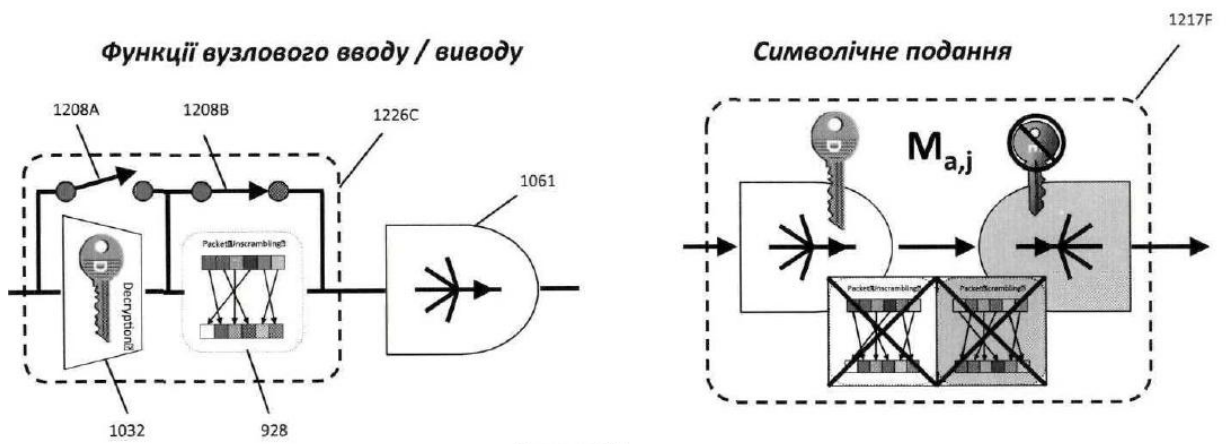
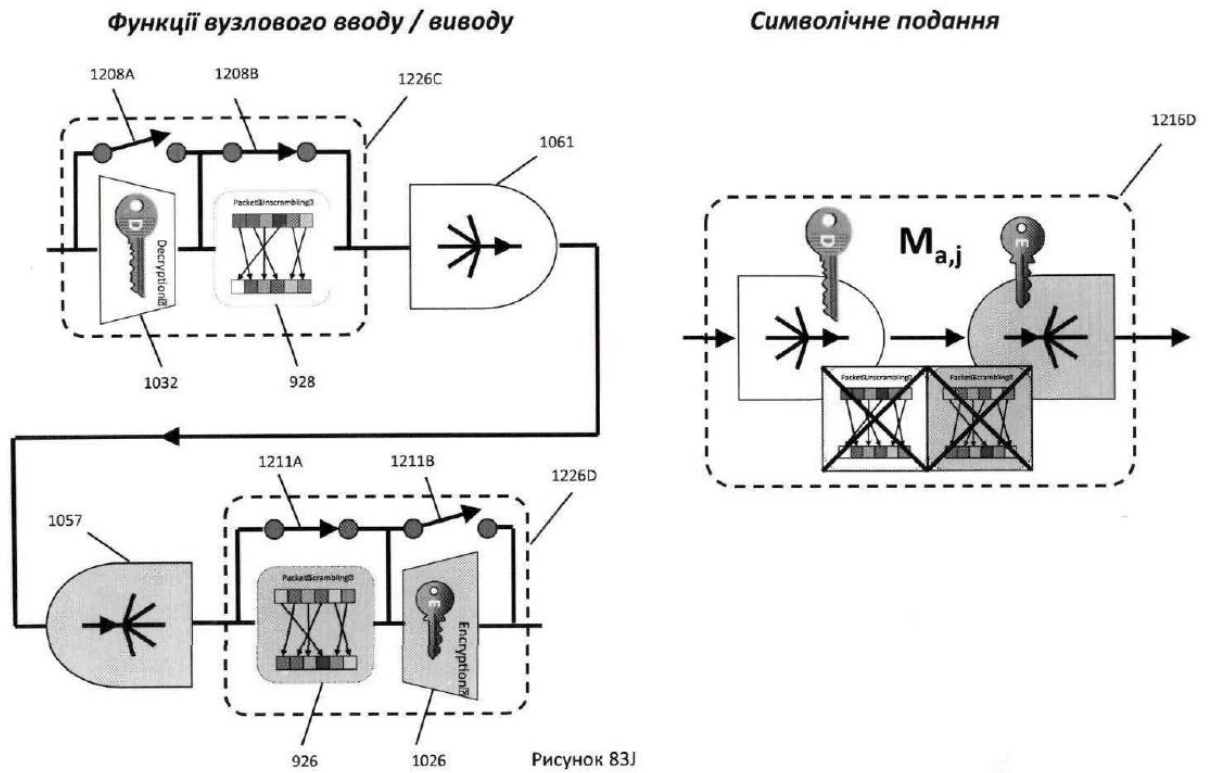
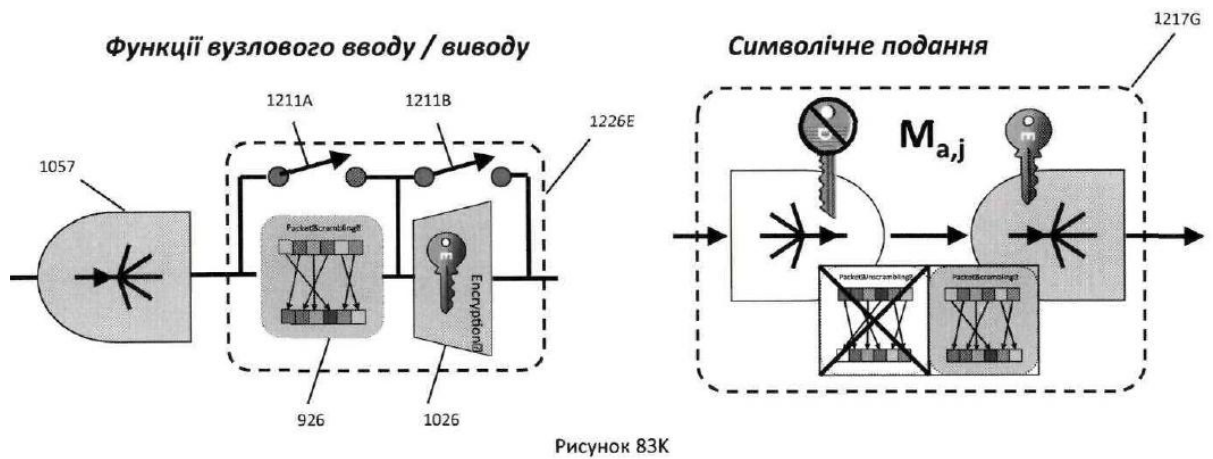


Рисунок 83I

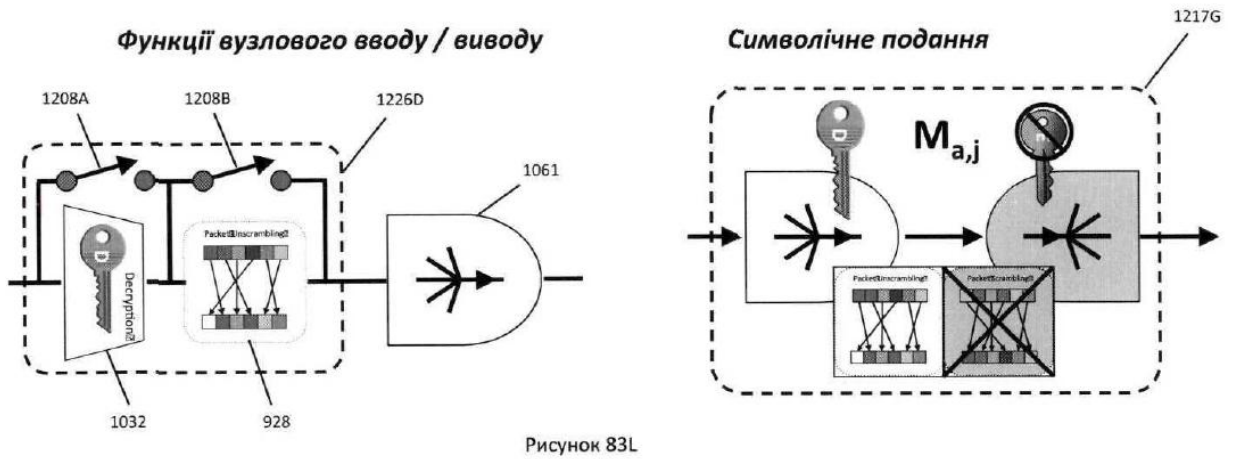
Повторне шифрування при одношляховій маршрутизації



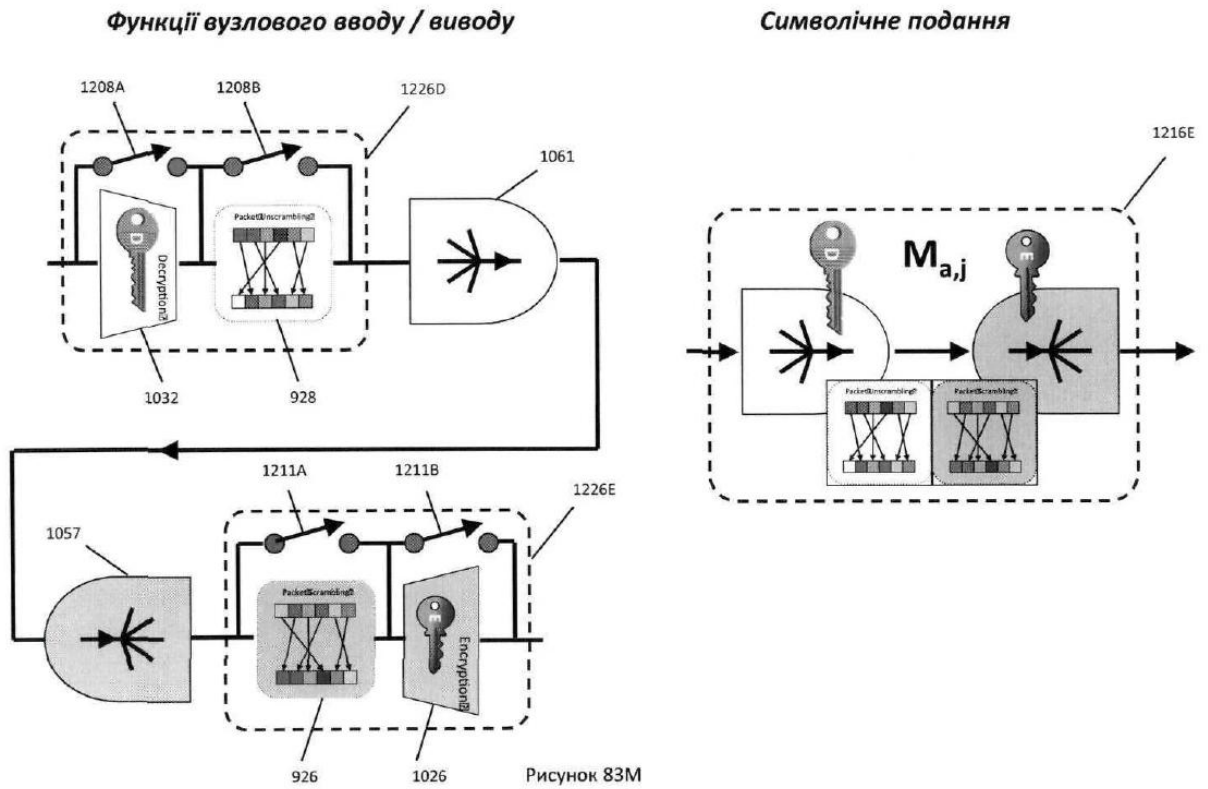
Шифрування з використанням скремблювання при одношляховій маршрутизації



Дешифрування з використанням дескремблювання при одношляховій маршрутизації



Повторне пакетування даних при одношляховій маршрутизації



Введення даних в решітчастий шлюз SDNP

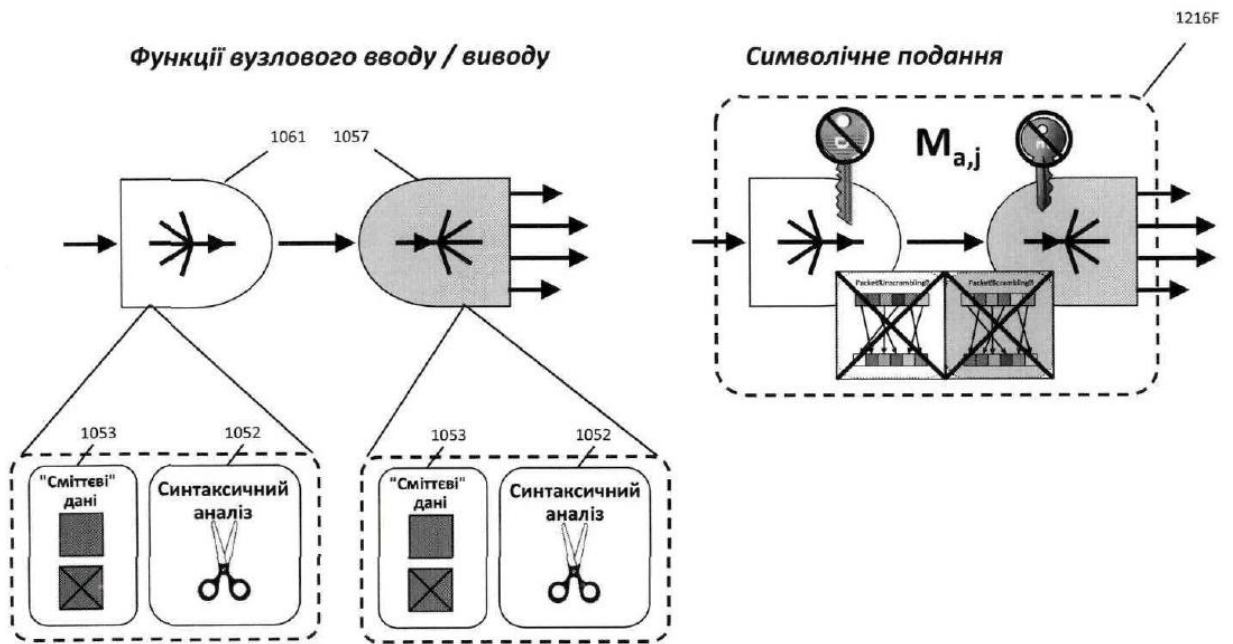


Рисунок 83N

Виведення даних із решітчастого шлюзу SDNP

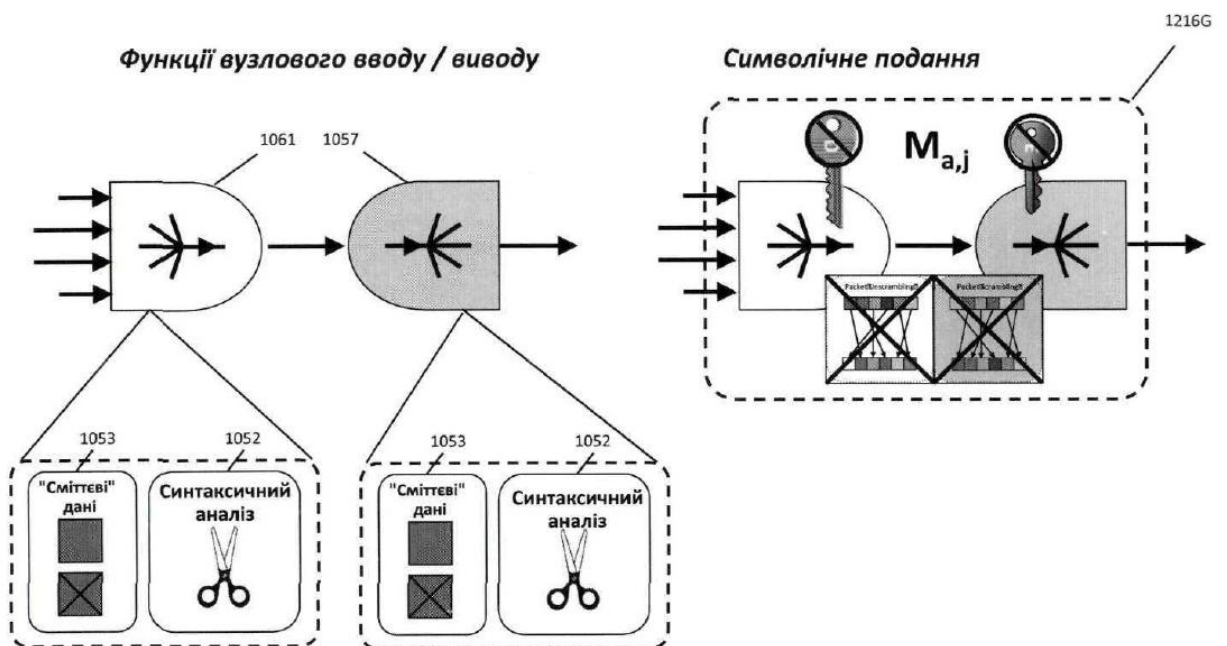
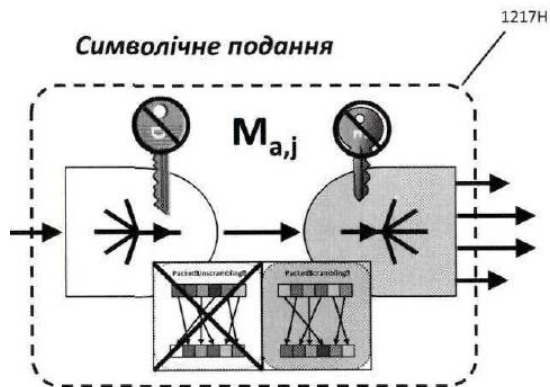


Рисунок 83O

Введення скрембльованих даних з використанням шлюзу SDNP



Виведення дескрембльованих даних з використанням шлюзу SDNP

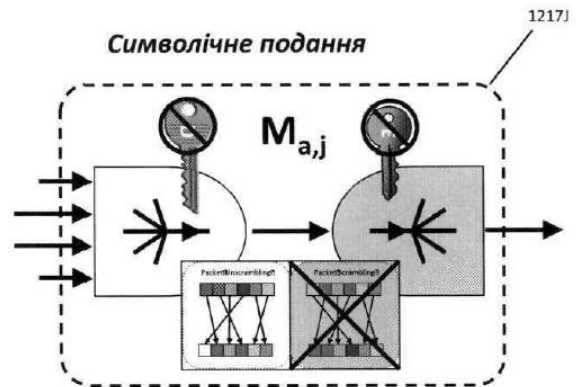
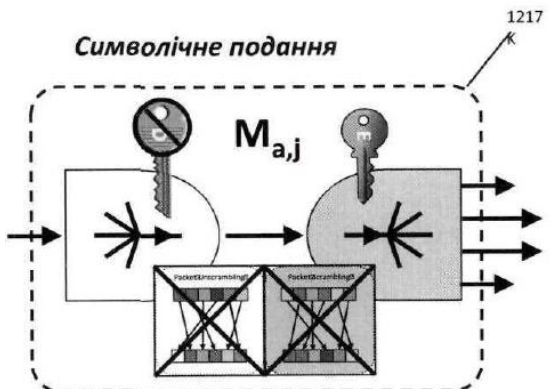


Рисунок 83P

Введення зашифрованих даних з використанням шлюзу SDNP



Виведення дешифрованих даних з використанням шлюзу SDNP

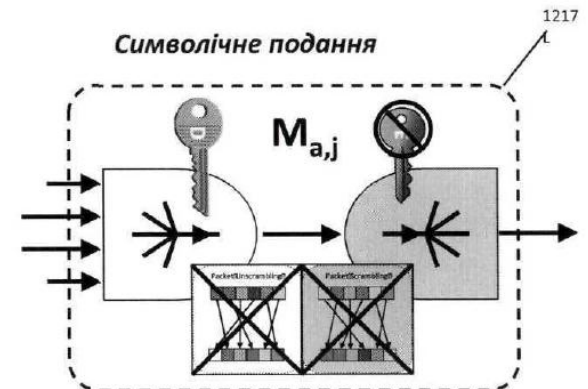
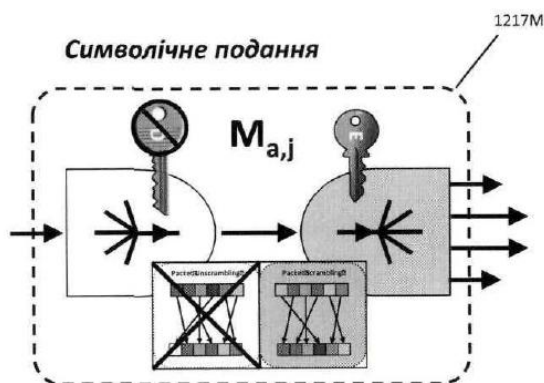


Рисунок 83Q

Введення скрембльованих зашифрованих даних з використанням шлюзу SDNP



Виведення дескрембльованих дешифрованих даних з використанням шлюзу SDNP

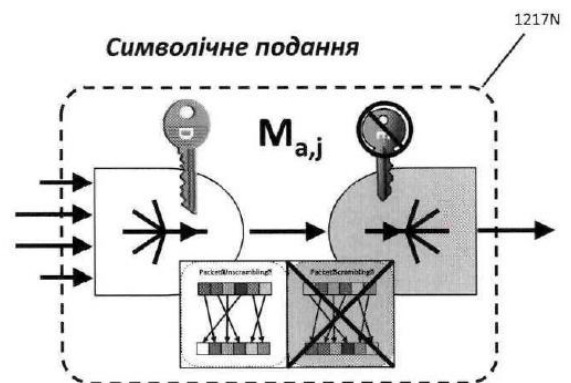
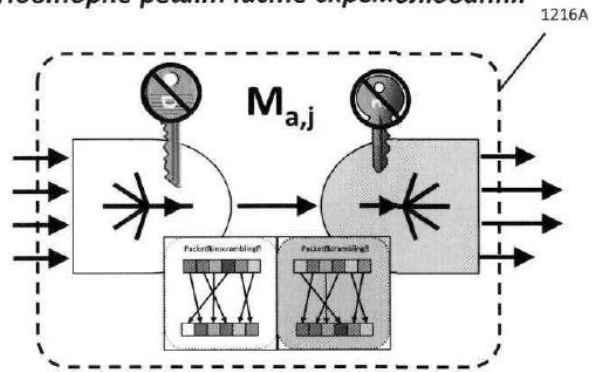


Рисунок 83R

Повторне решітчасте скремблювання



Повторне решітчасте шифрування

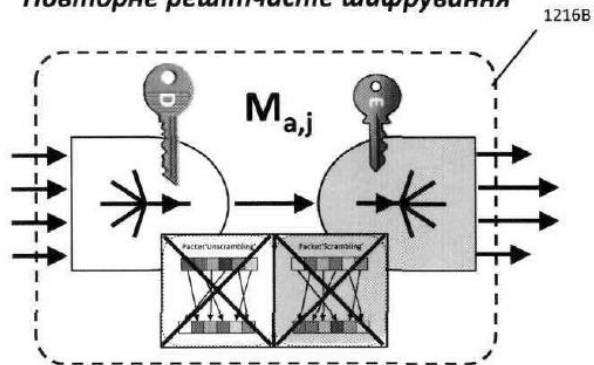


Рисунок 835

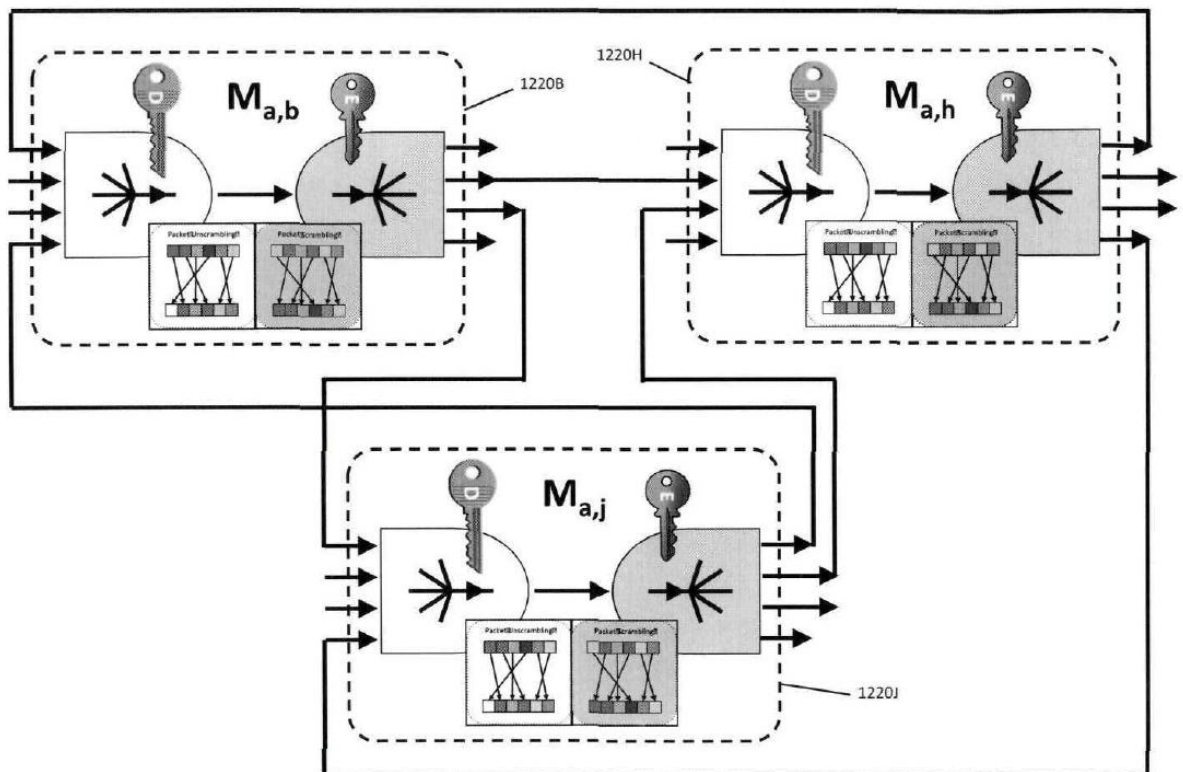


Рисунок 84A

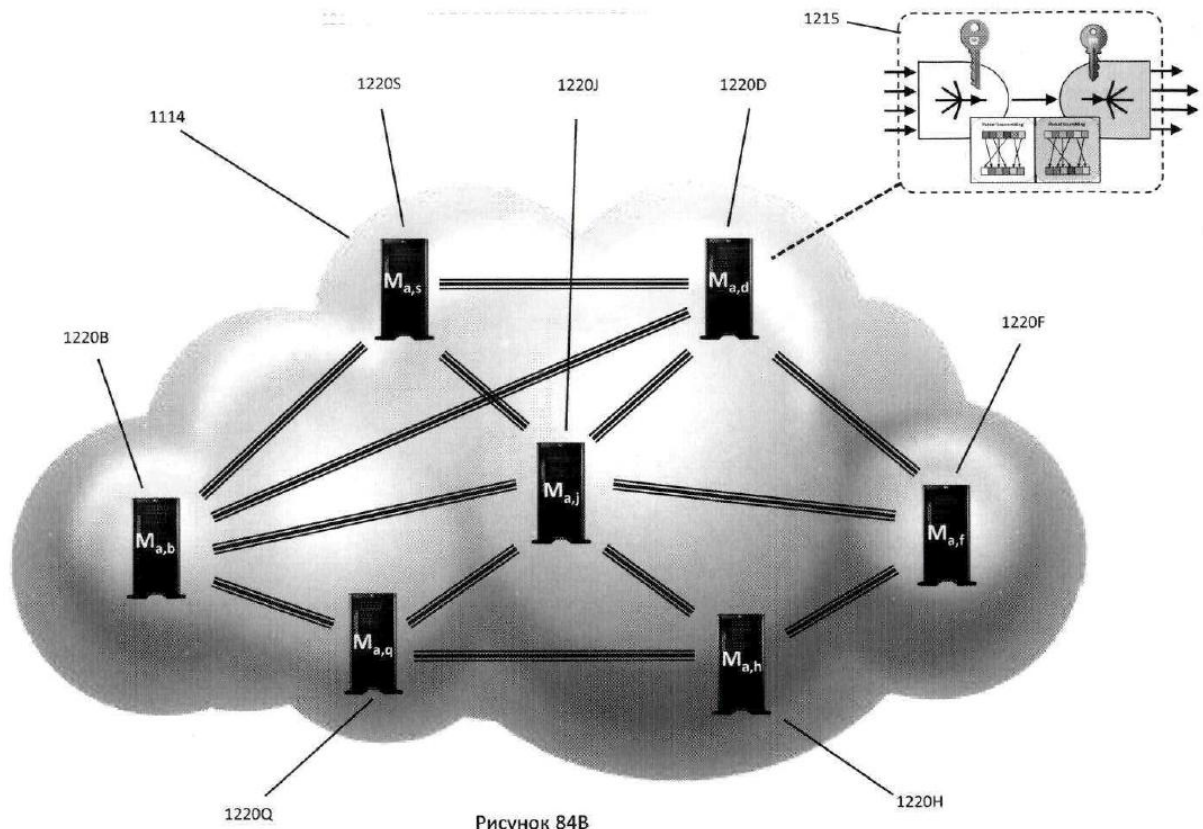


Рисунок 84В

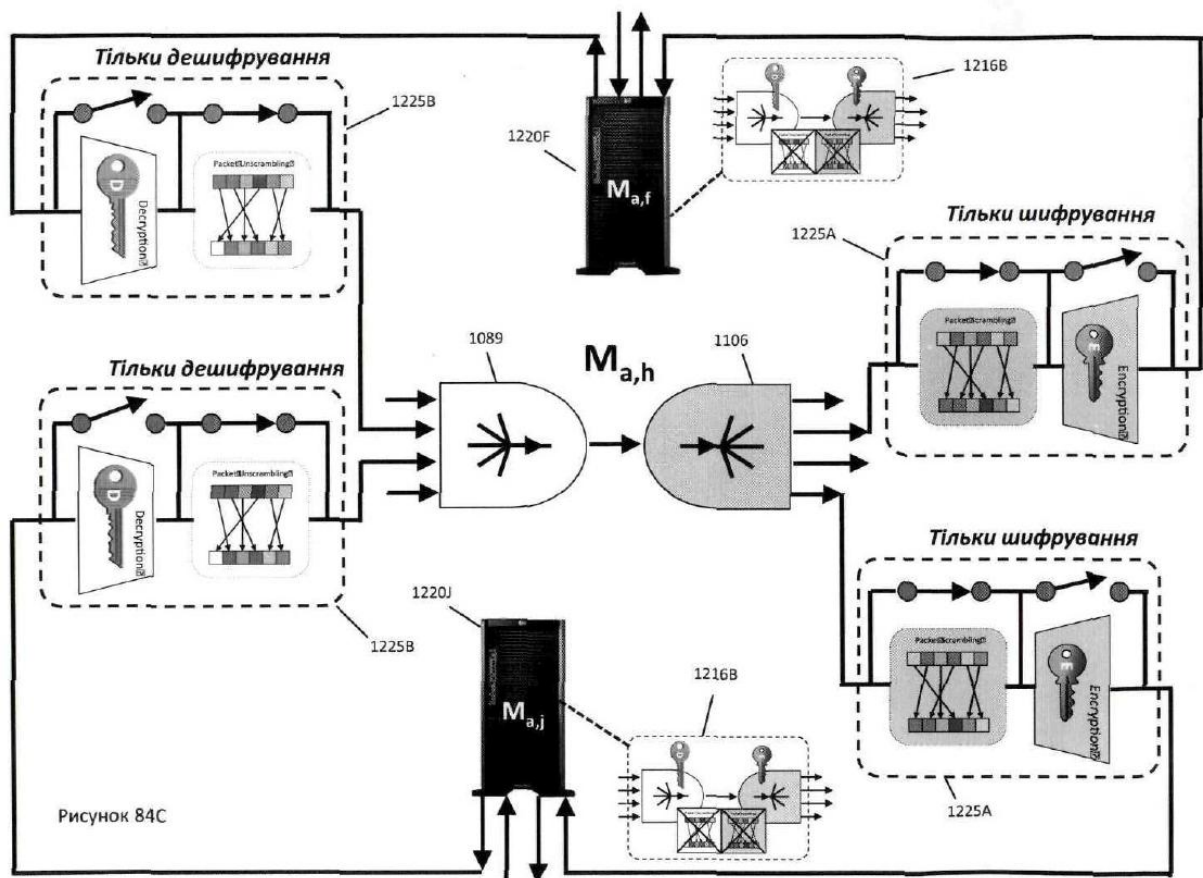


Рисунок 84С

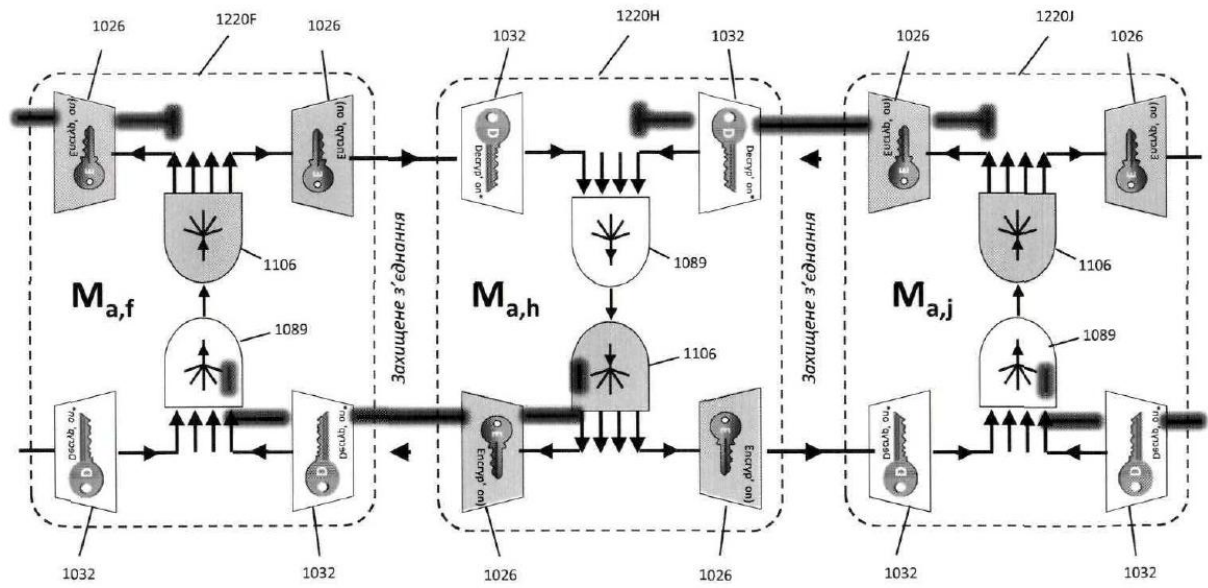


Рисунок 84D

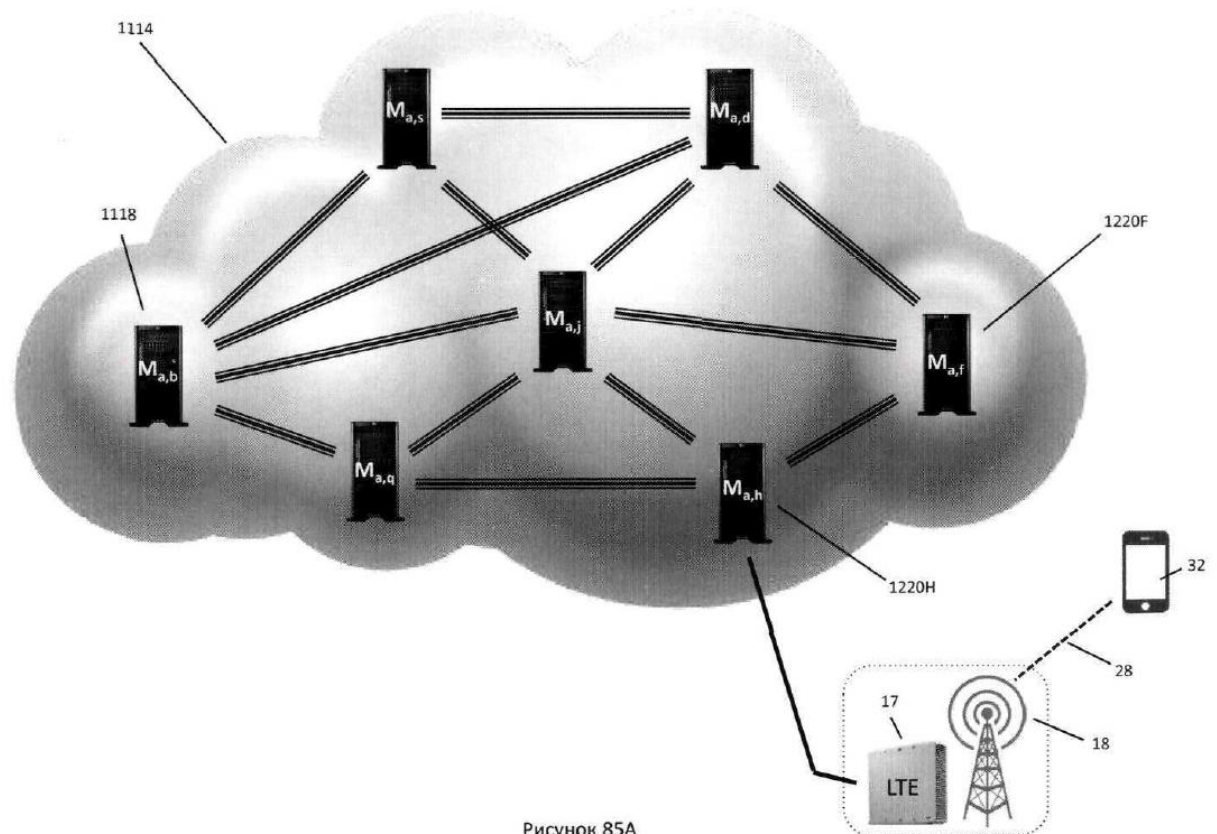
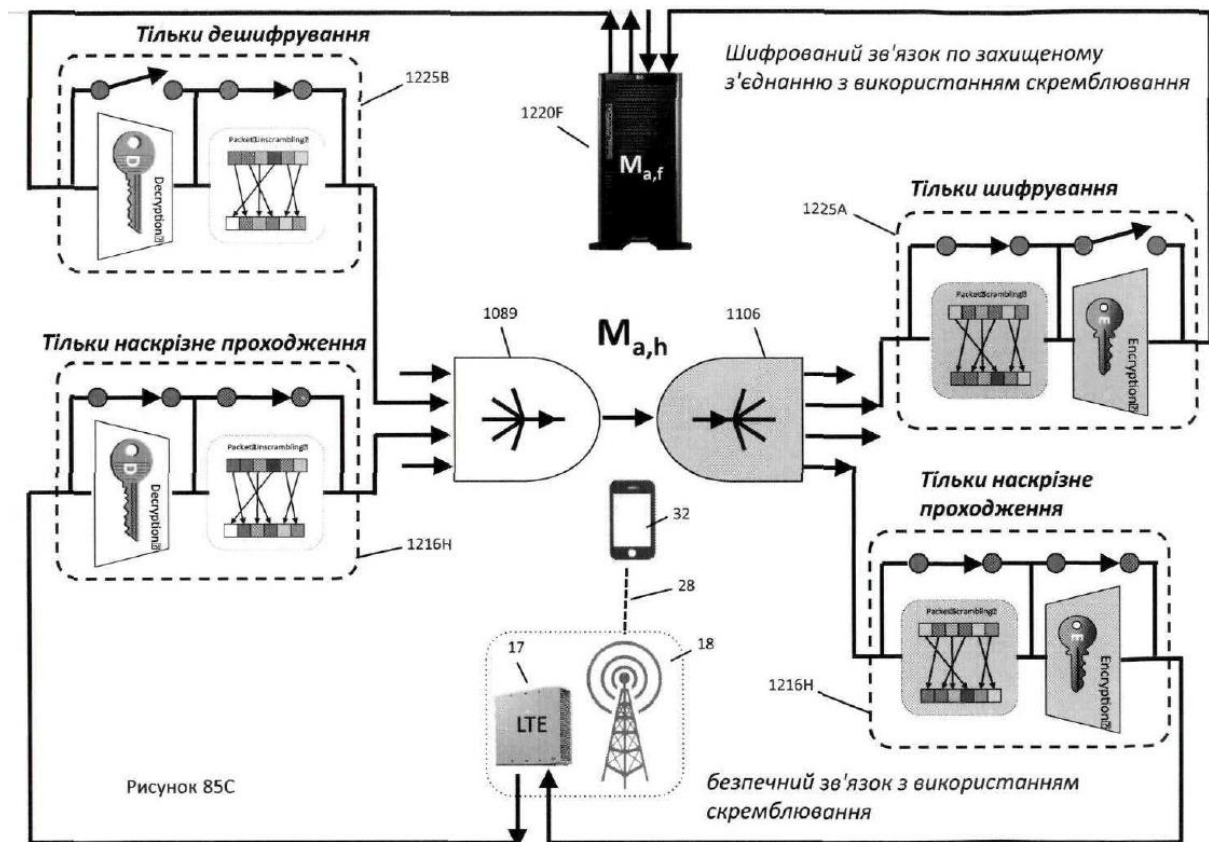
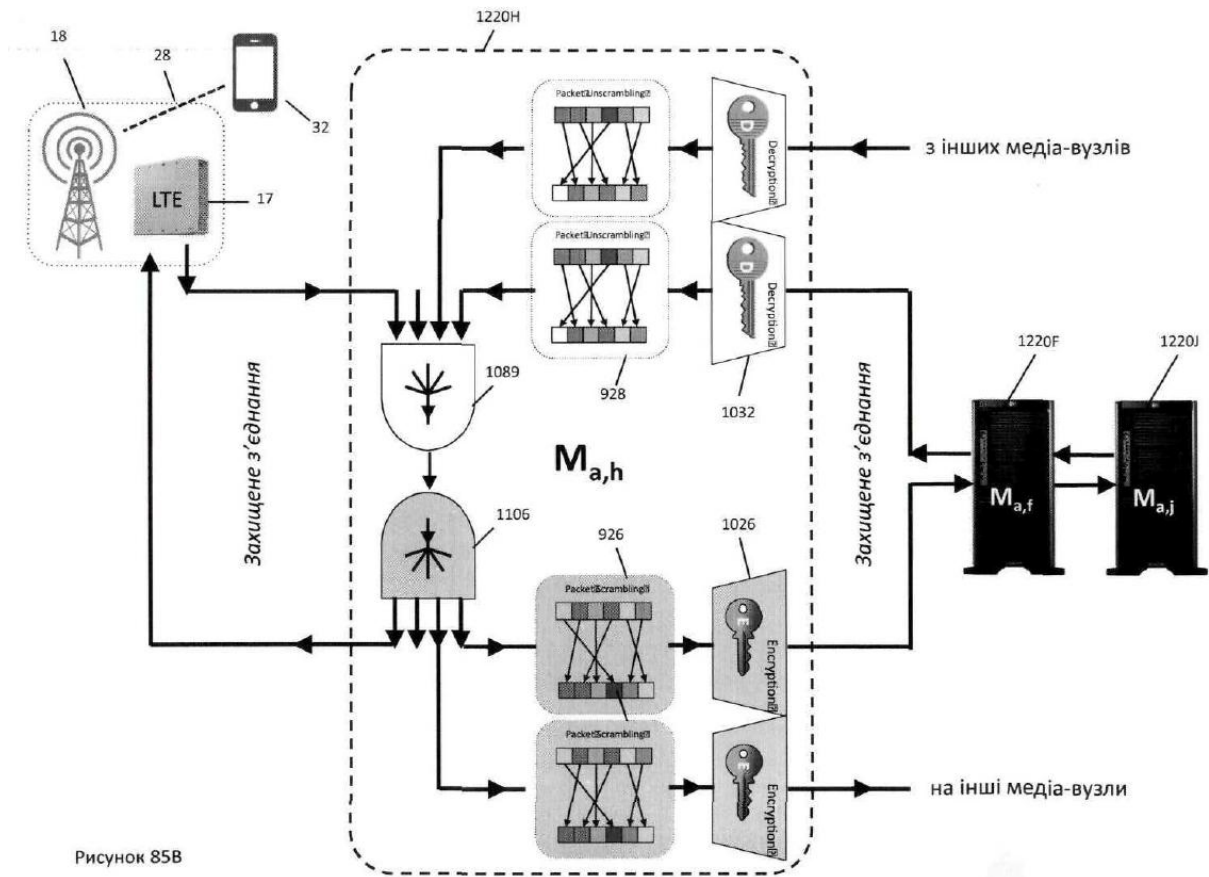
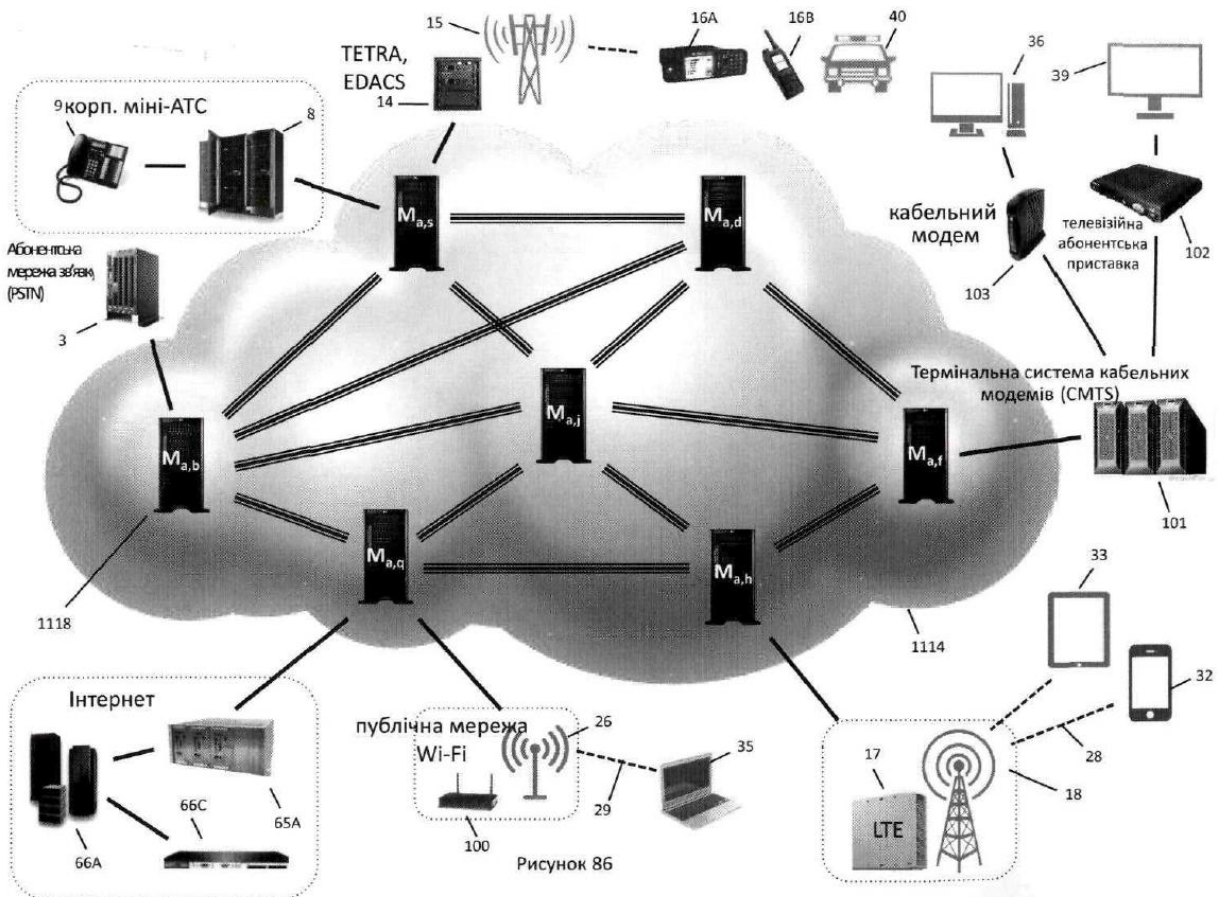
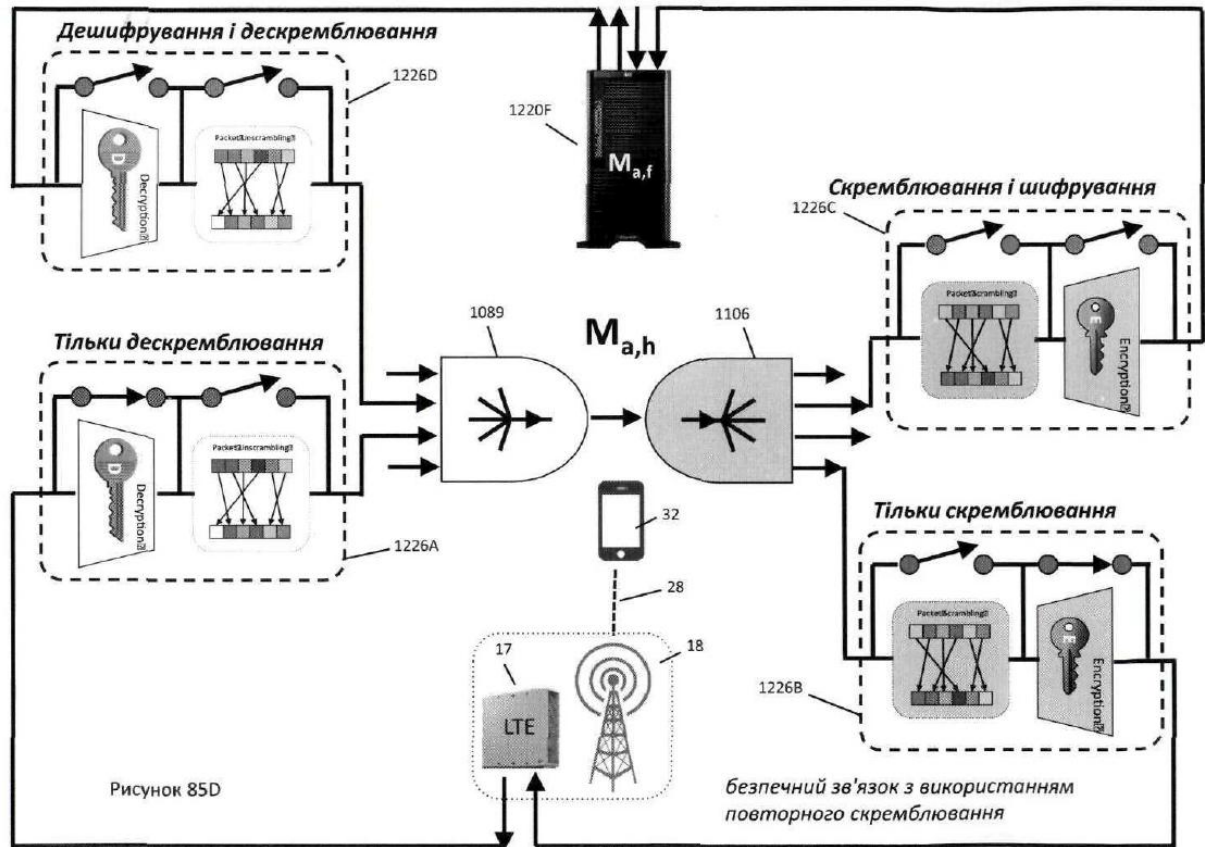
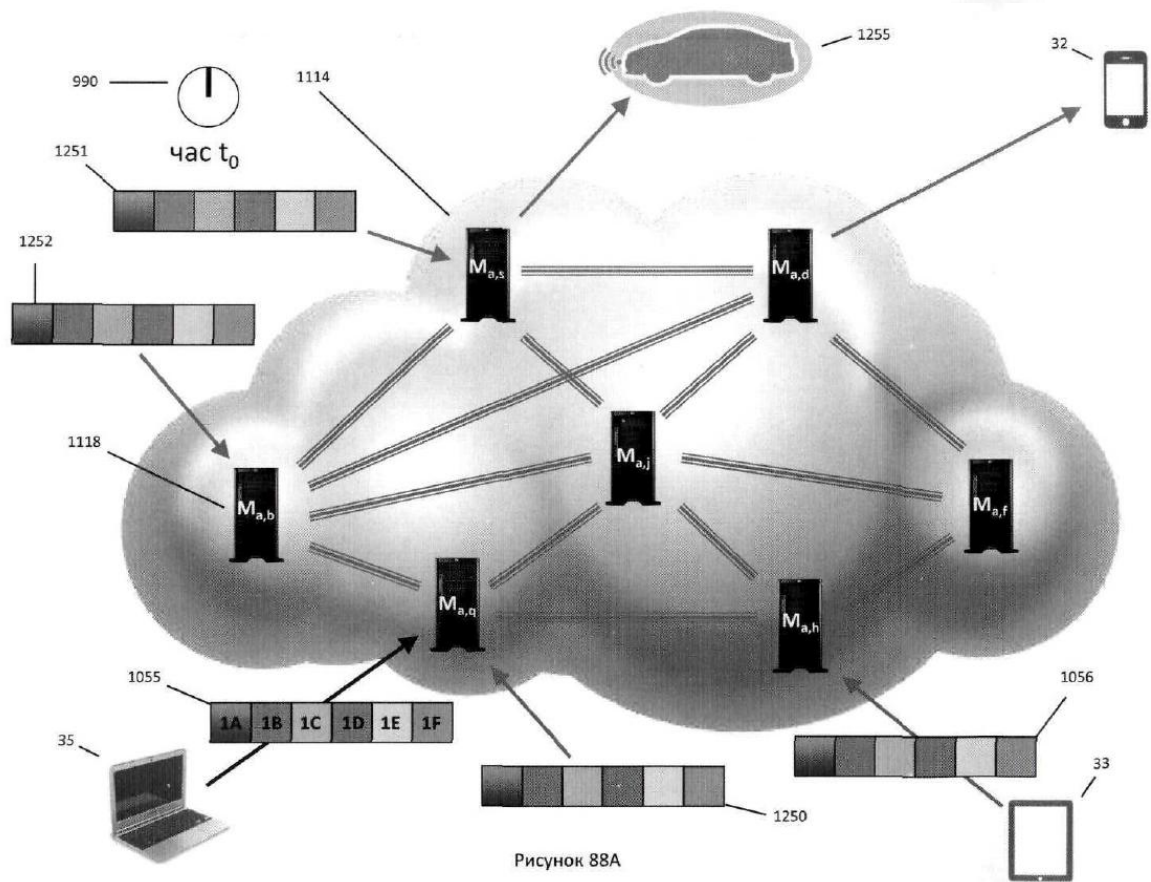
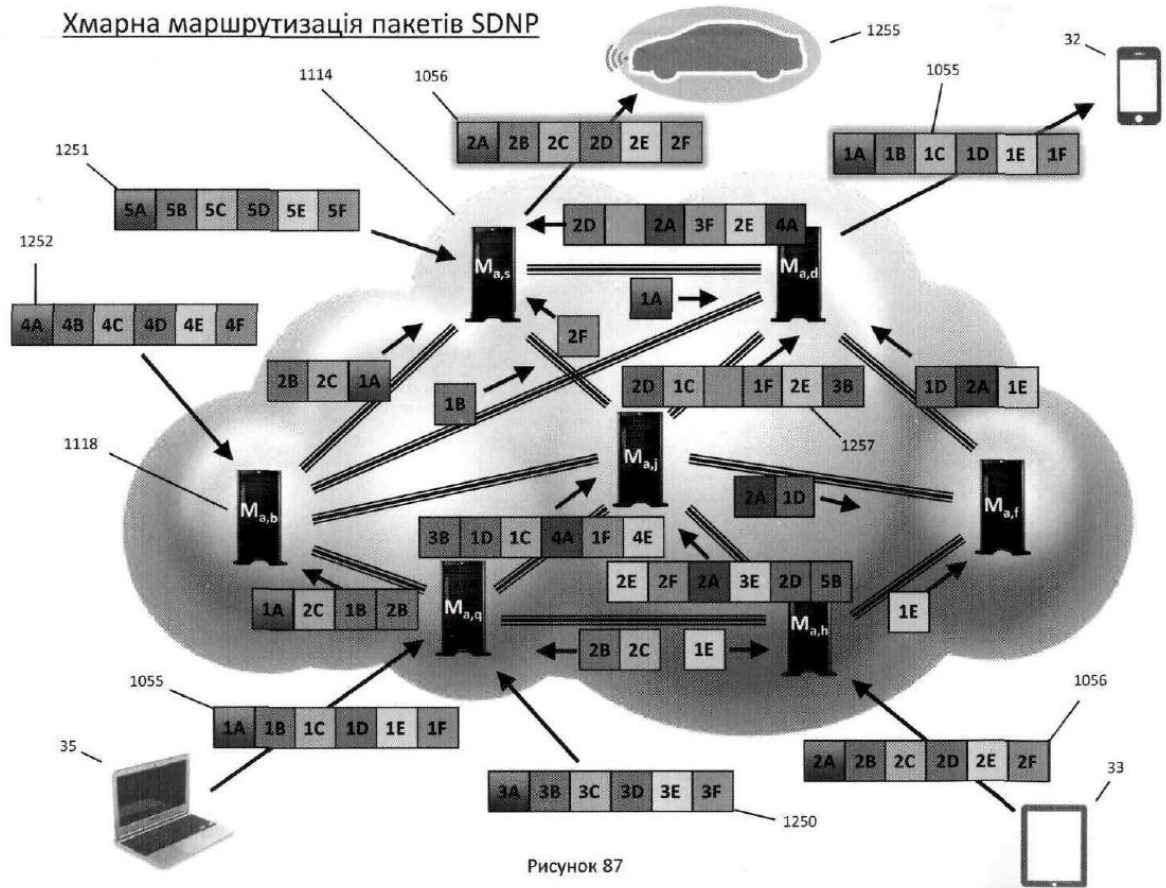


Рисунок 85A





Хмарна маршрутизація пакетів SDNP



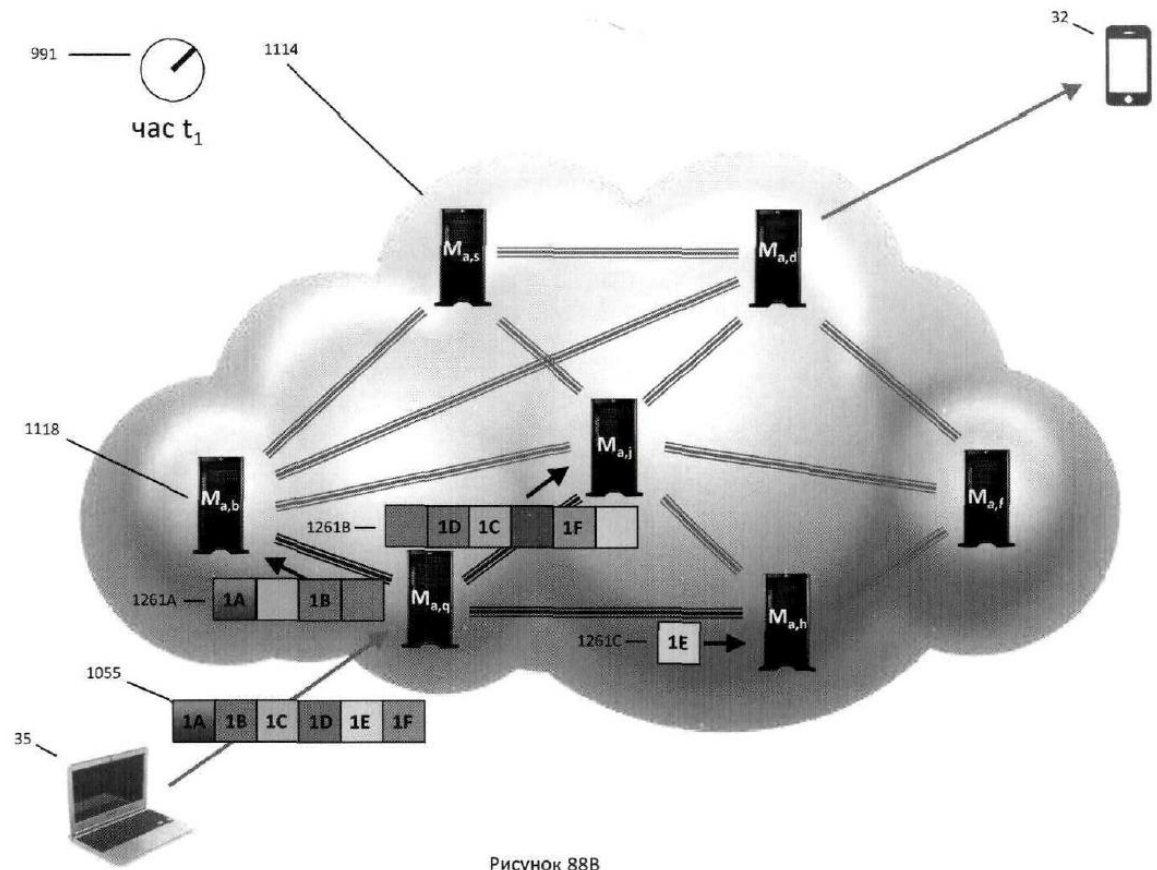


Рисунок 88B

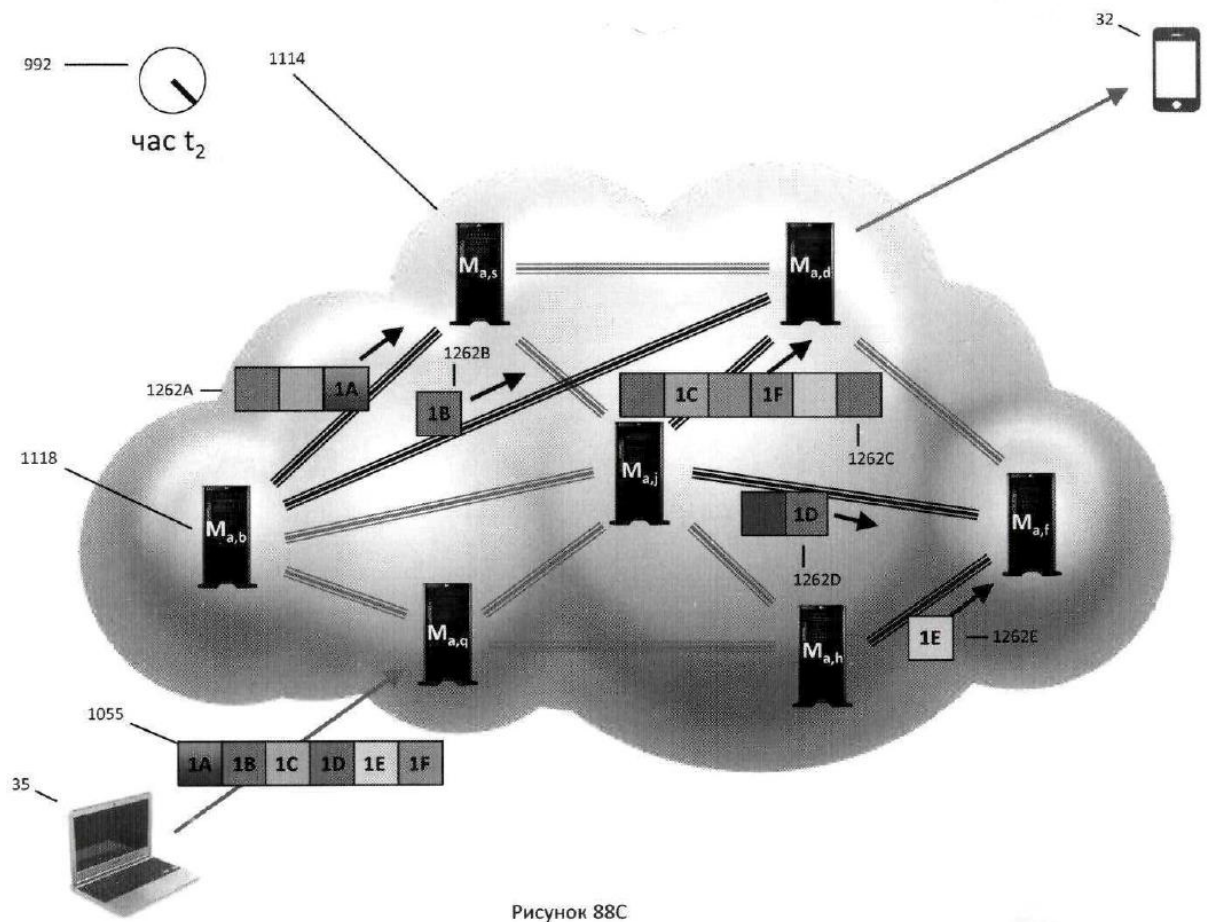


Рисунок 88C

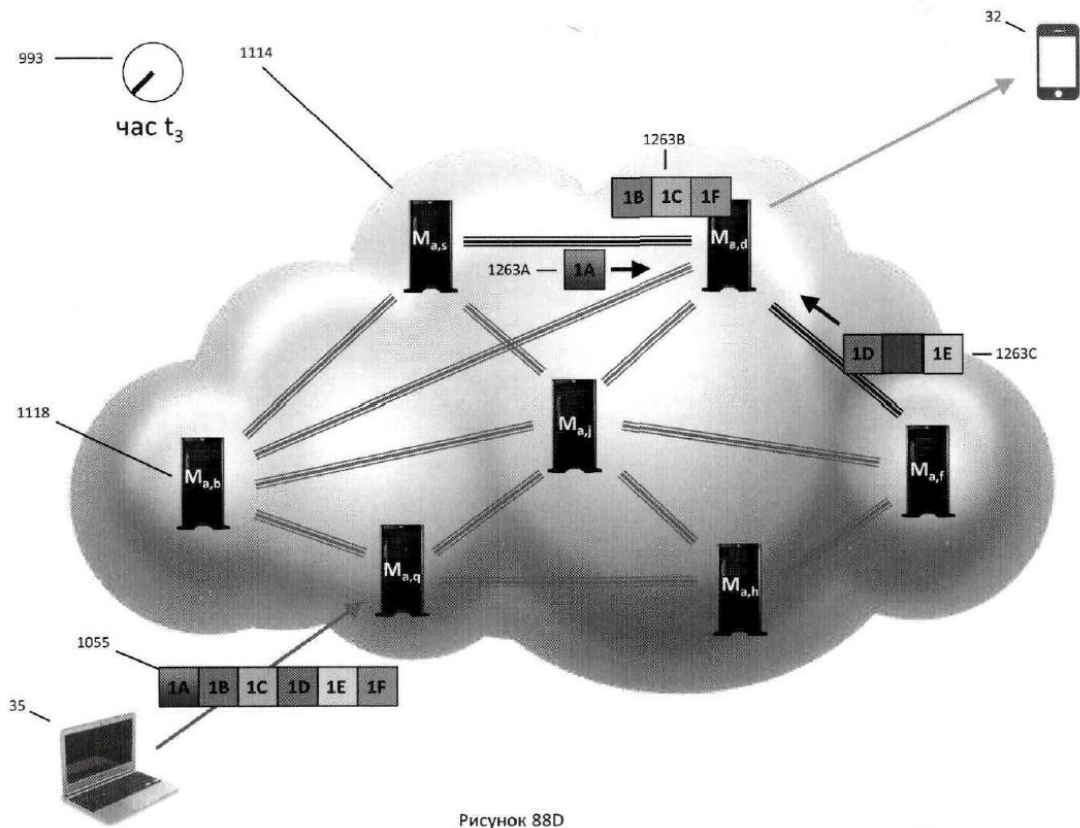


Рисунок 88D

Хмарна маршрутизація пакетів SDNP

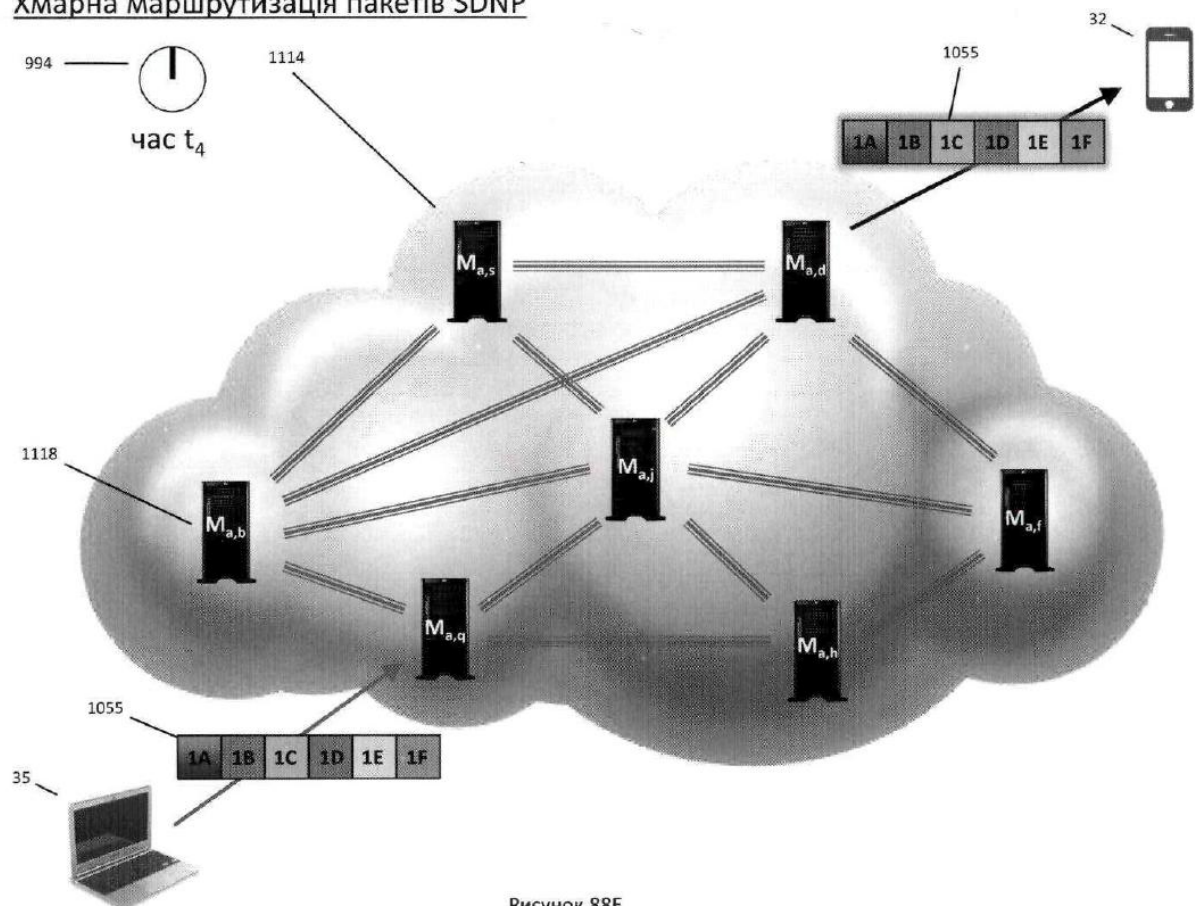
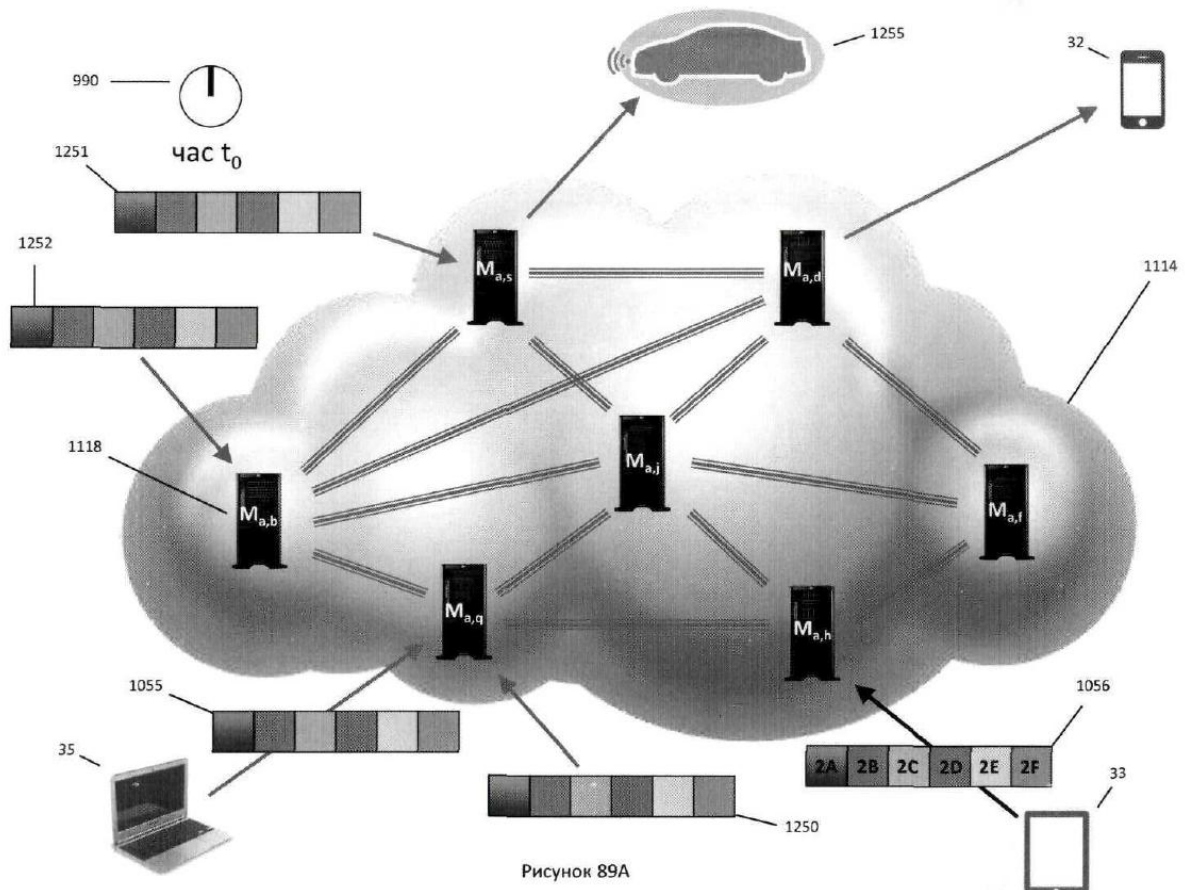
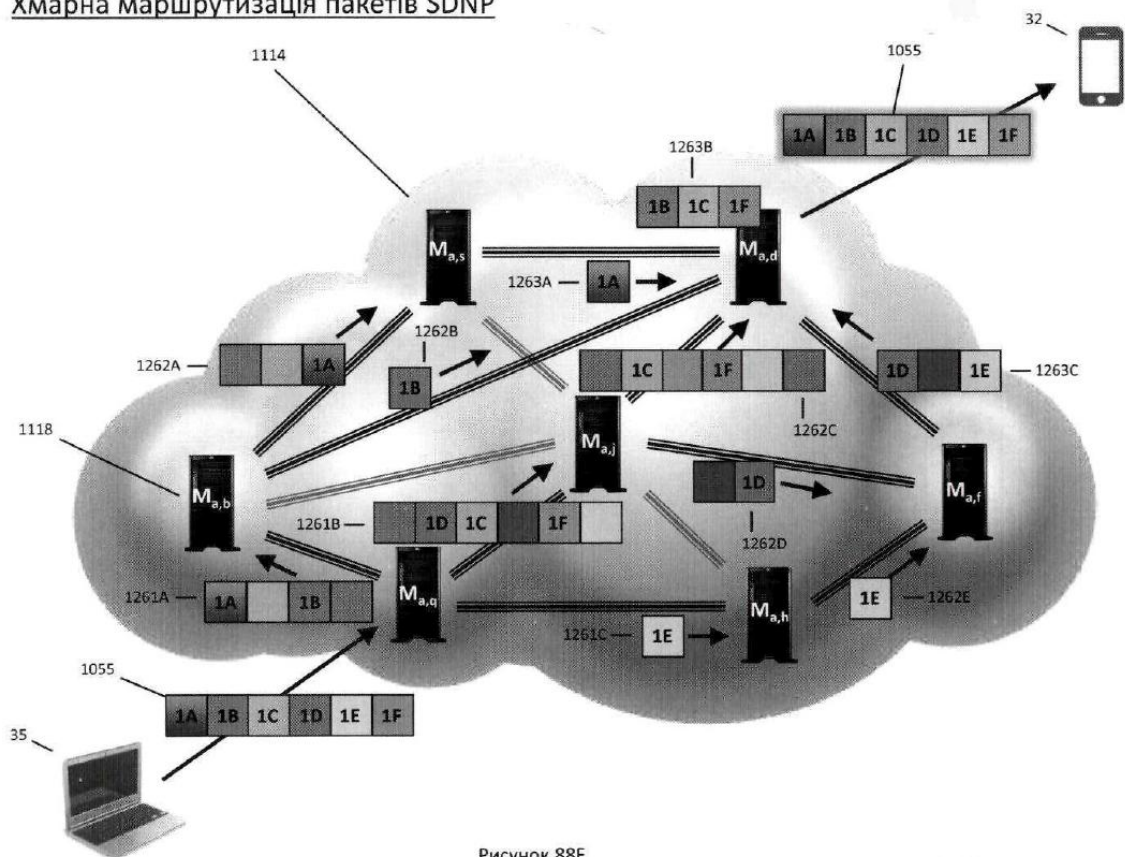
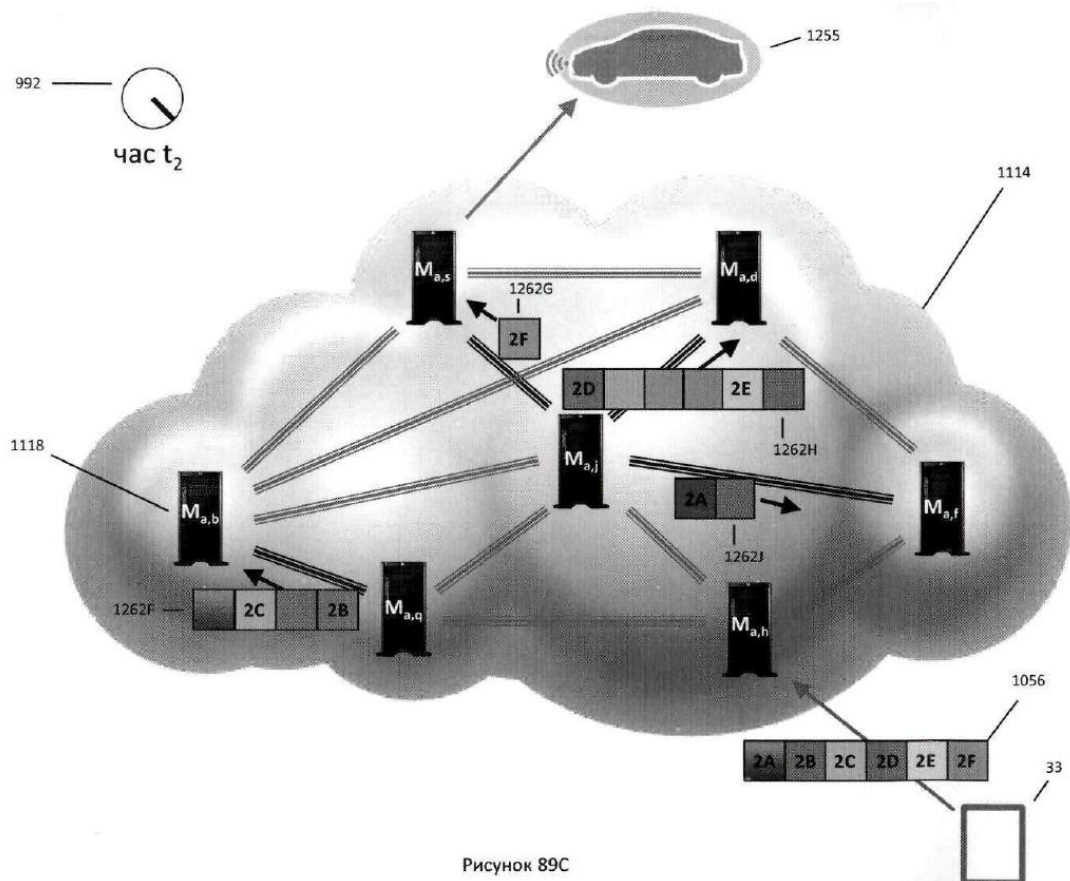
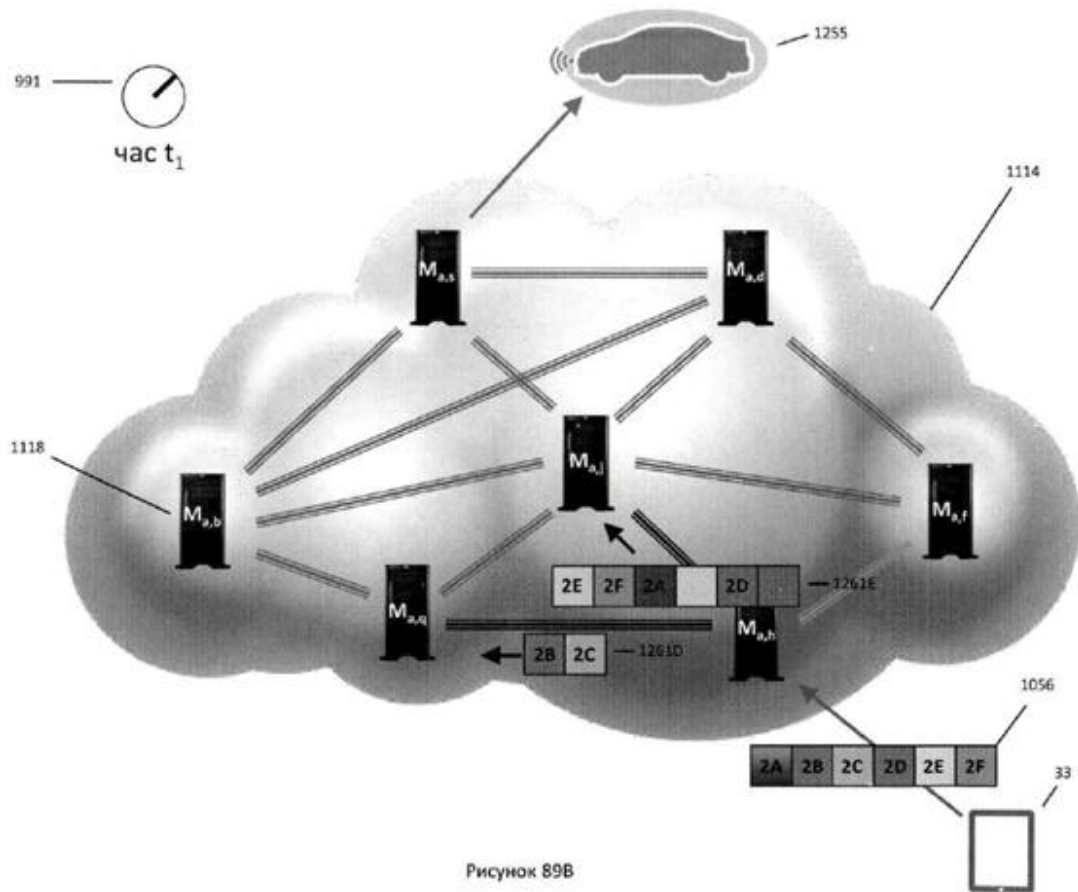
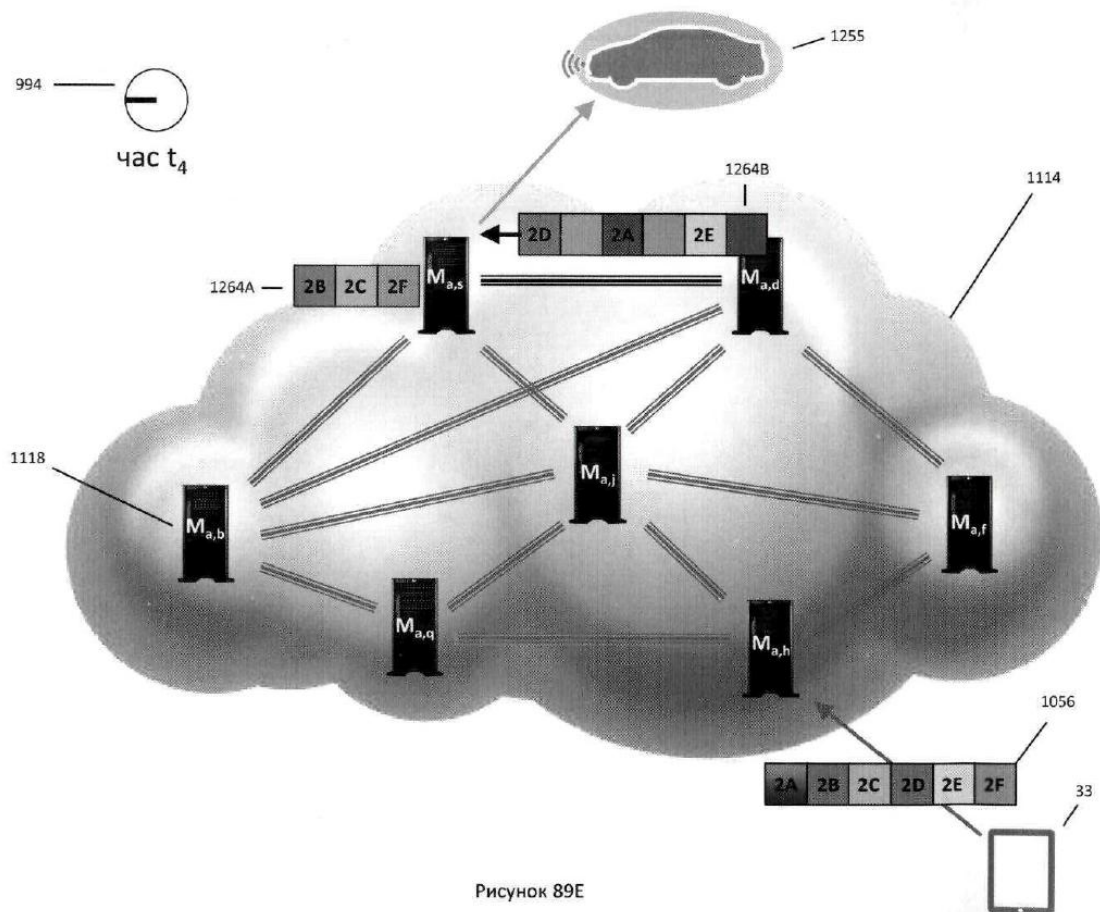
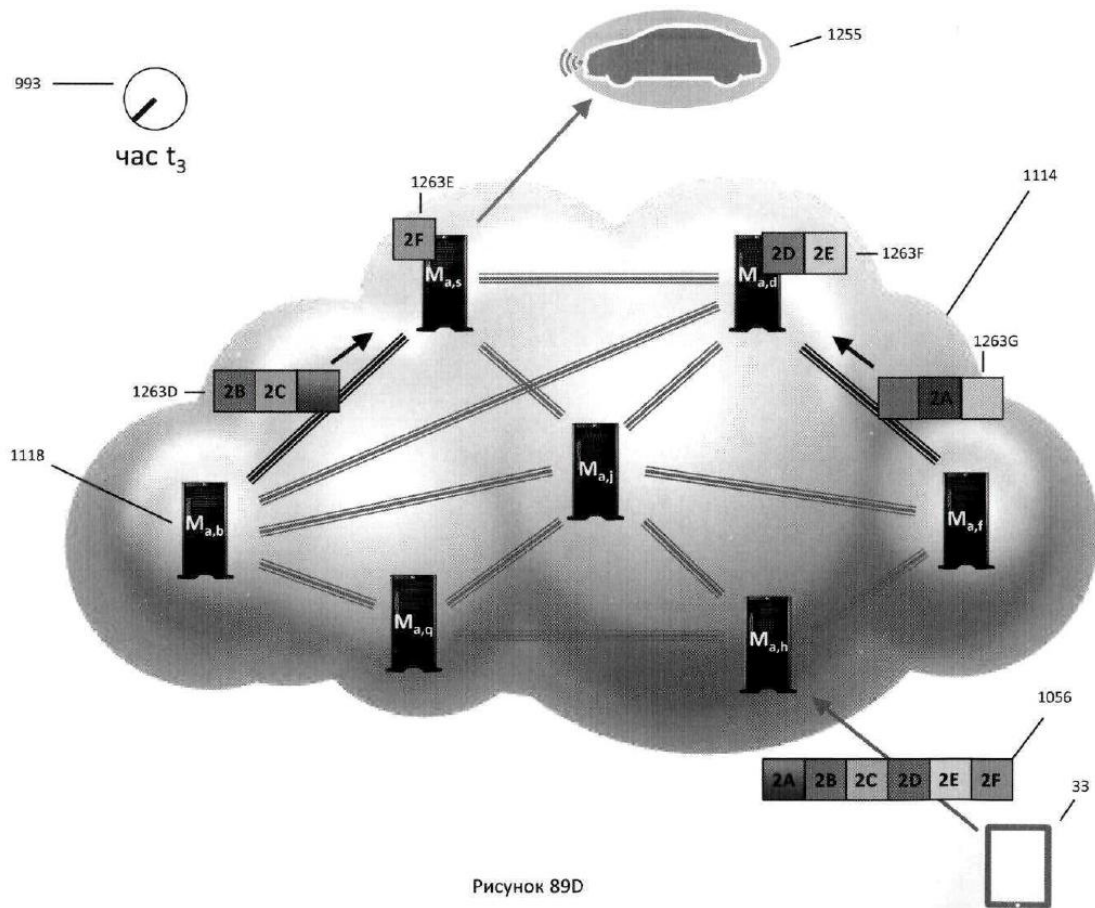


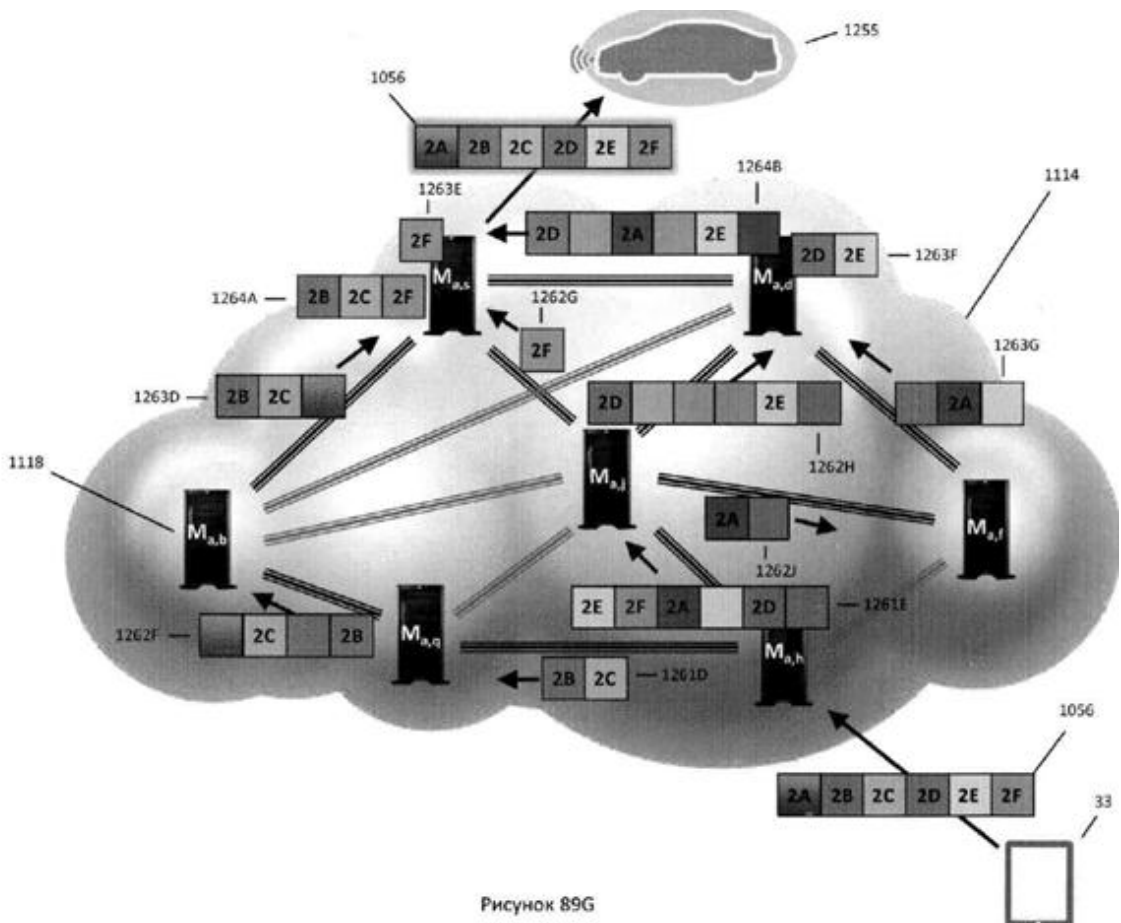
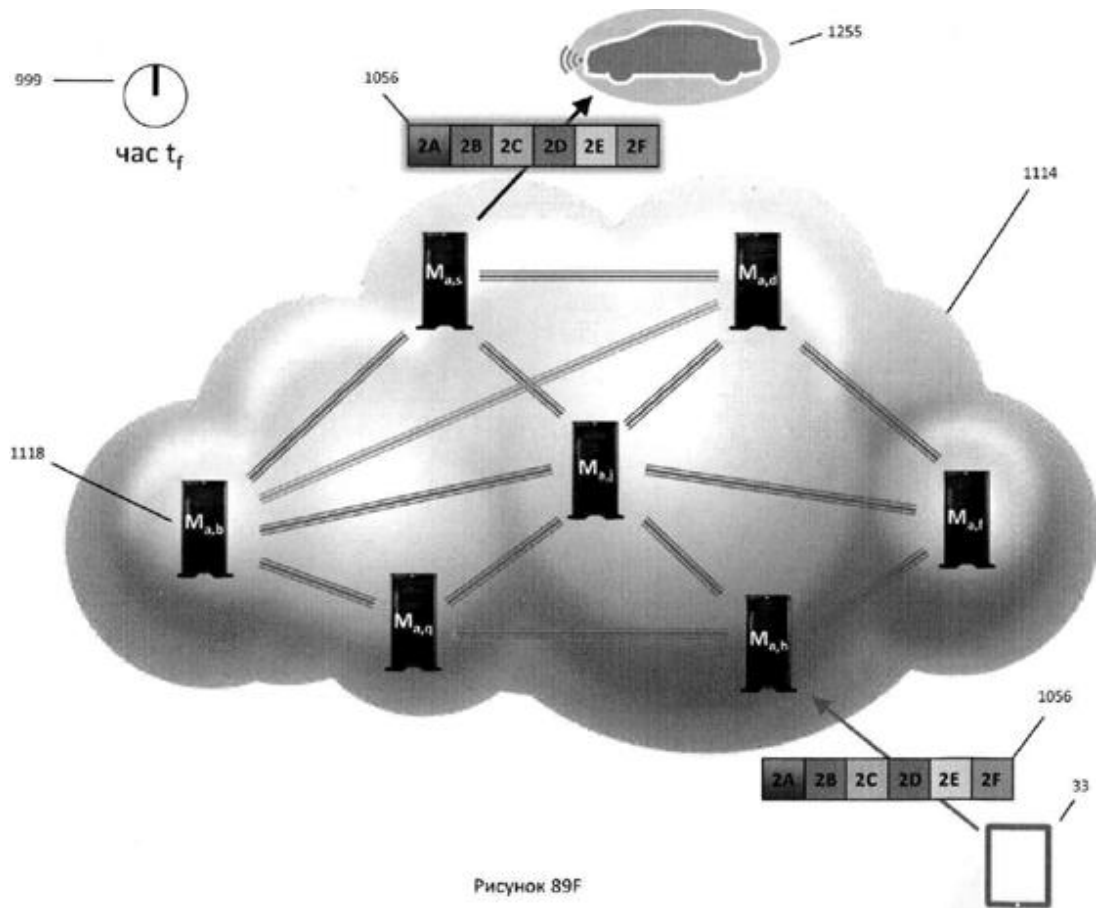
Рисунок 88E

Хмарна маршрутизація пакетів SDNP









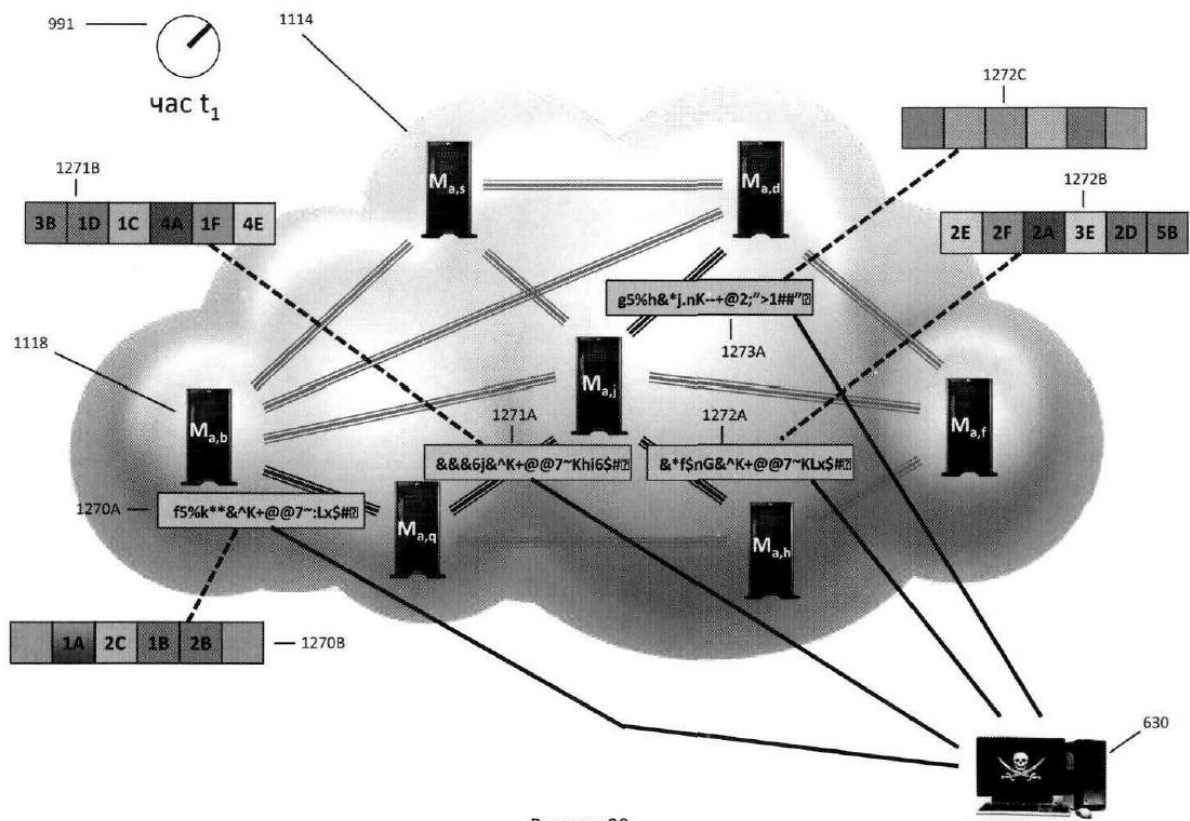


Рисунок 90

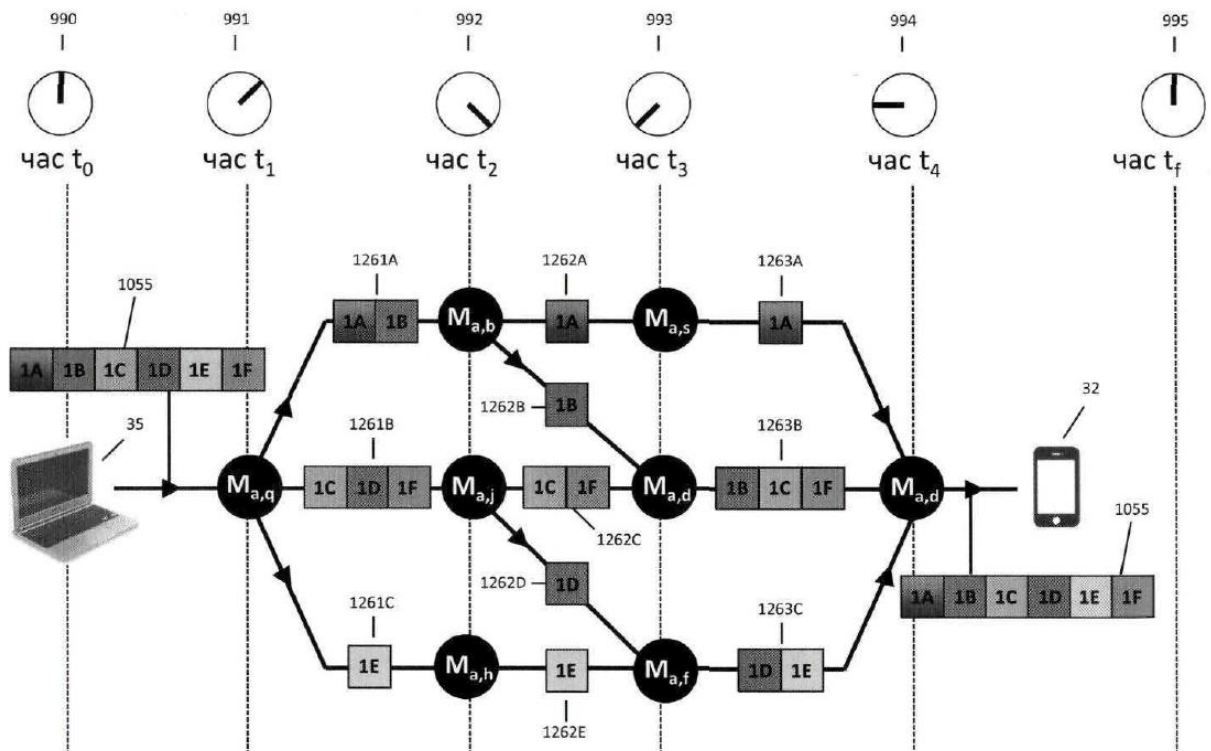


Рисунок 91A




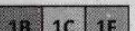




Промінок часу	Вузли-джерела	Вузли, які приймають	Вхідний пакет	Час останнього шифрування	Час останнього скремблювання	Час останнього змішування і розділення	Вихідний пакет
t_3	$M_{a,b}$	$M_{a,s}$	 1A ↗ 1262A	t_2	t_1 or t_2	T_2	 1A ↗ 1263A
t_3	$M_{a,b}$	$M_{a,d}$	 1B ↗ 1262B	t_2	t_1 or t_2	t_2	 1B 1C 1F ↗ 1263B
t_3	$M_{a,j}$	$M_{a,d}$	 1C 1F ↗ 1262C	t_2	t_1 or t_2	t_2	
t_3	$M_{a,j}$	$M_{a,f}$	 1D ↗ 1262D	t_2	t_1 or t_2	t_2	 1D 1E ↗ 1263C
t_3	$M_{a,h}$	$M_{a,f}$	 1E ↗ 1262E	t_2	t_1 or t_2	t_2	

Рисунок 91В

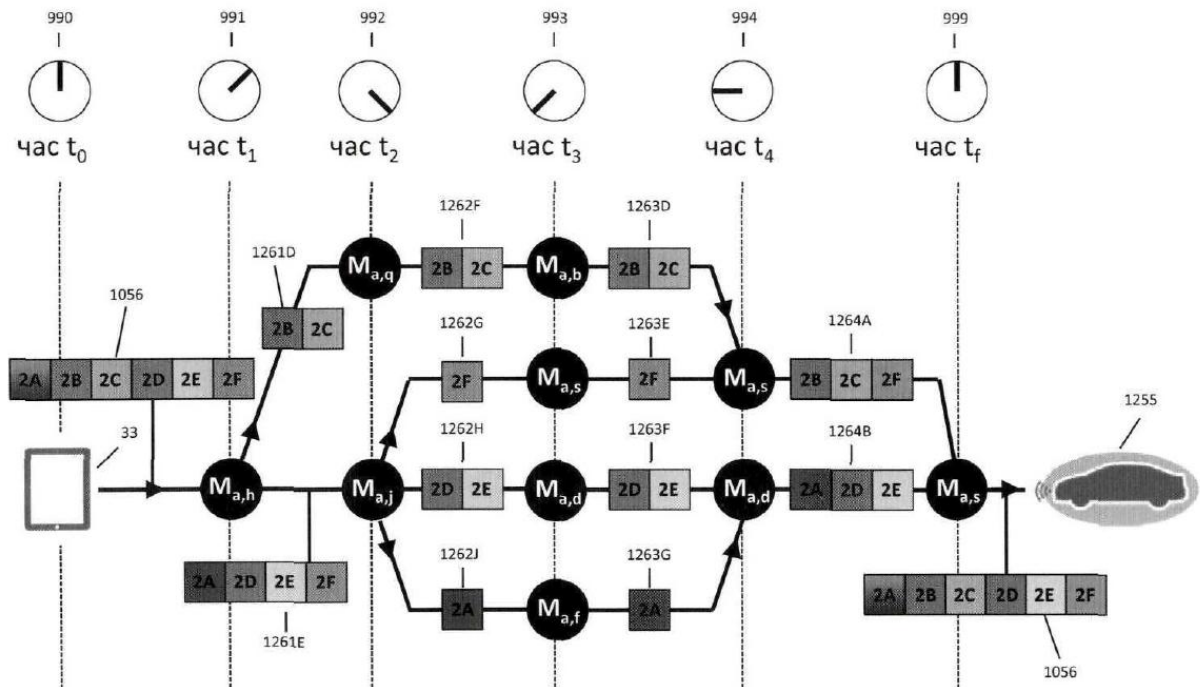


Рисунок 91С

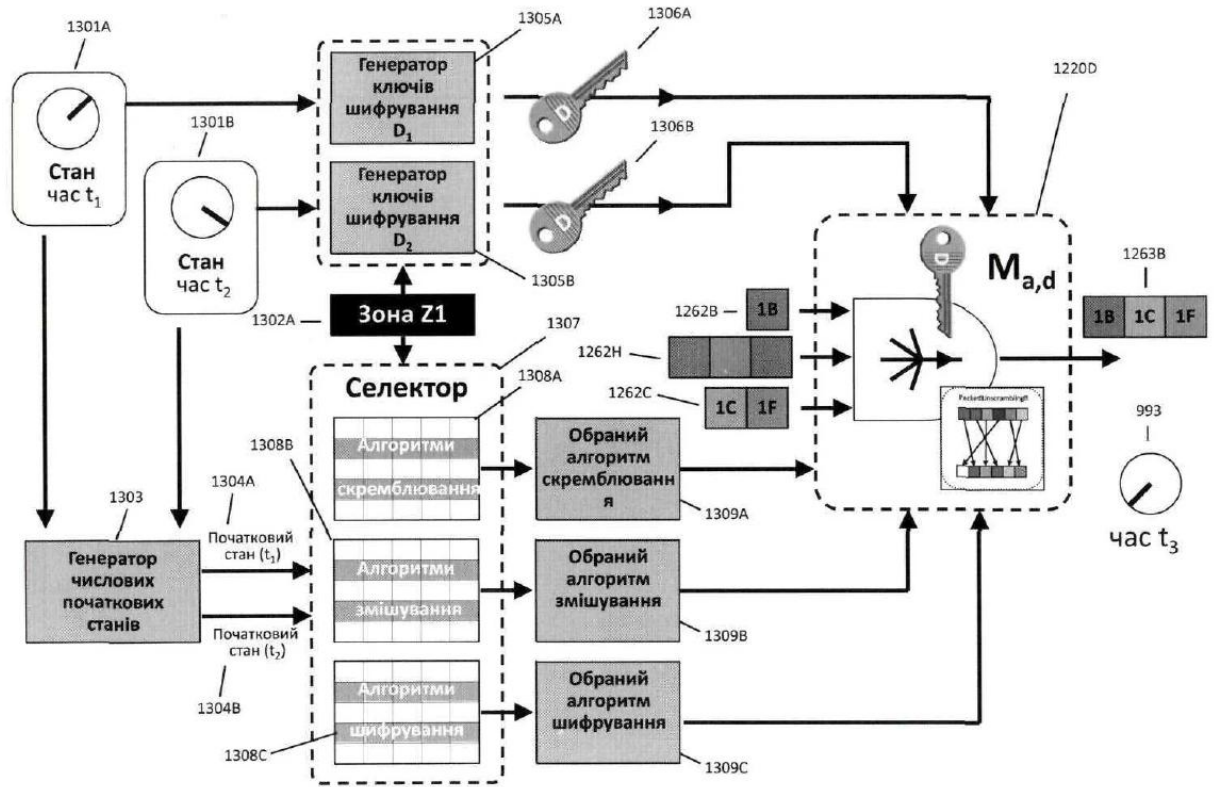


Рисунок 92А

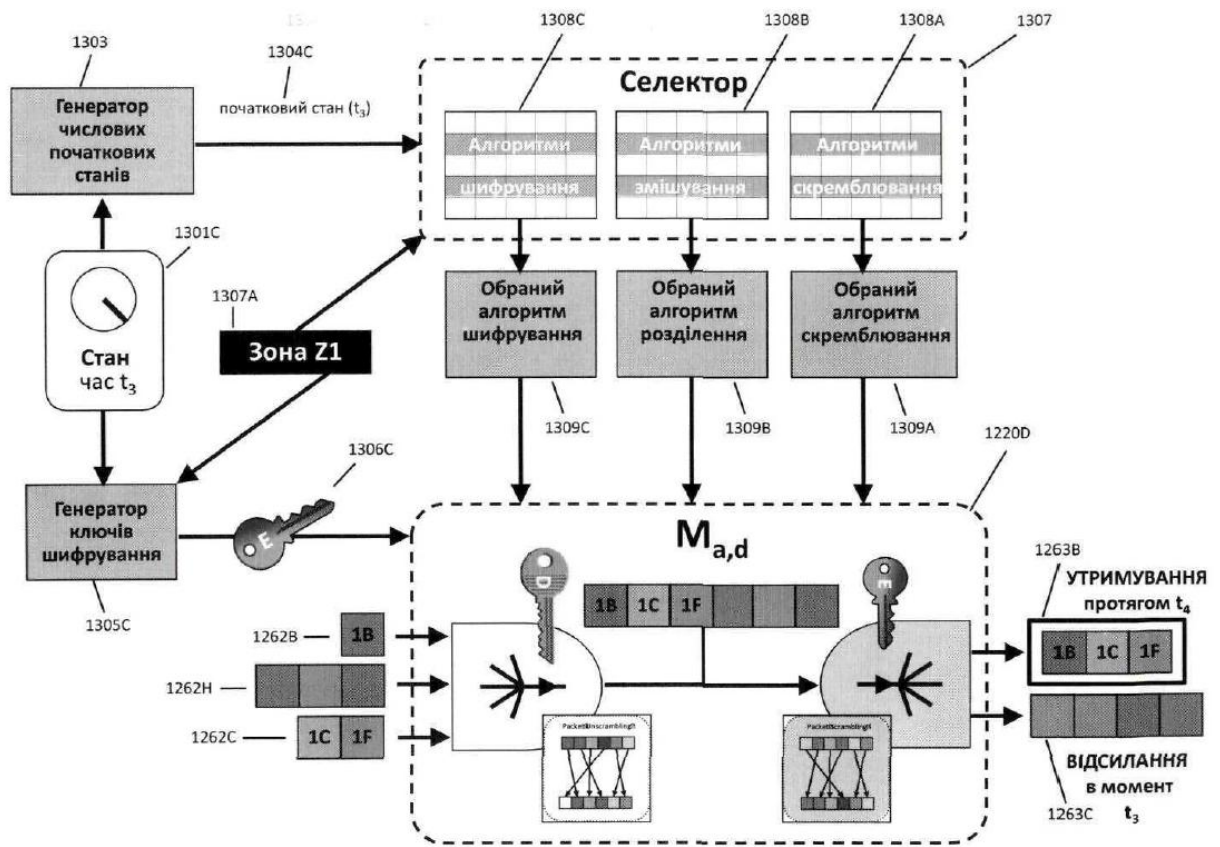


Рисунок 92В

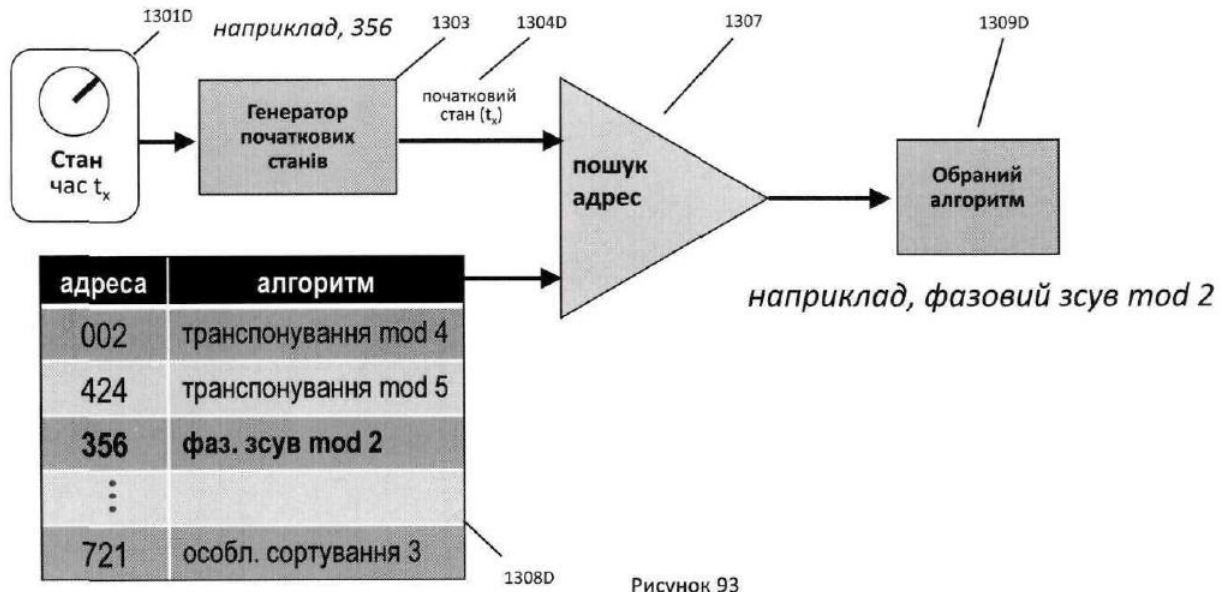
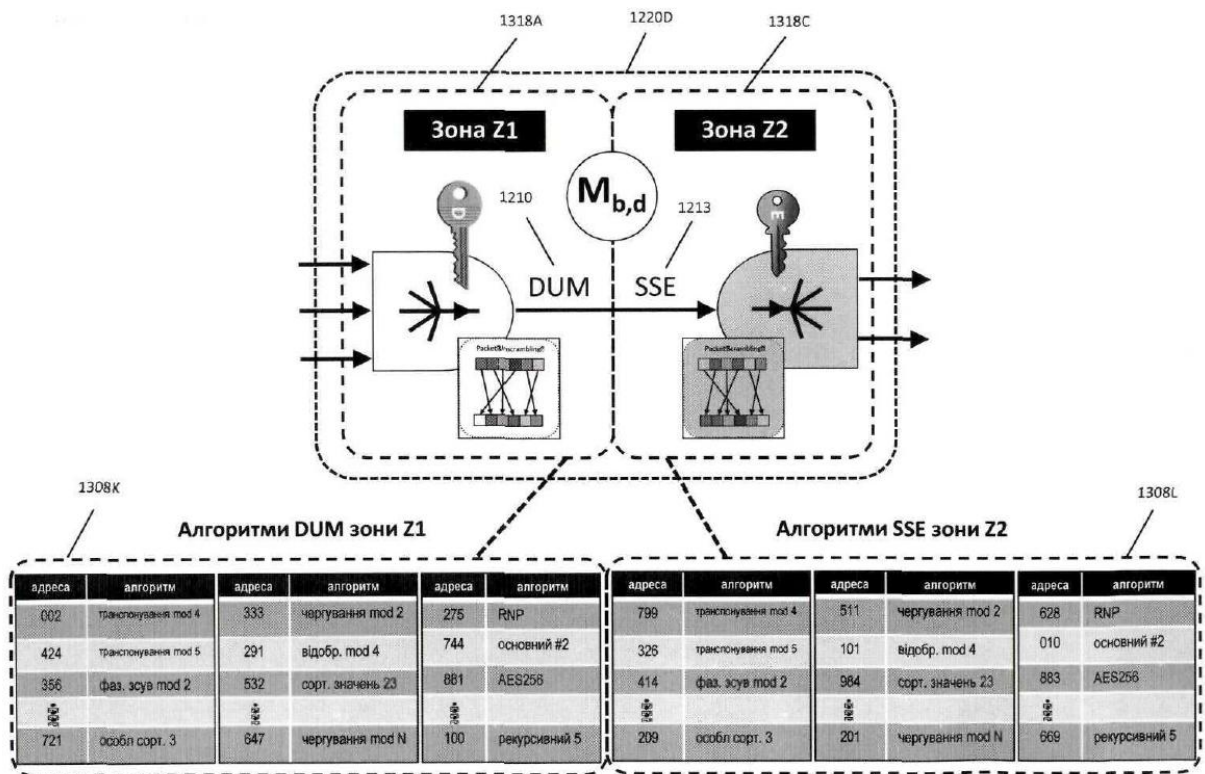
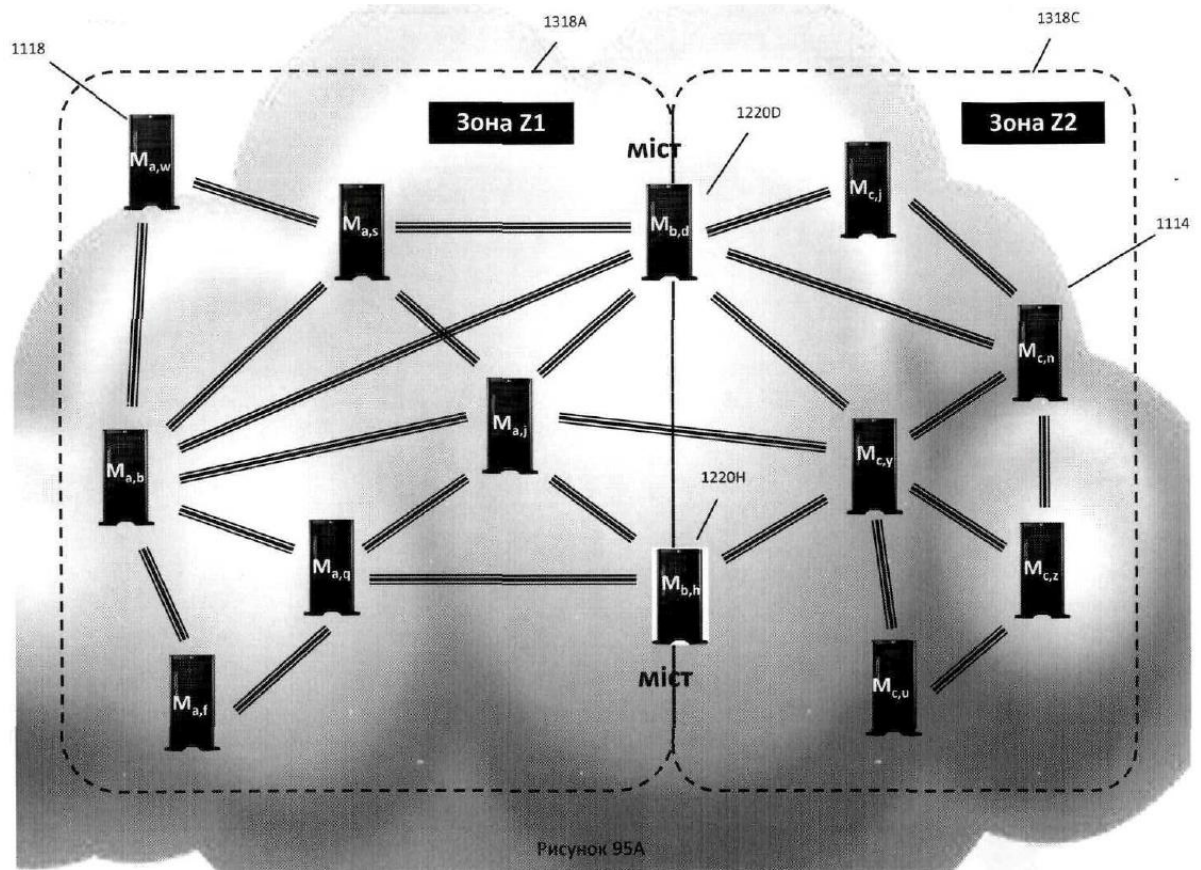


Рисунок 93



Рисунок 94



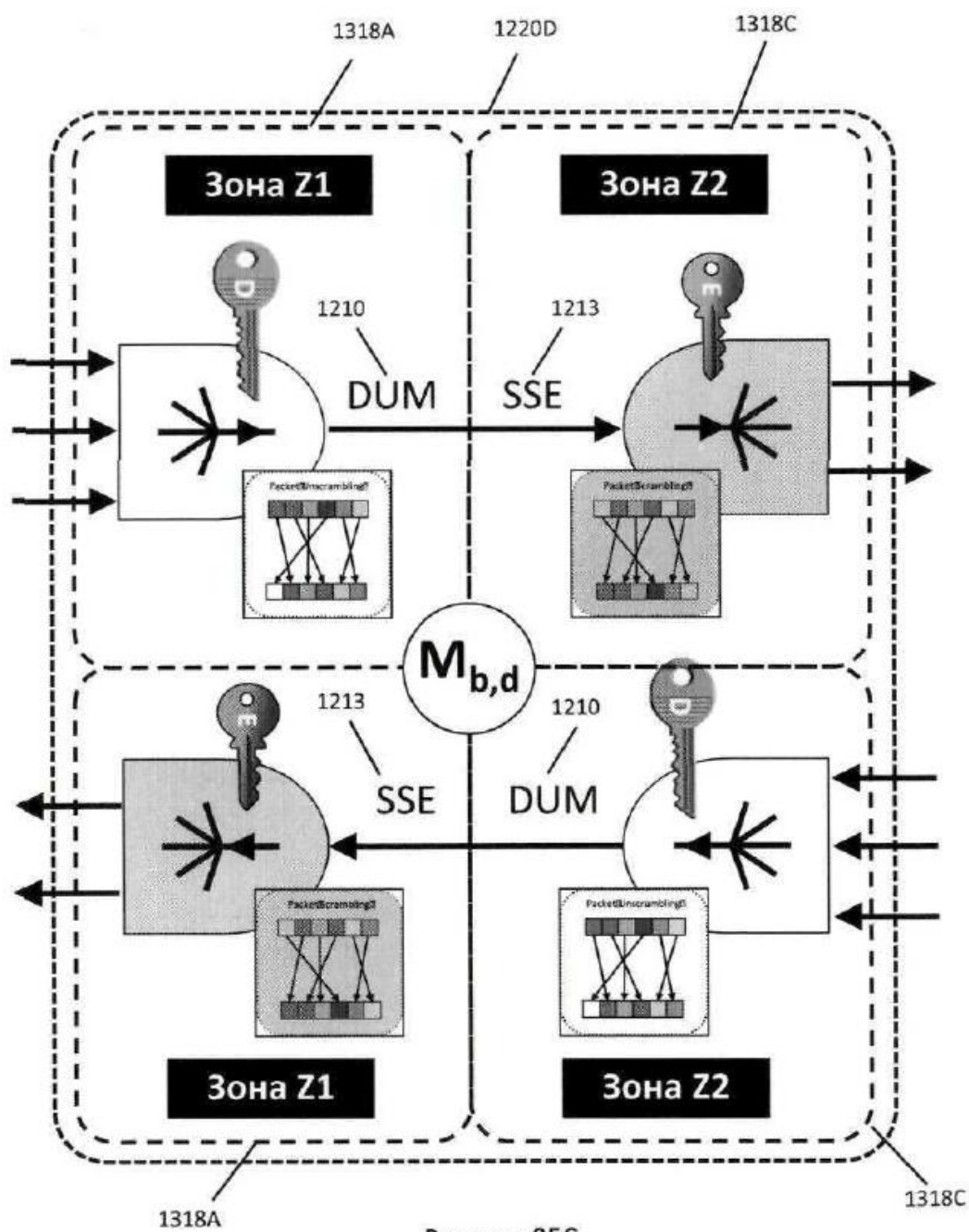


Рисунок 95C

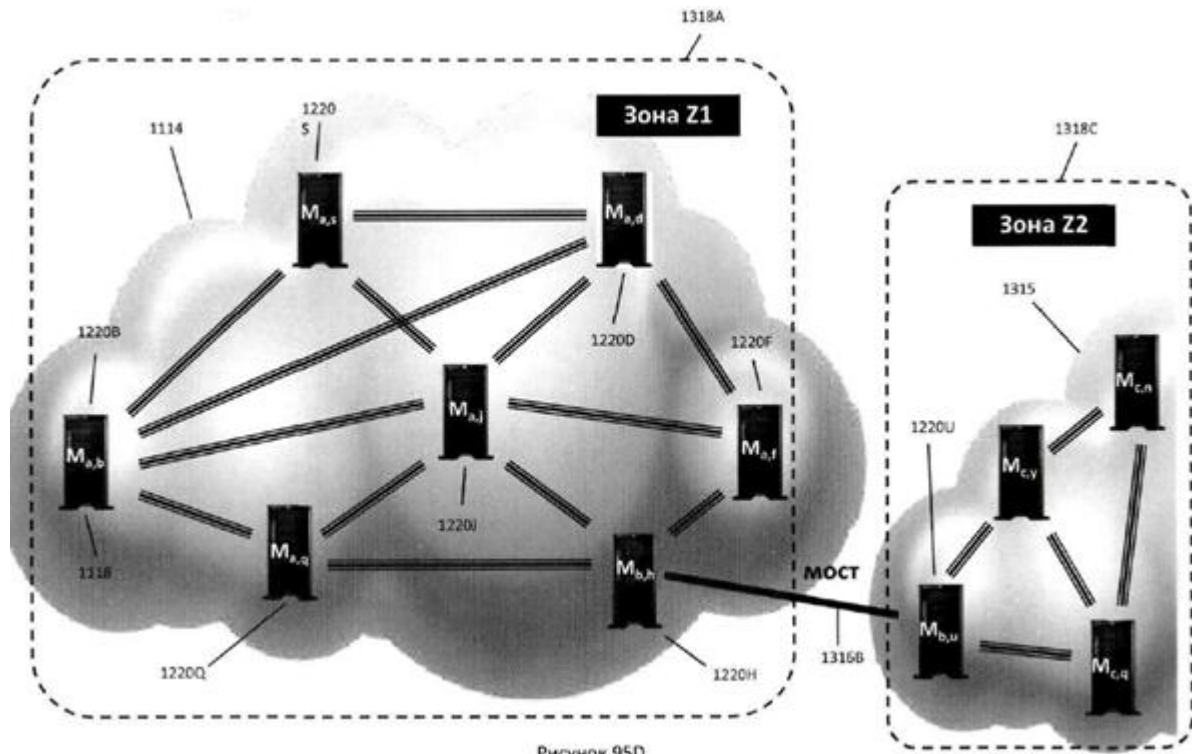


Рисунок 95D

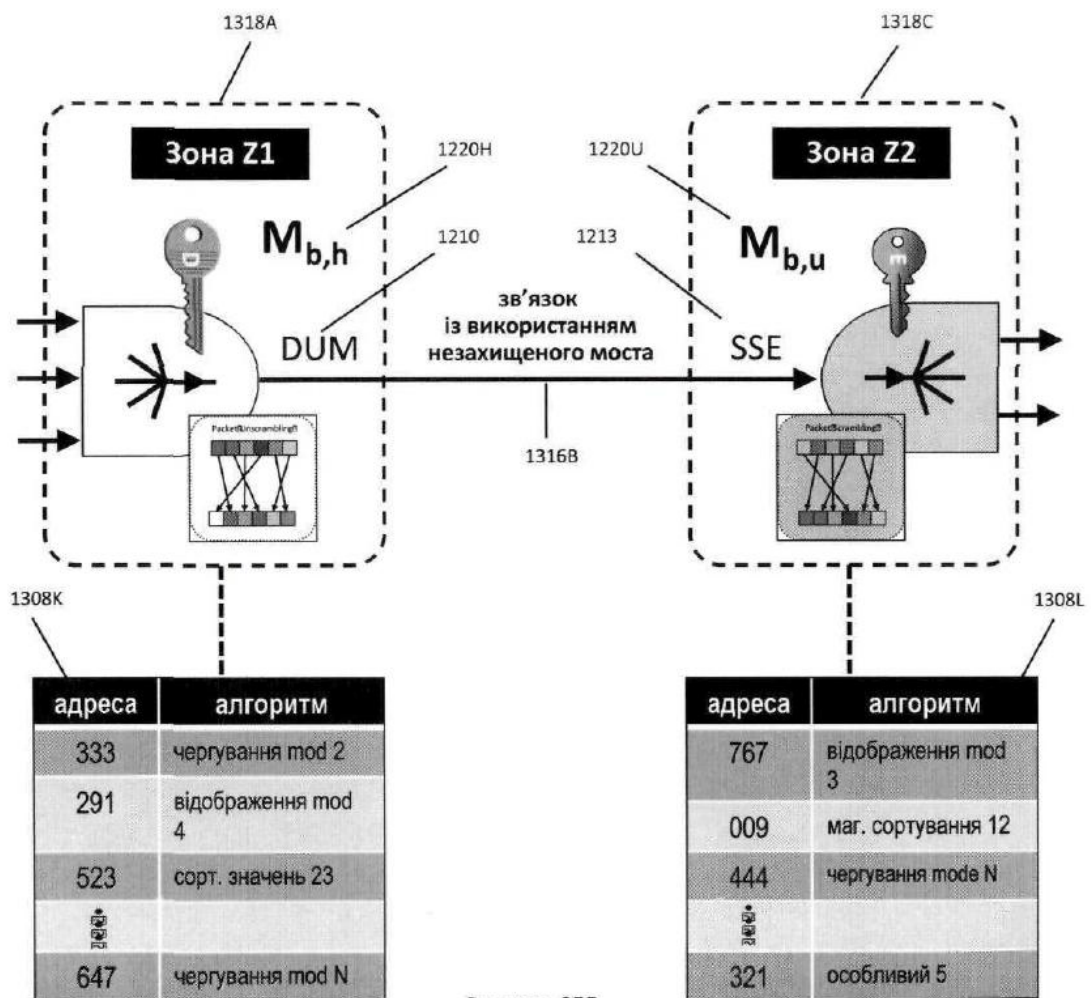


Рисунок 95E

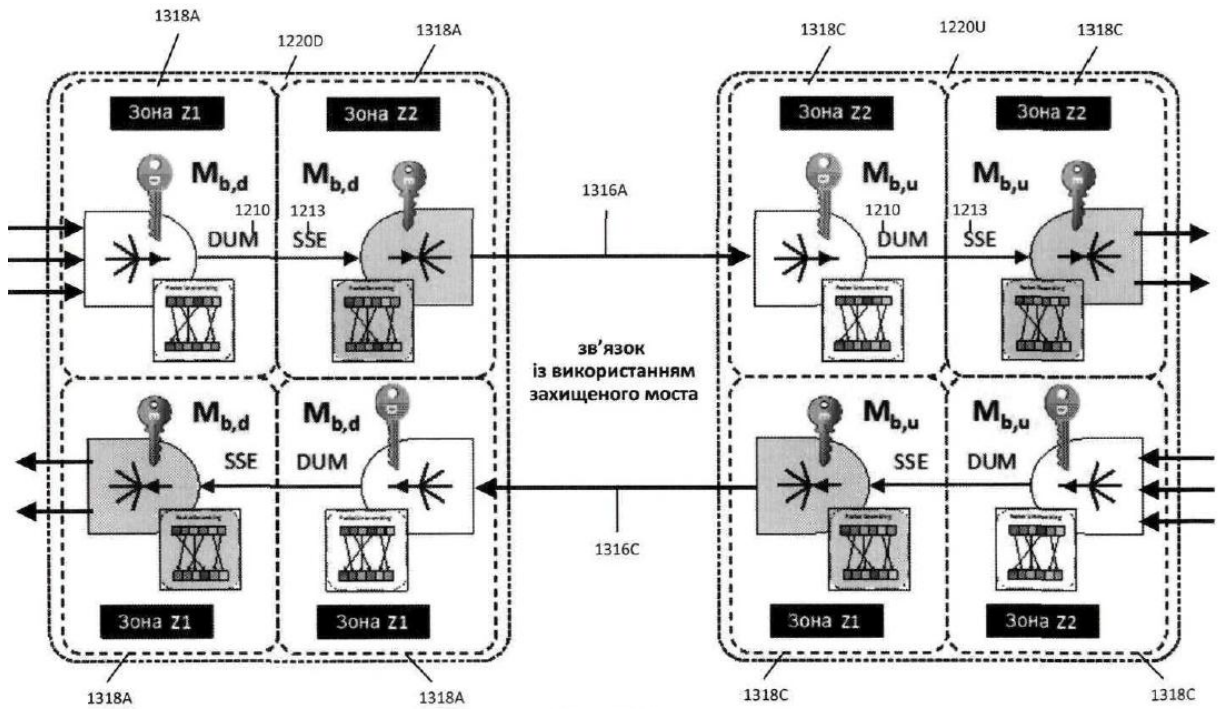


Рисунок 95F

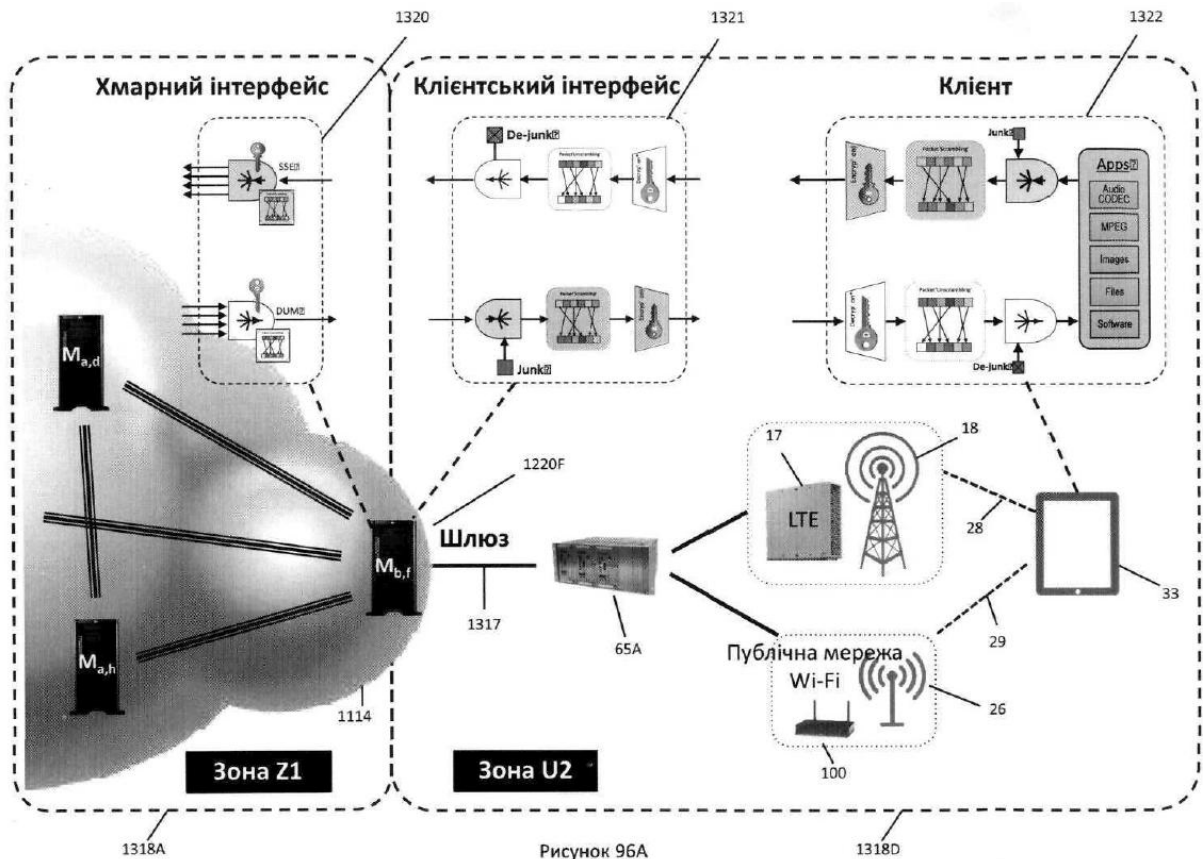


Рисунок 96A

Хмарний інтерфейс

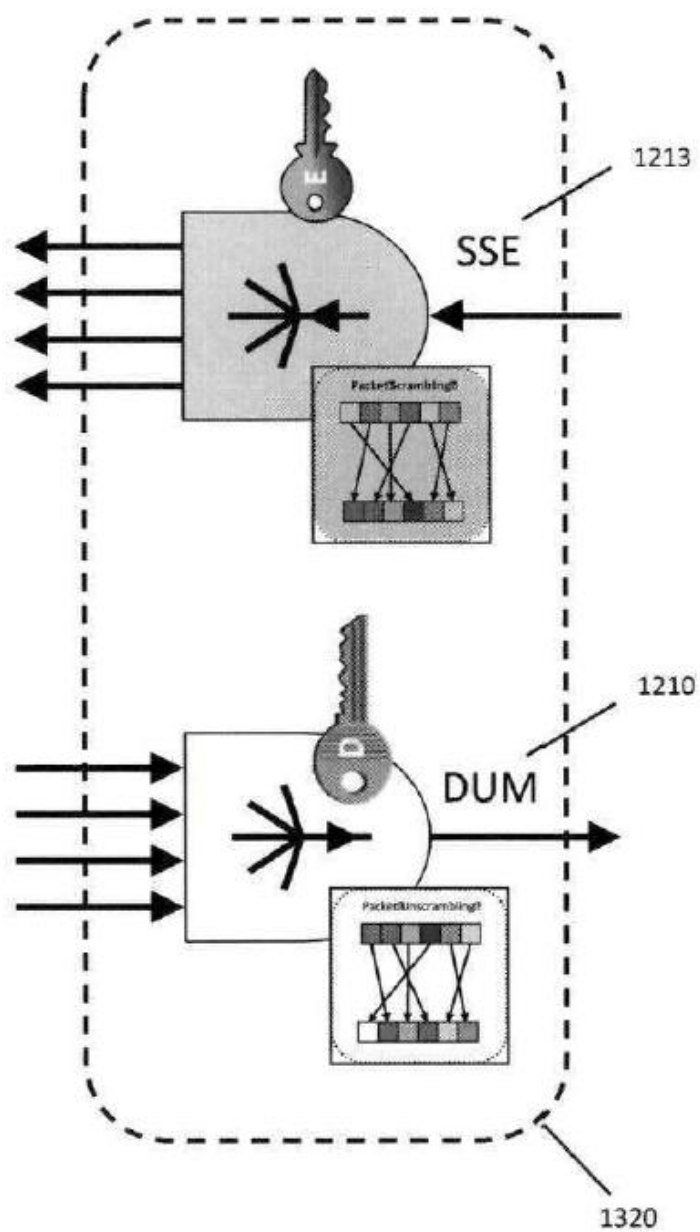


Рисунок 96В

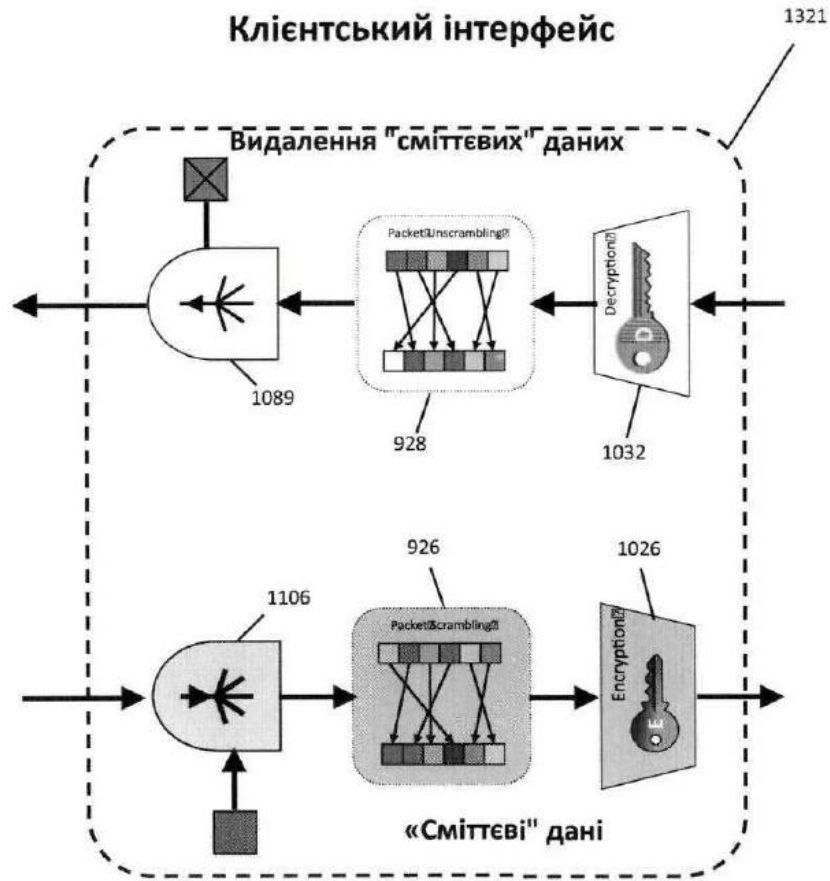


Рисунок 96C

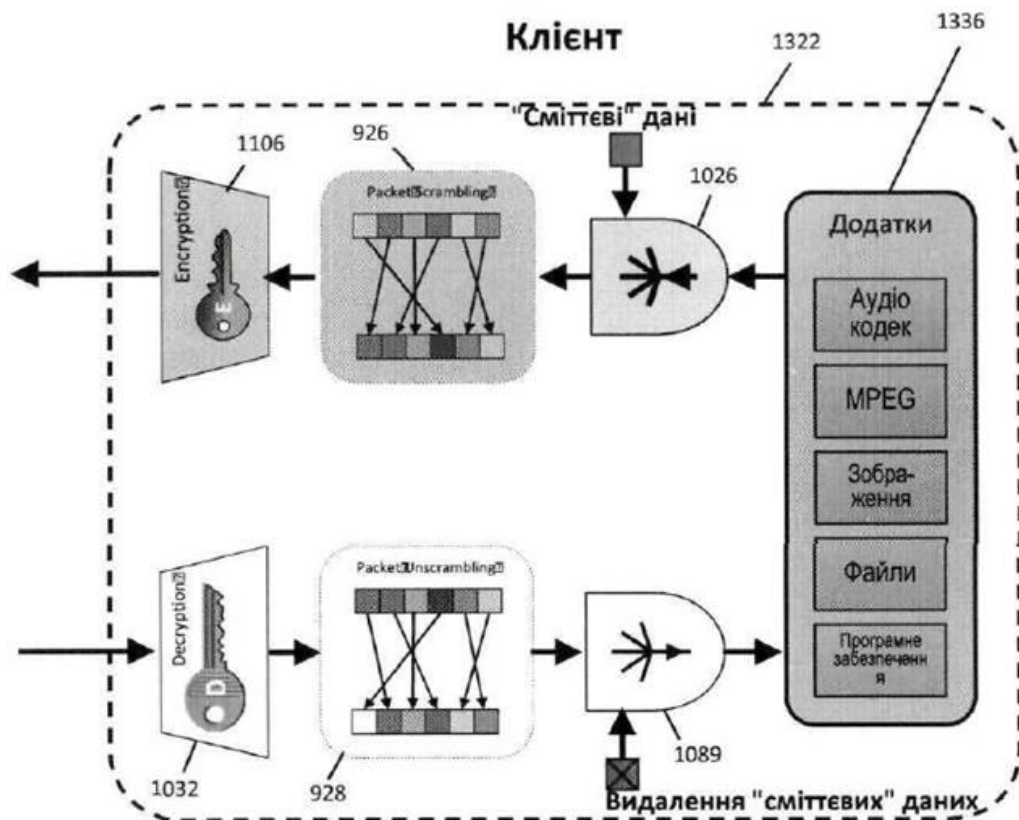


Рисунок 96D

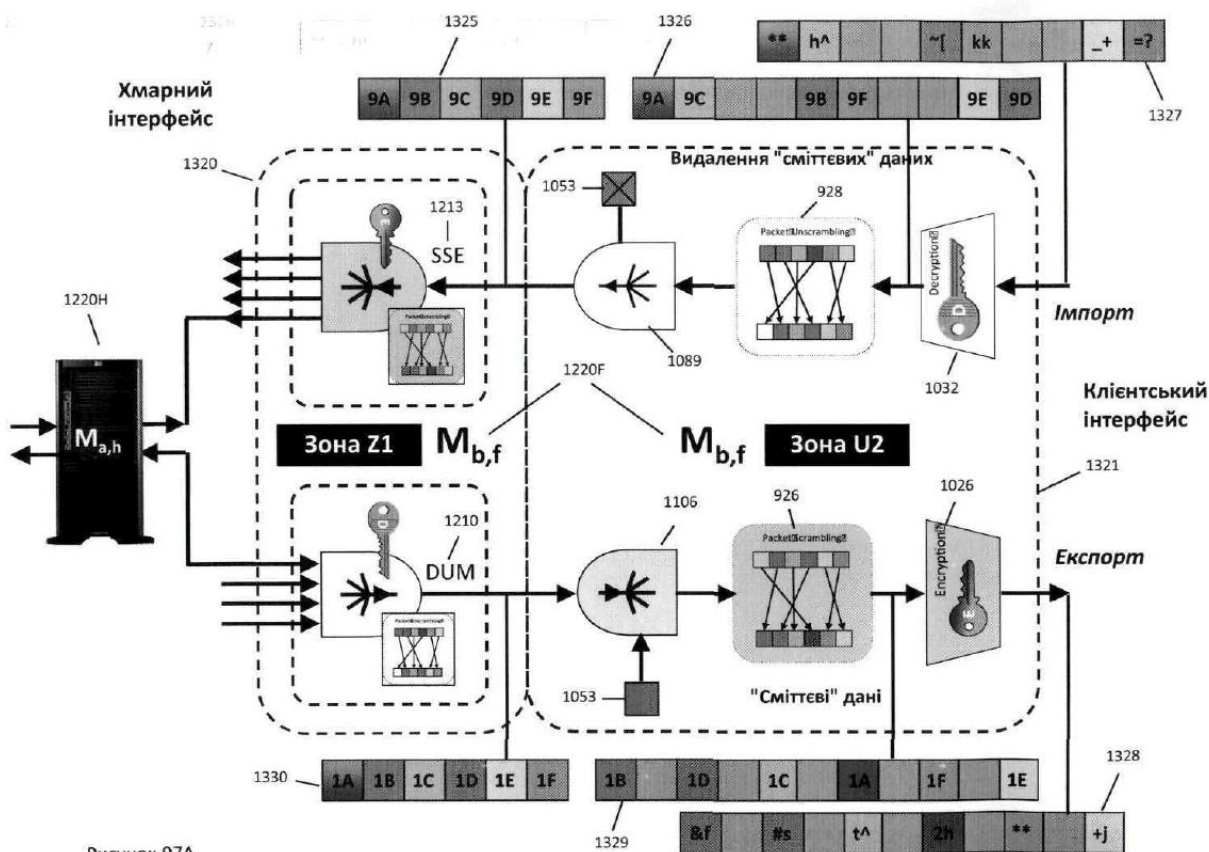


Рисунок 97А

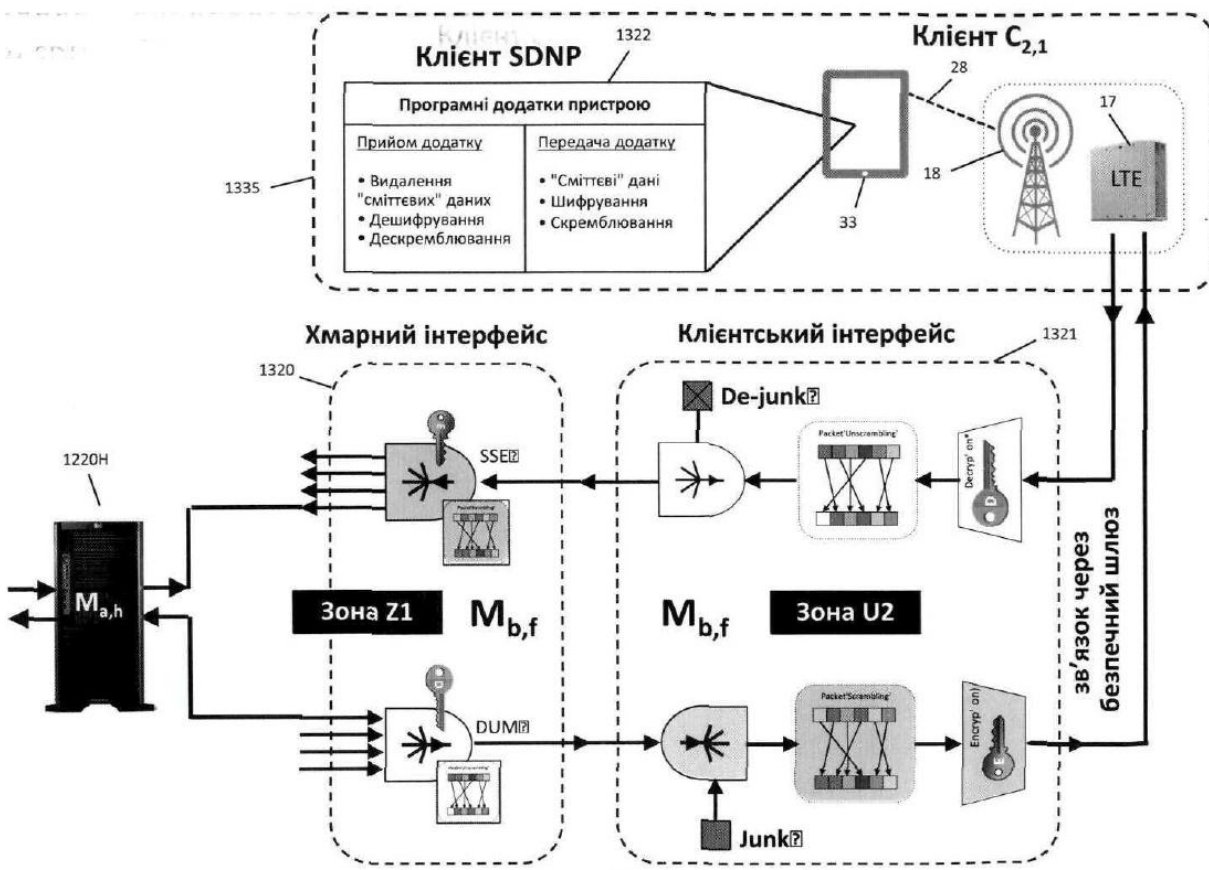


Рисунок 97В

Клієнтський інтерфейс SDNP

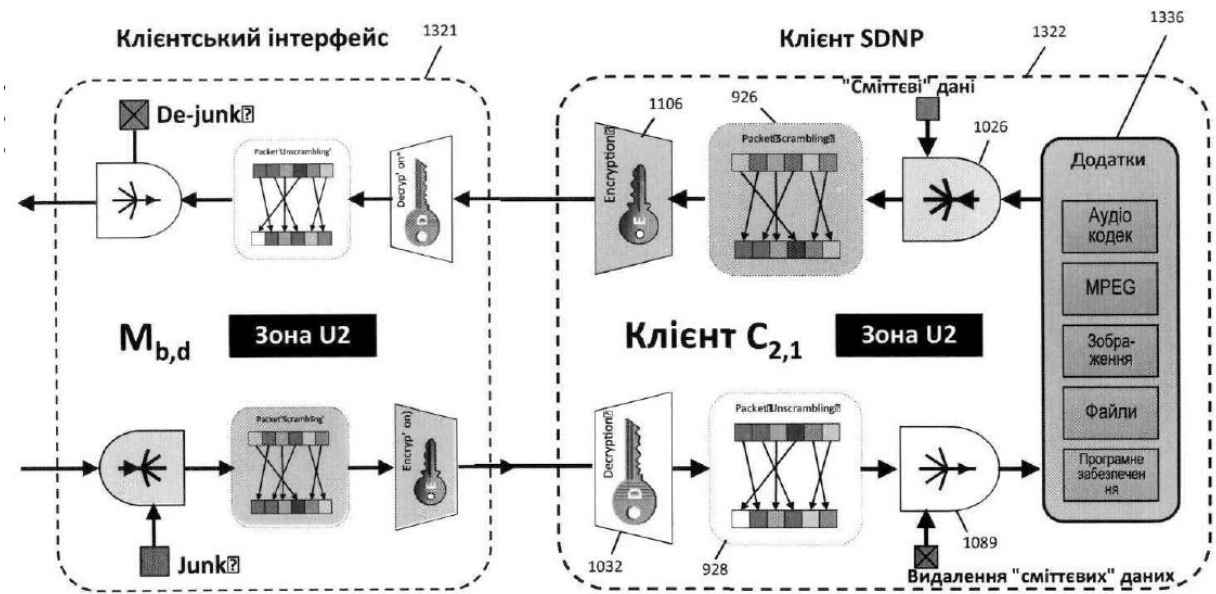


Рисунок 98

Розподілення ключів по зонах

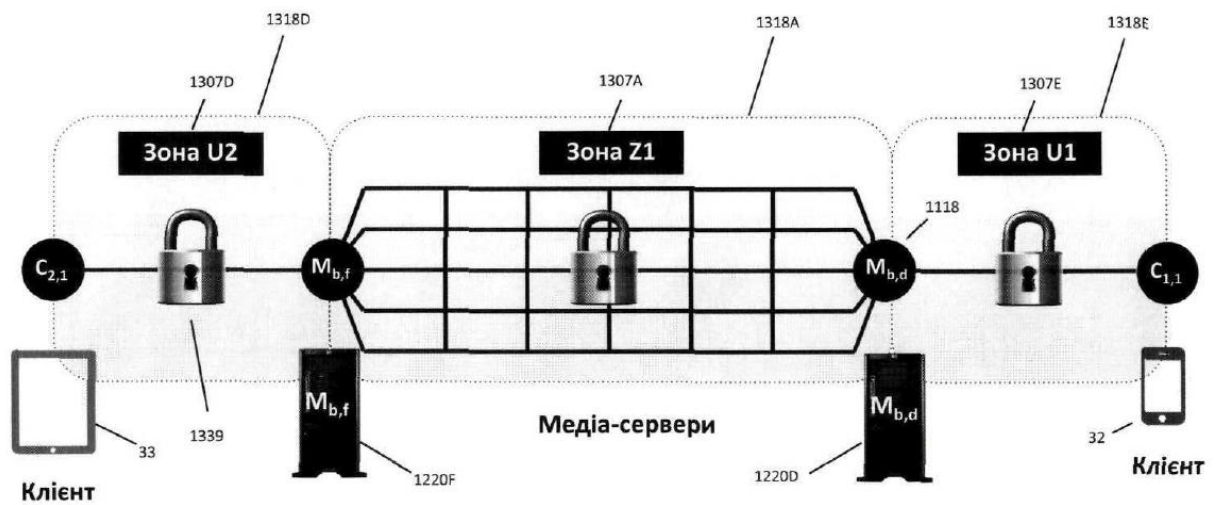


Рисунок 99A

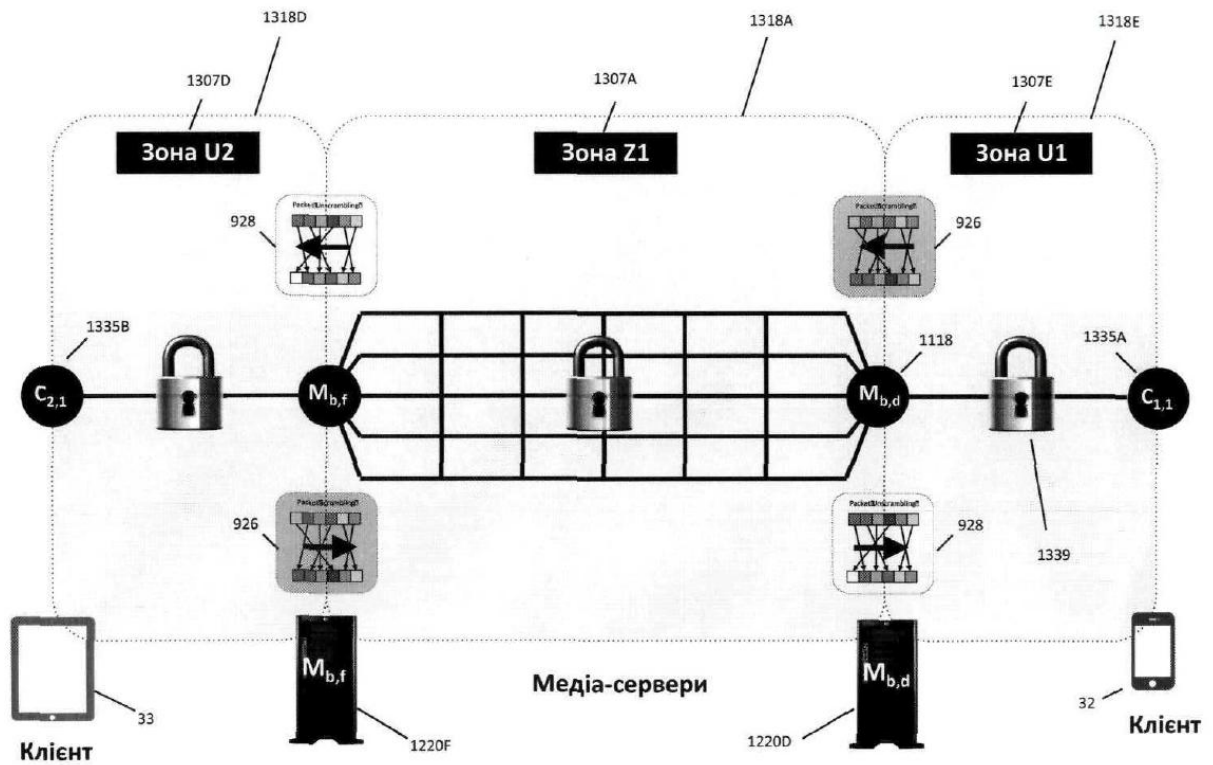


Рисунок 99В

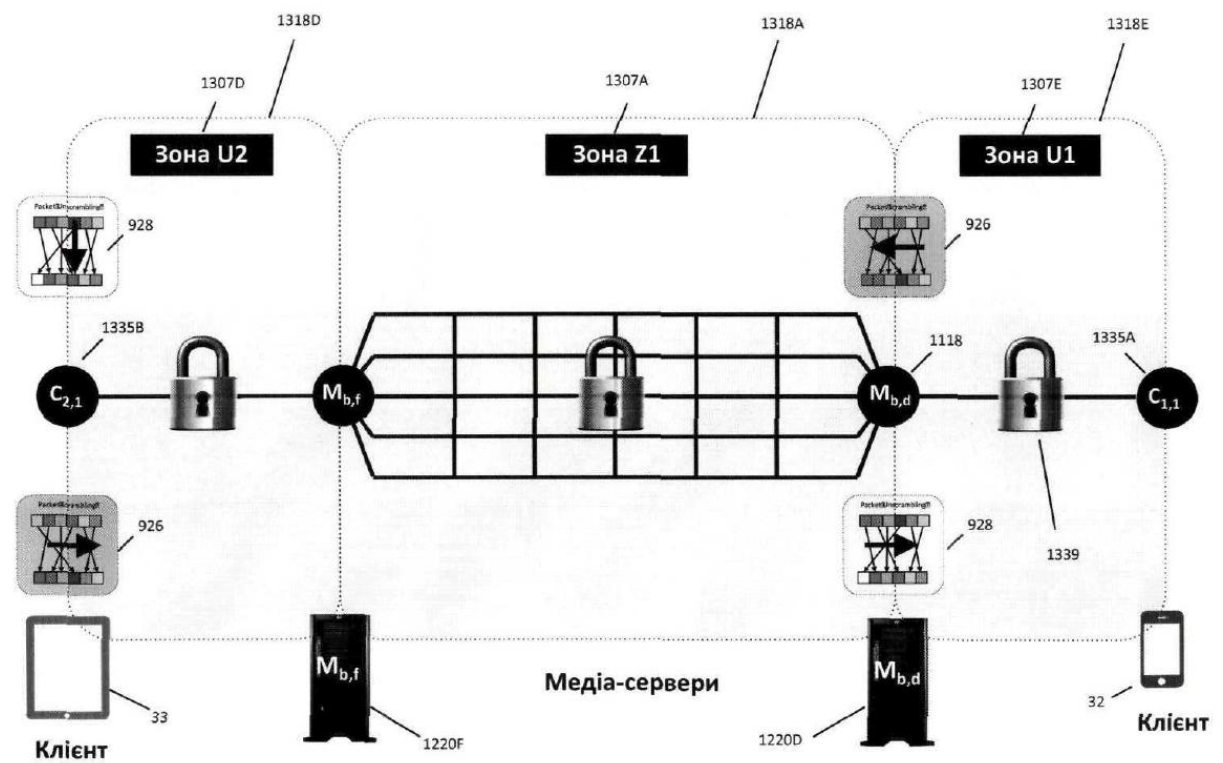


Рисунок 99С

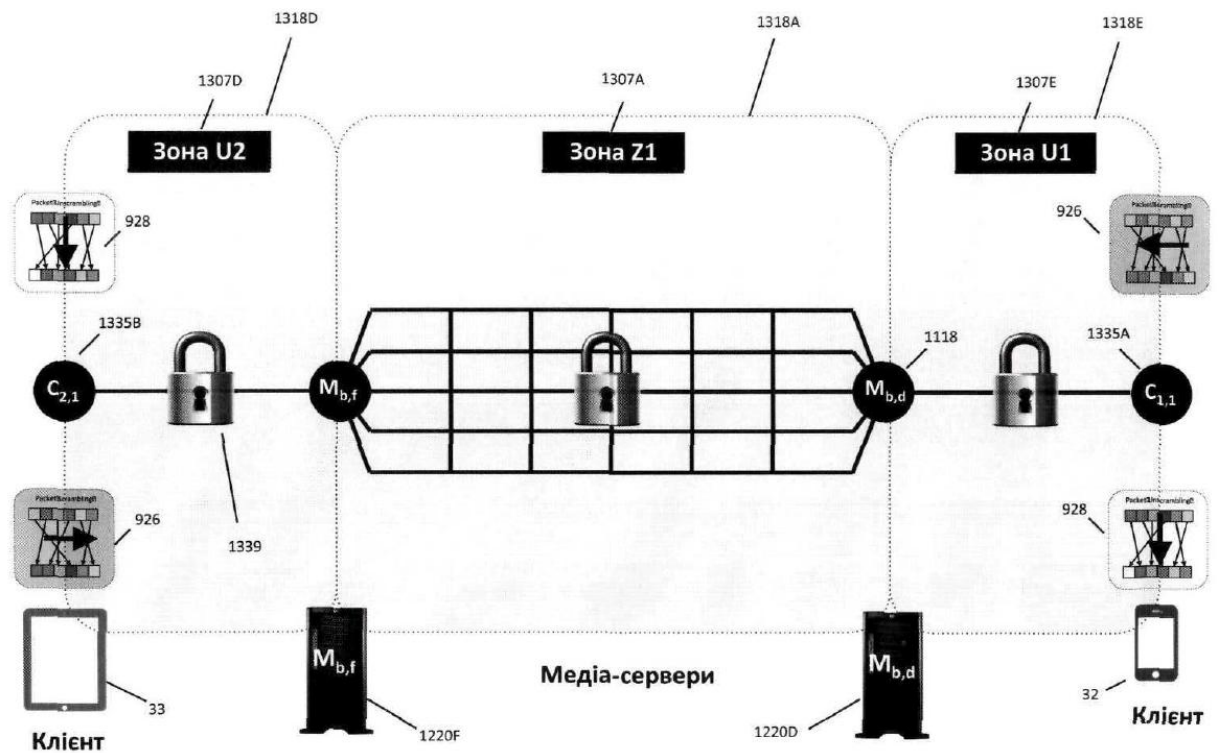


Рисунок 99D

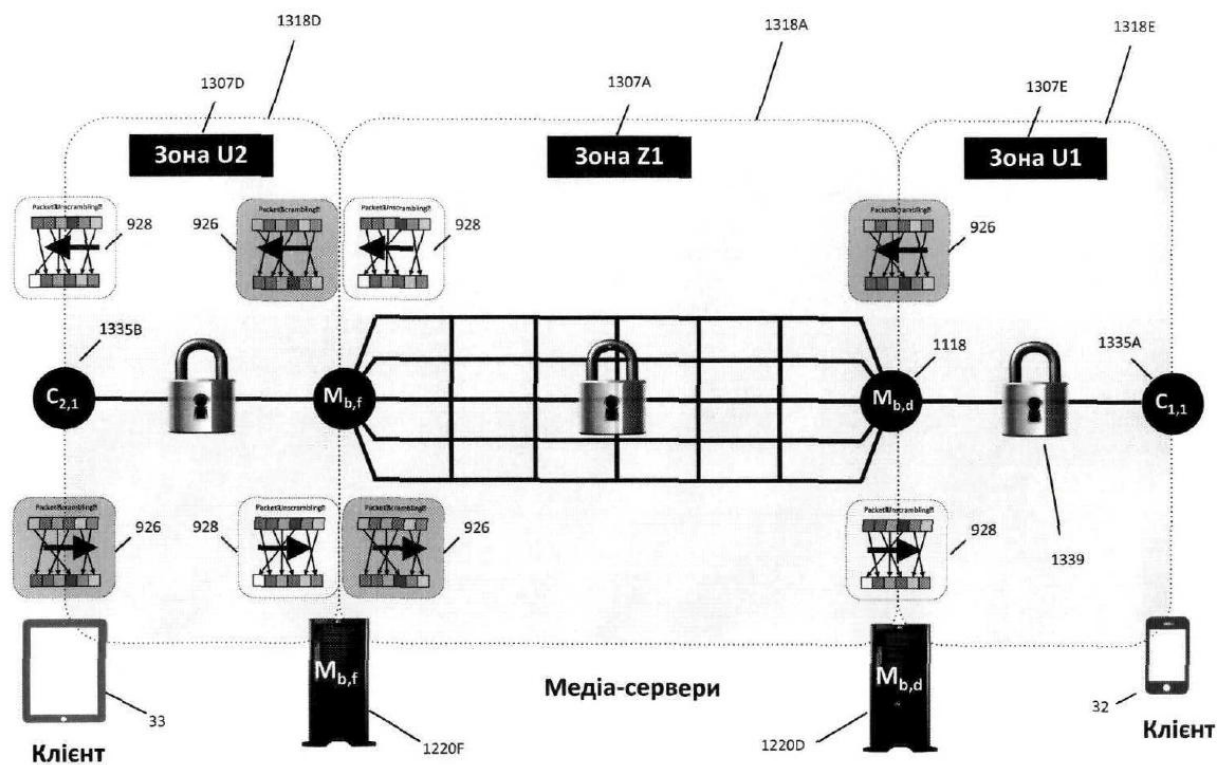


Рисунок 99E

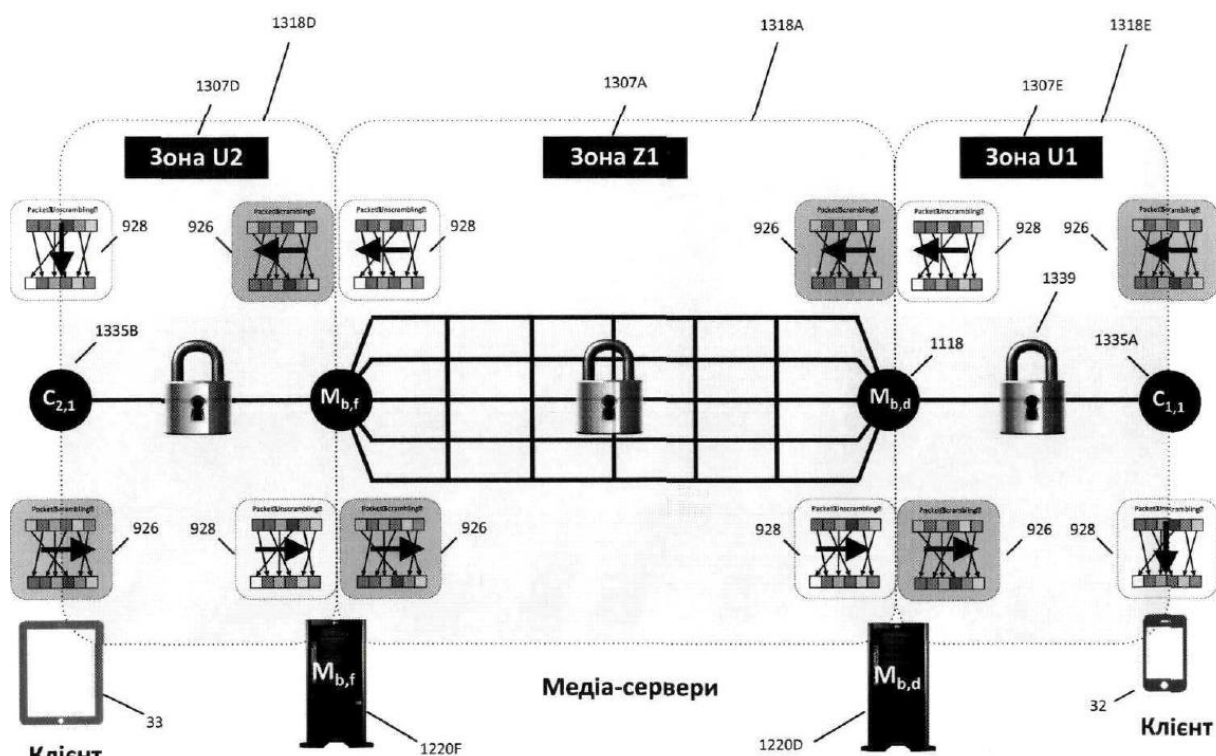


Рисунок 99F

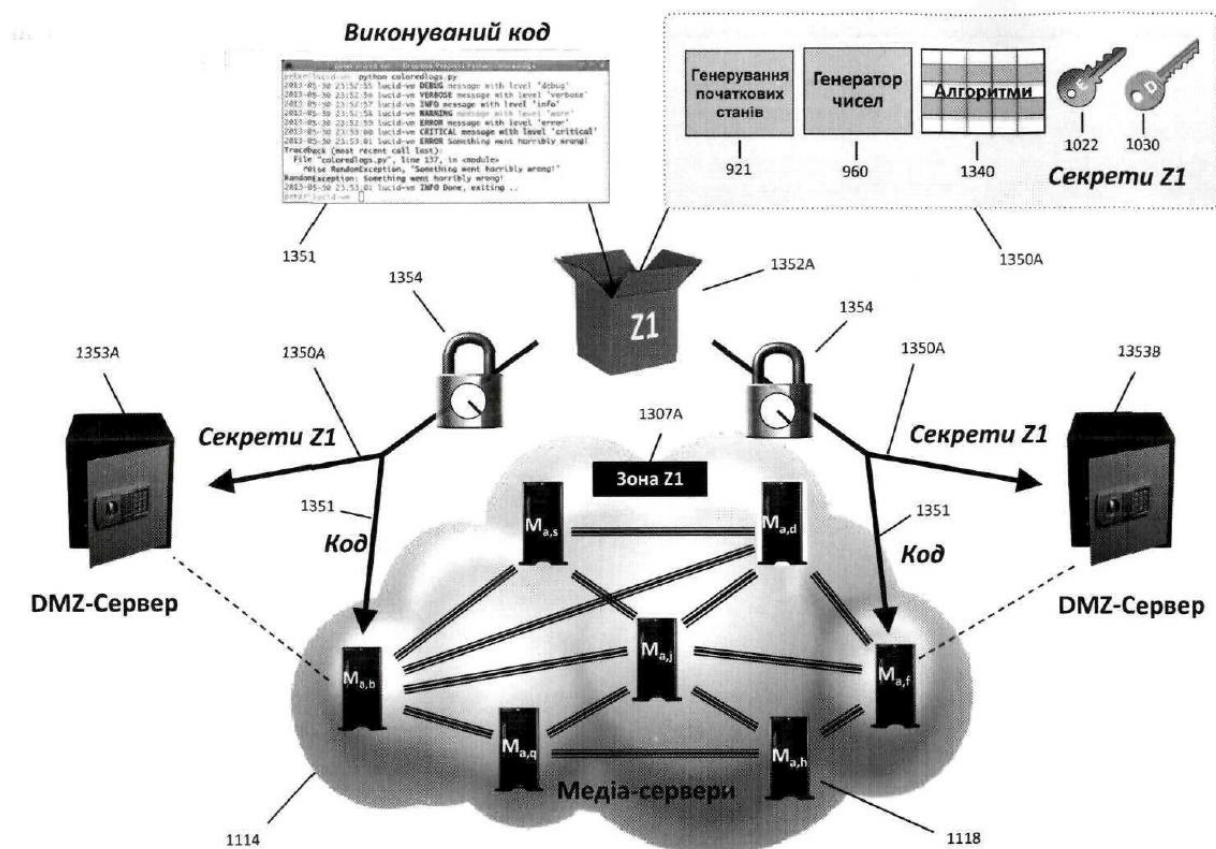


Рисунок 100А

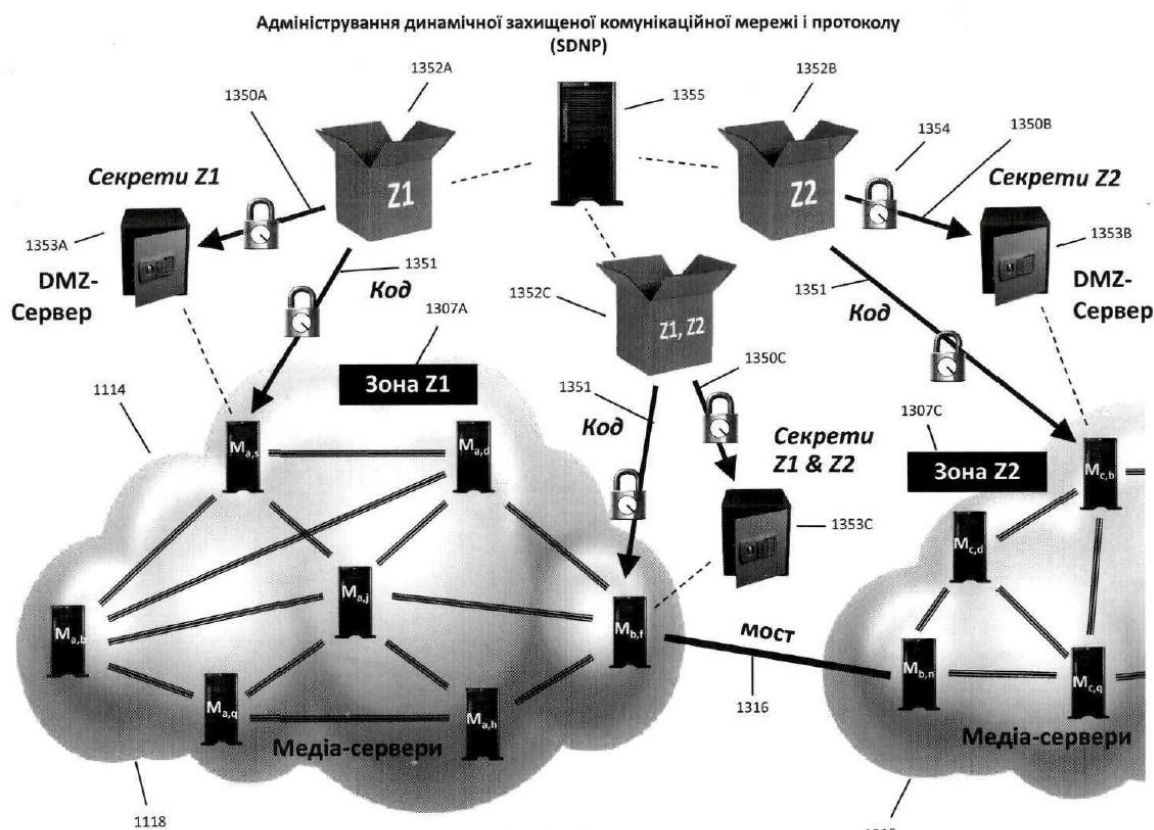


Рисунок 100В

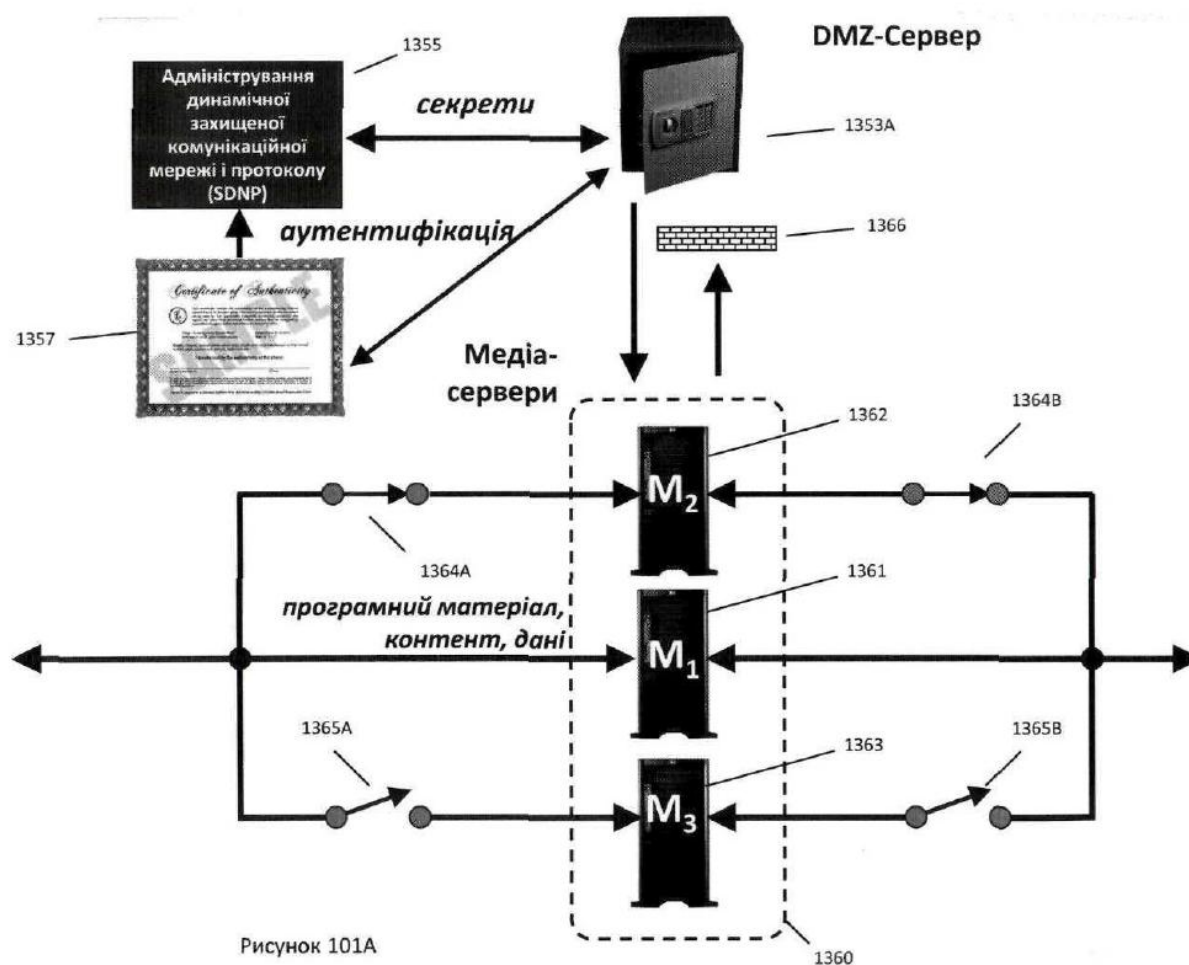
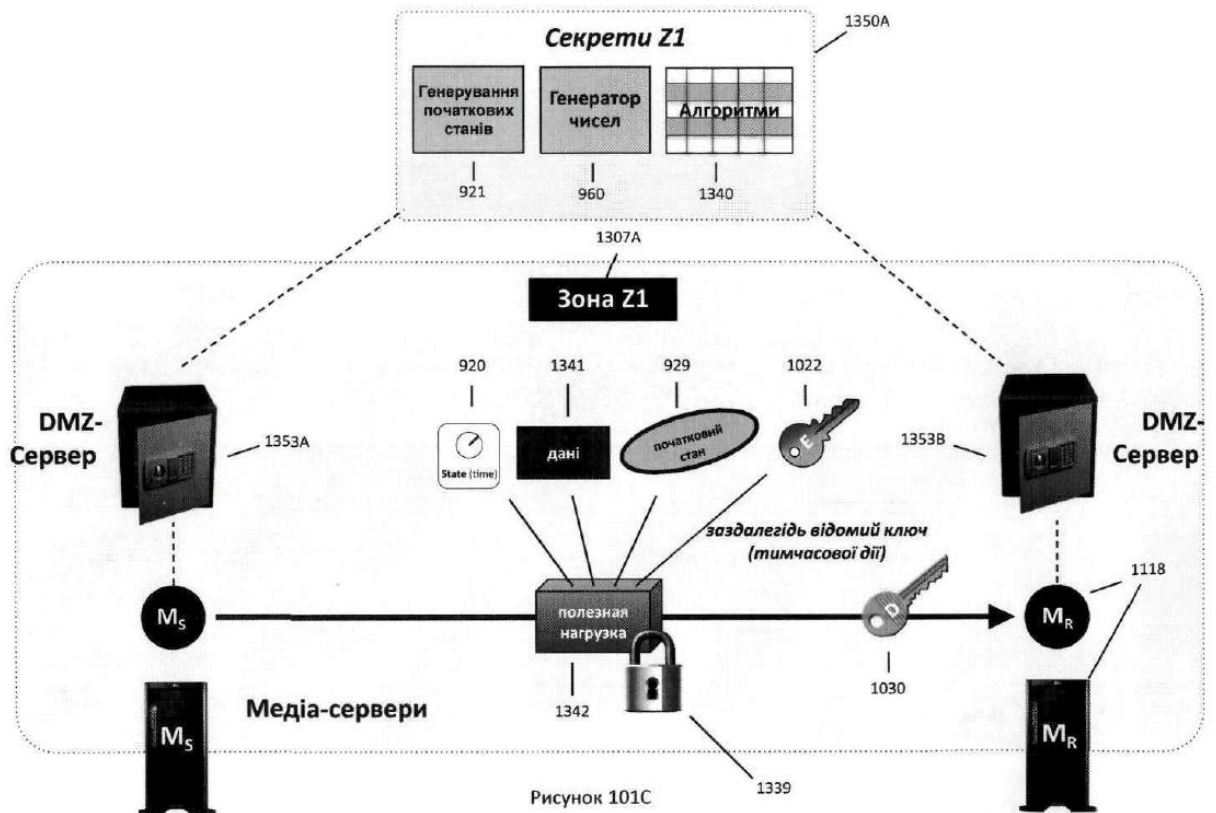
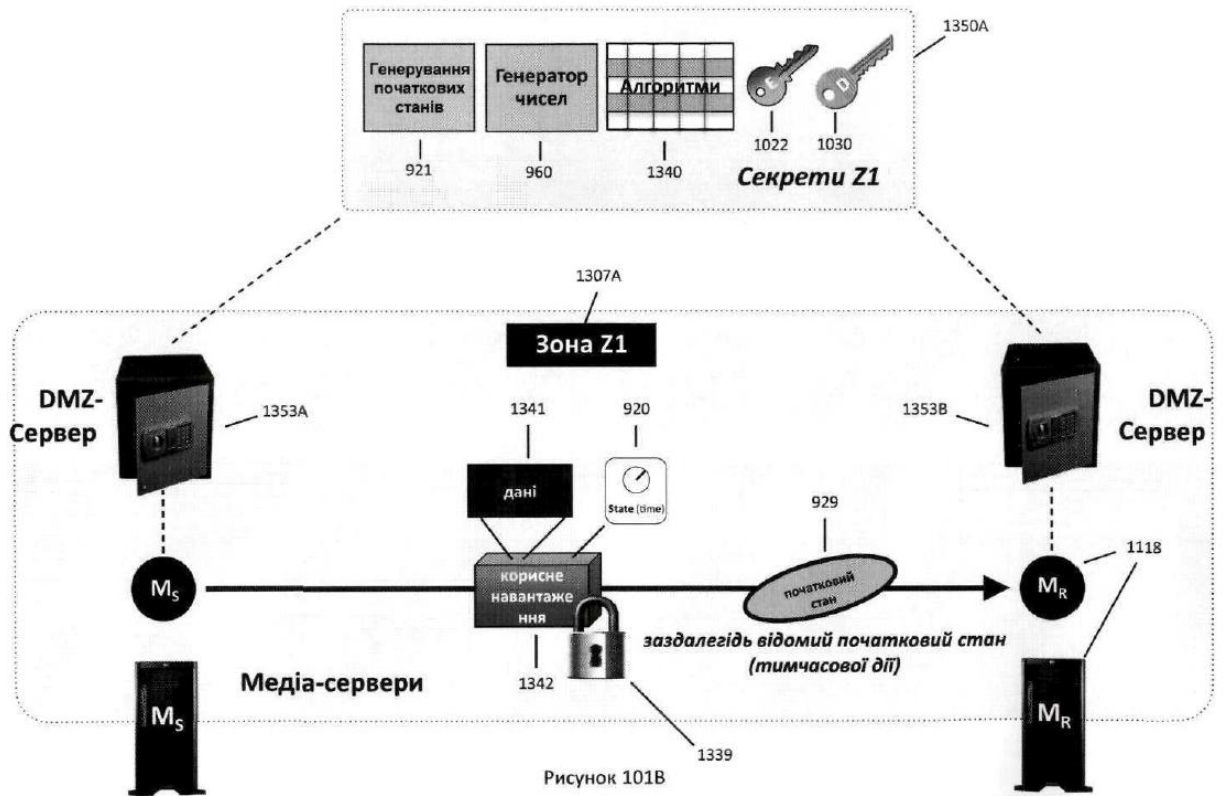


Рисунок 101А



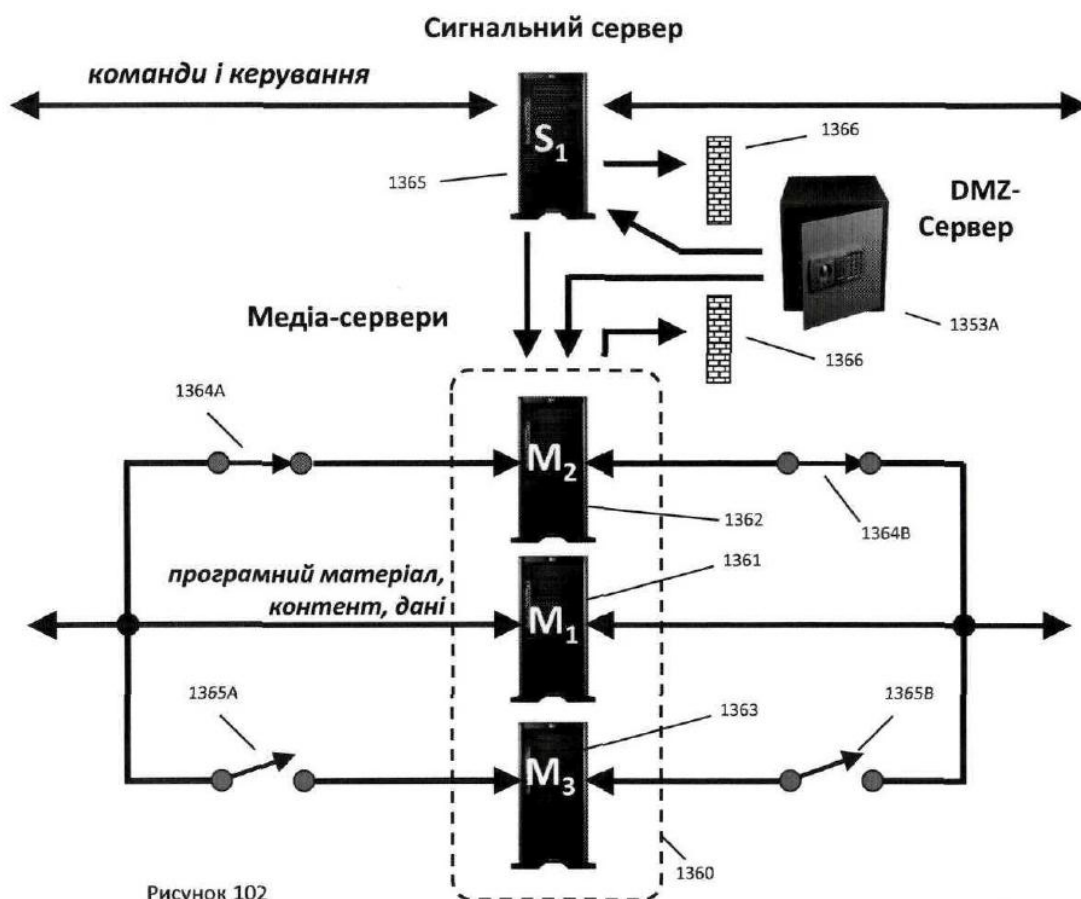


Рисунок 102

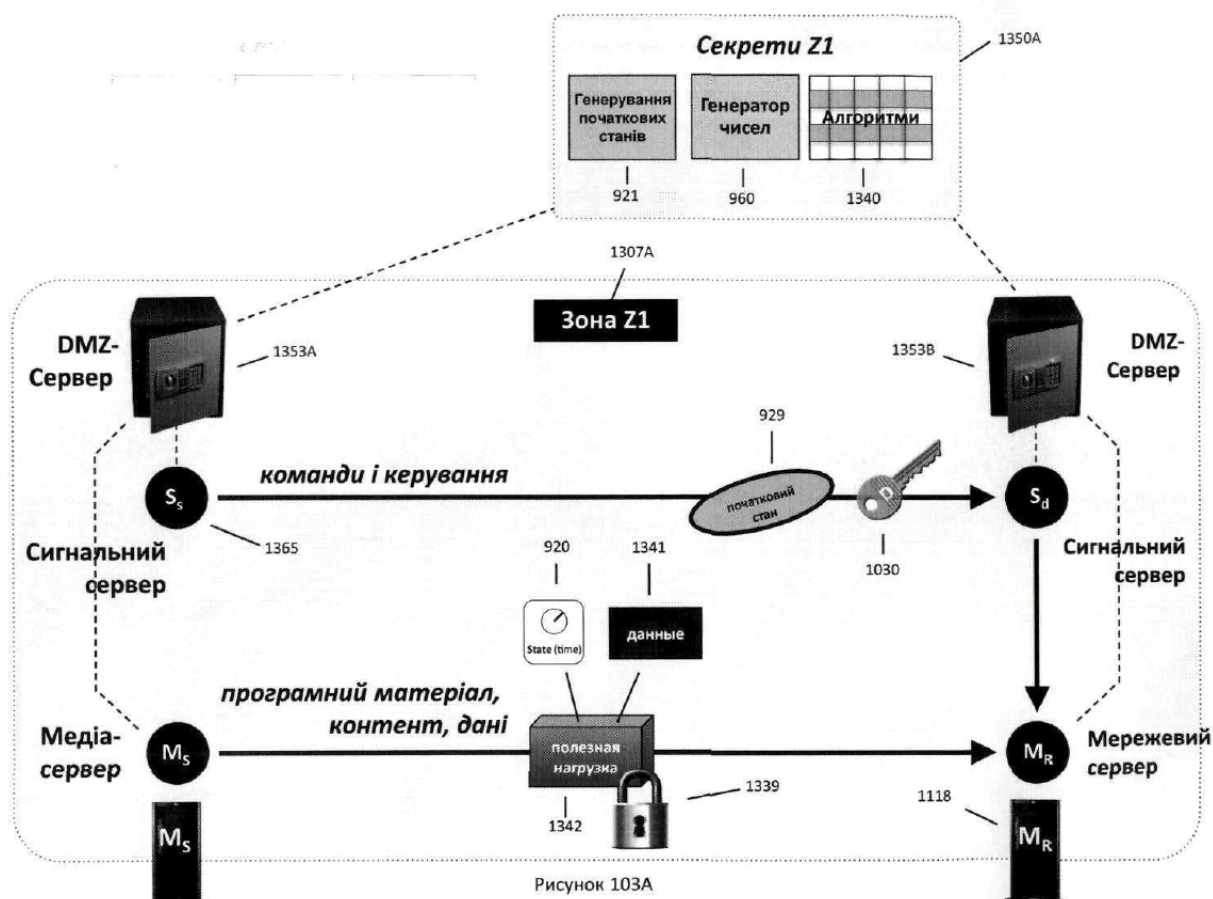
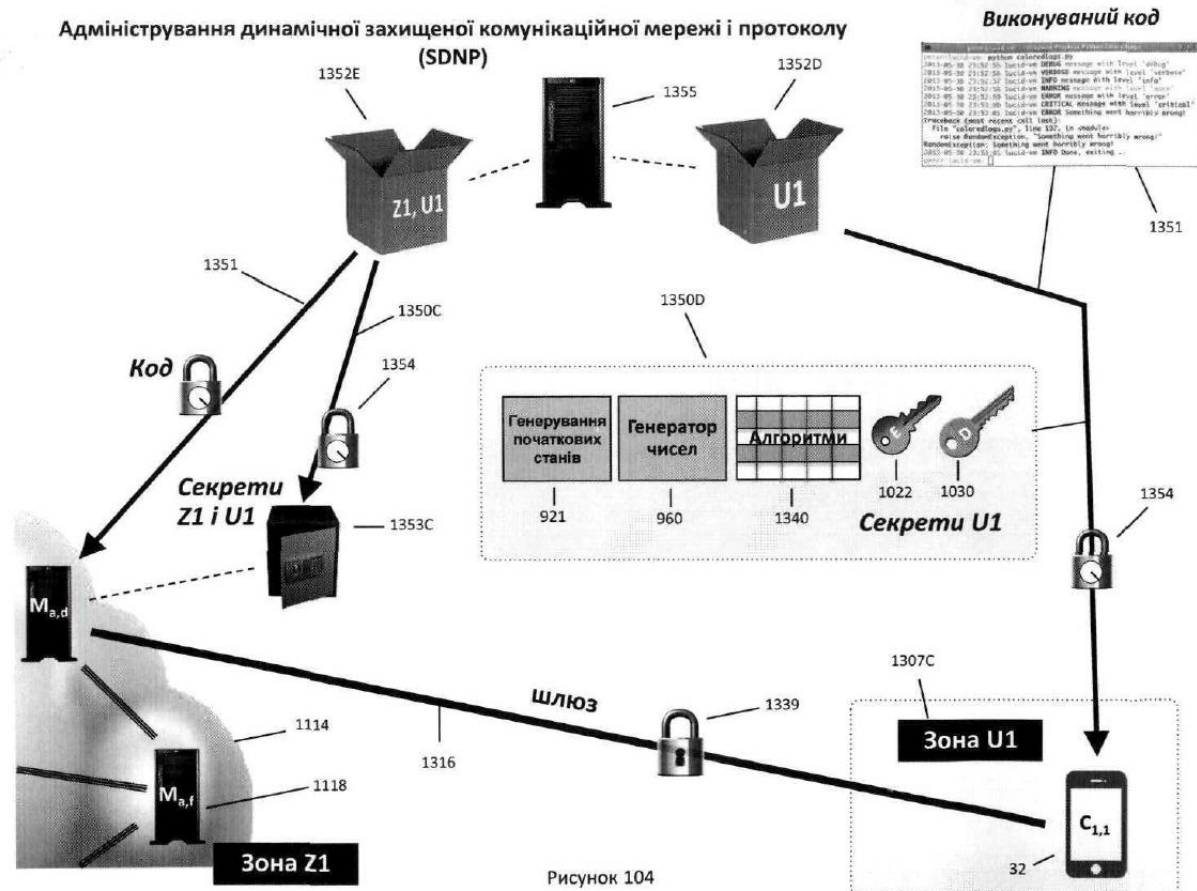
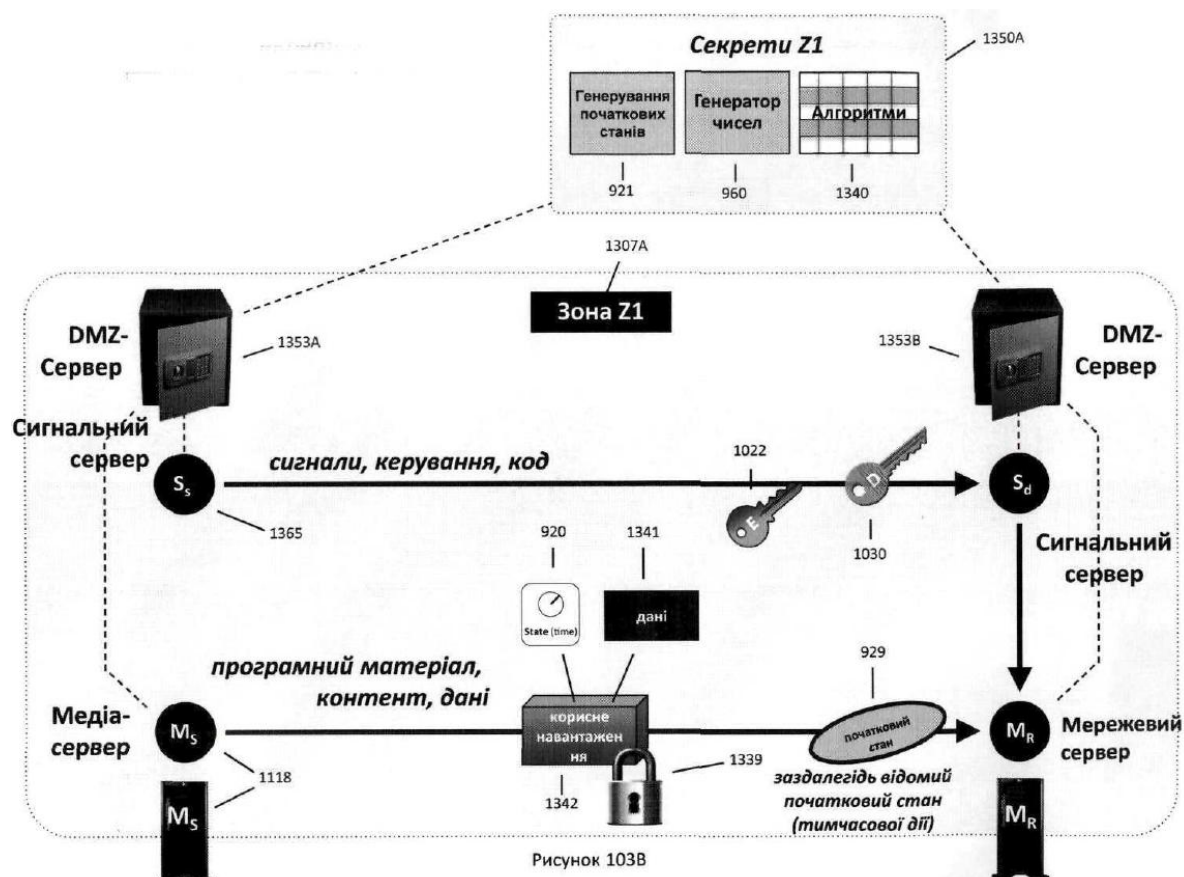
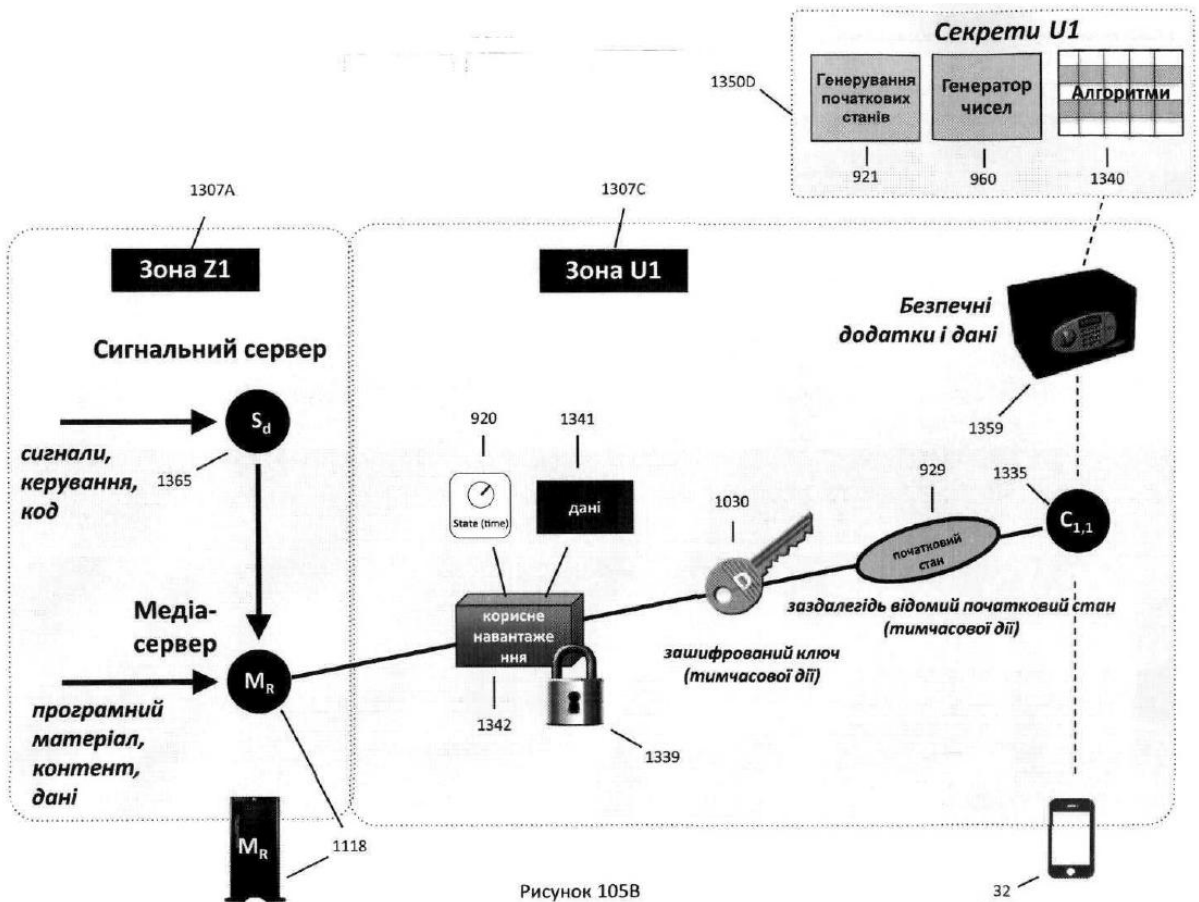
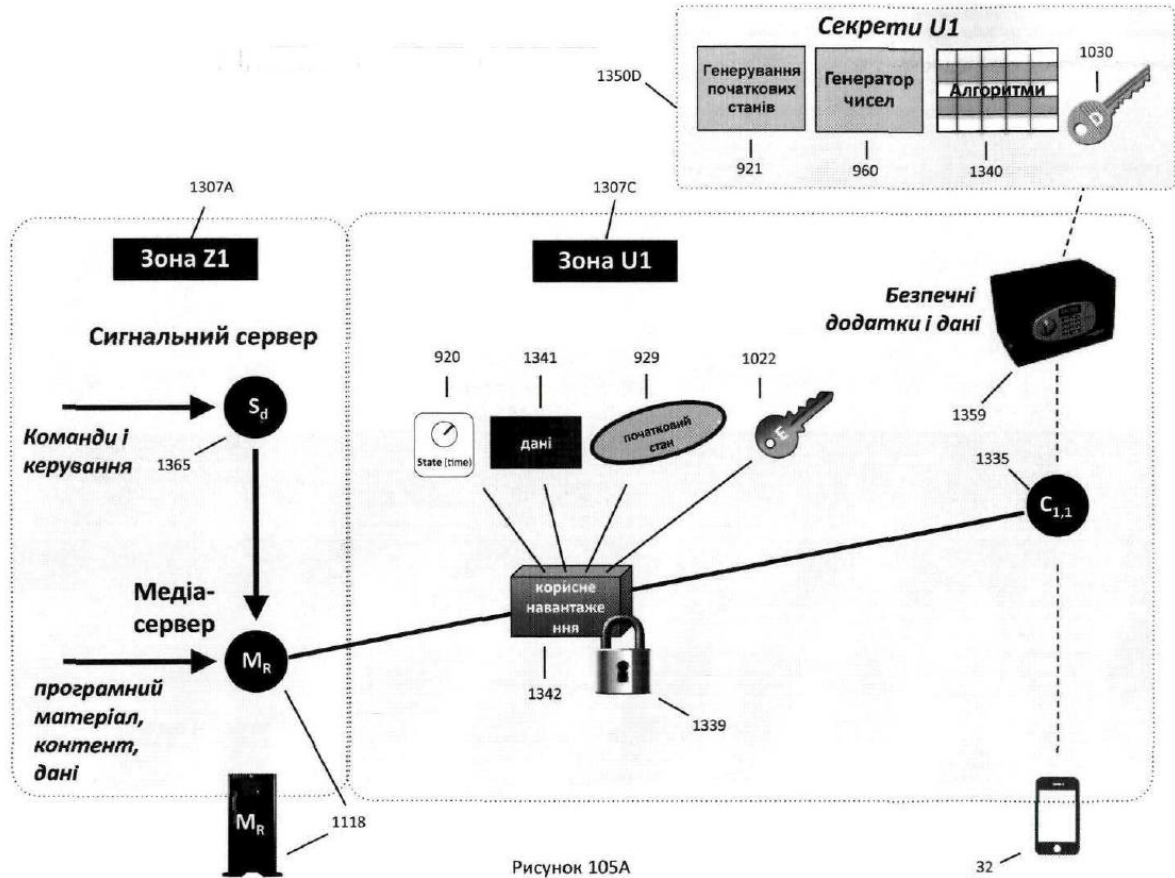


Рисунок 103А





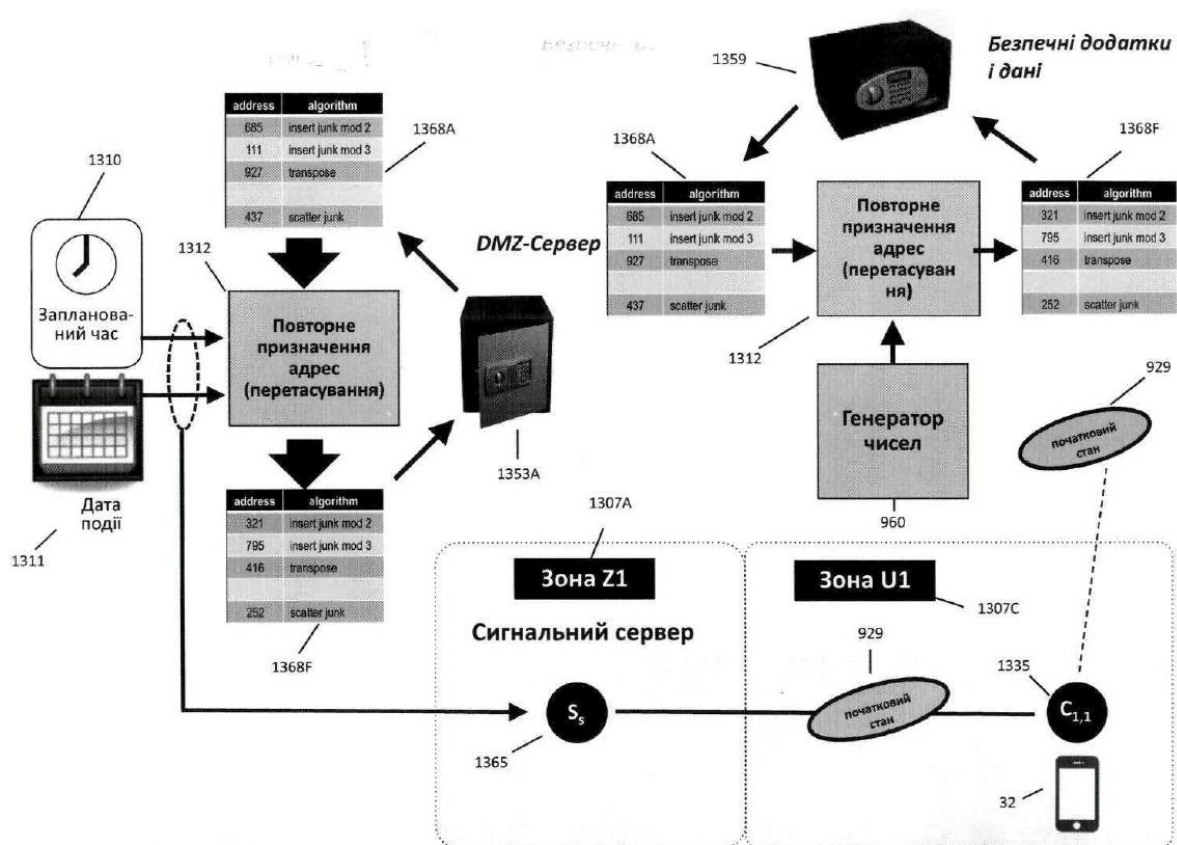


Рисунок 106

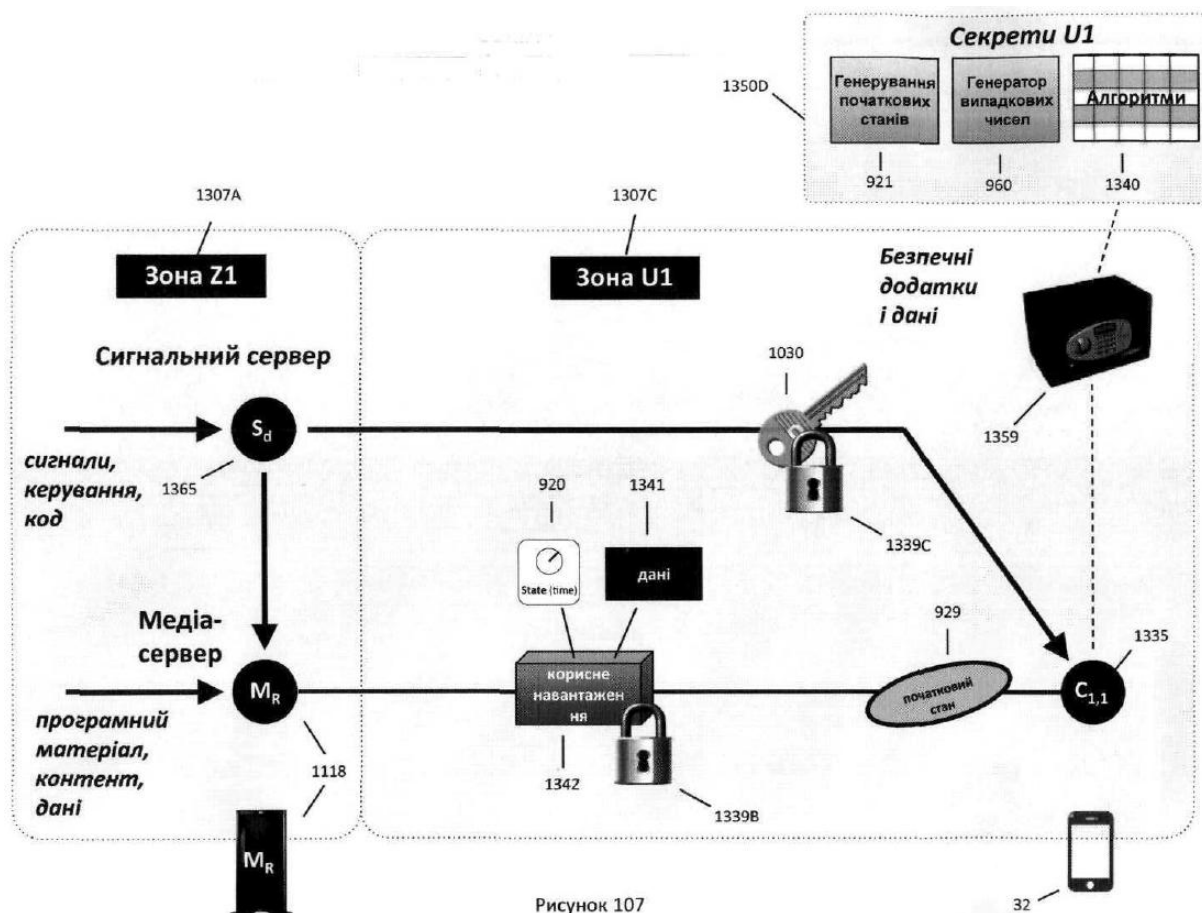
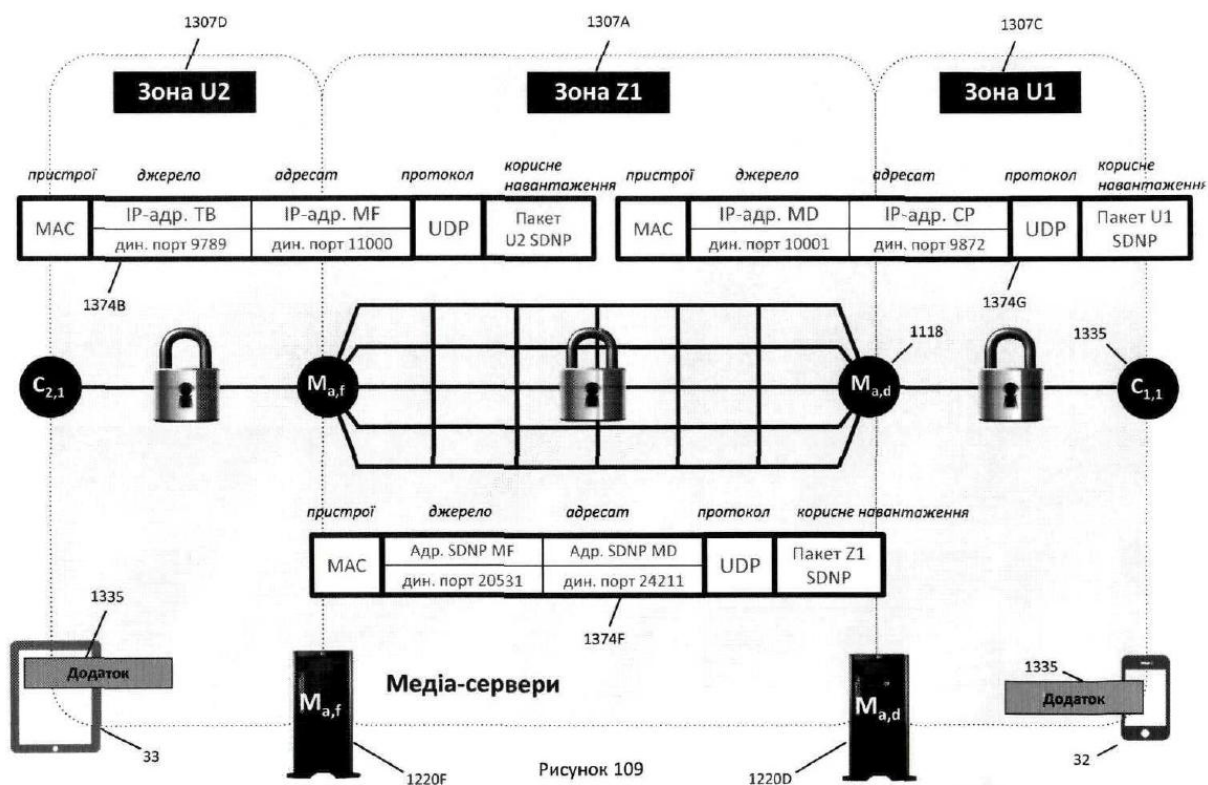
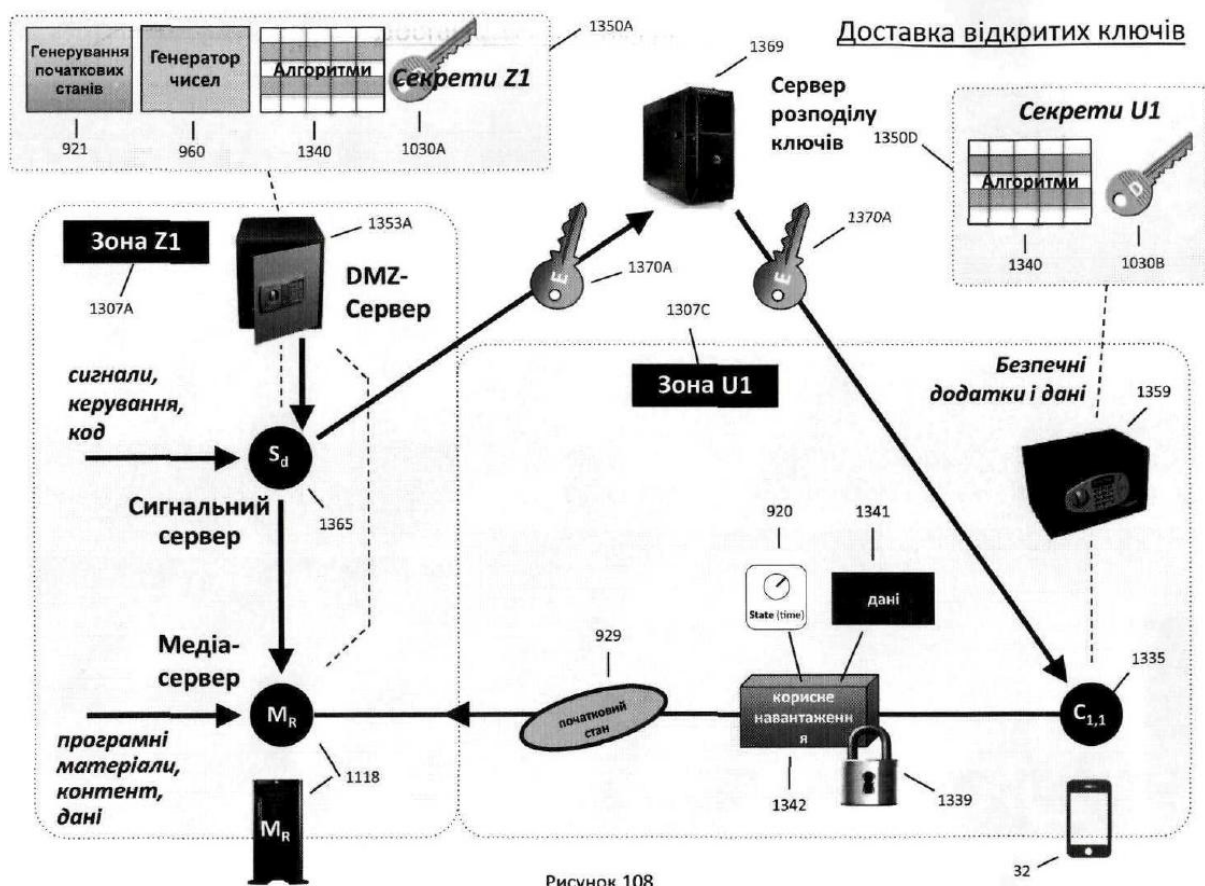


Рисунок 107



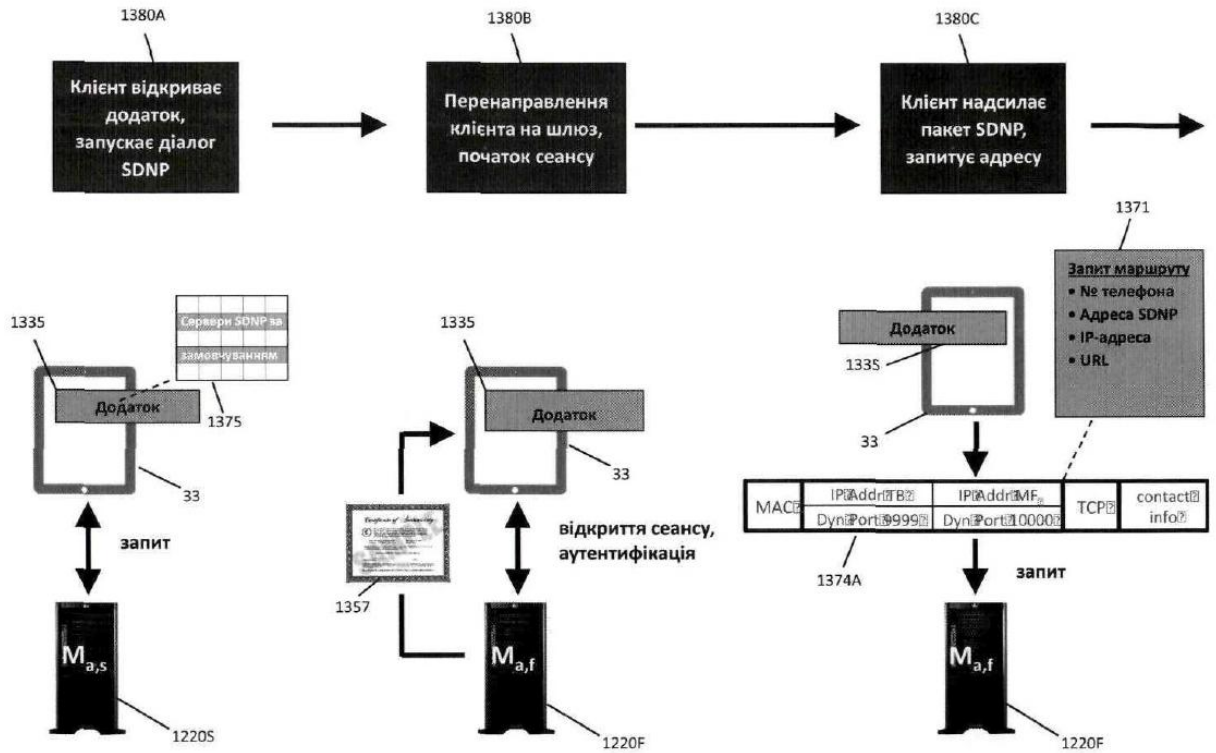


Рисунок 110A

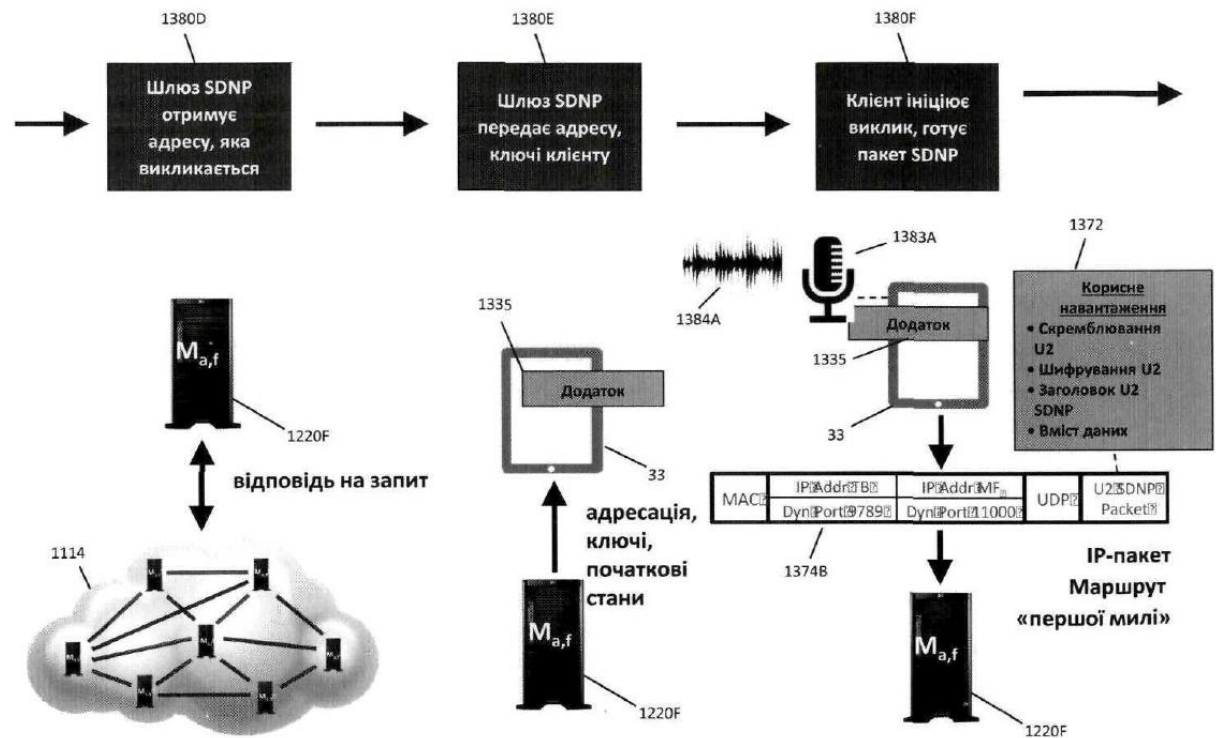
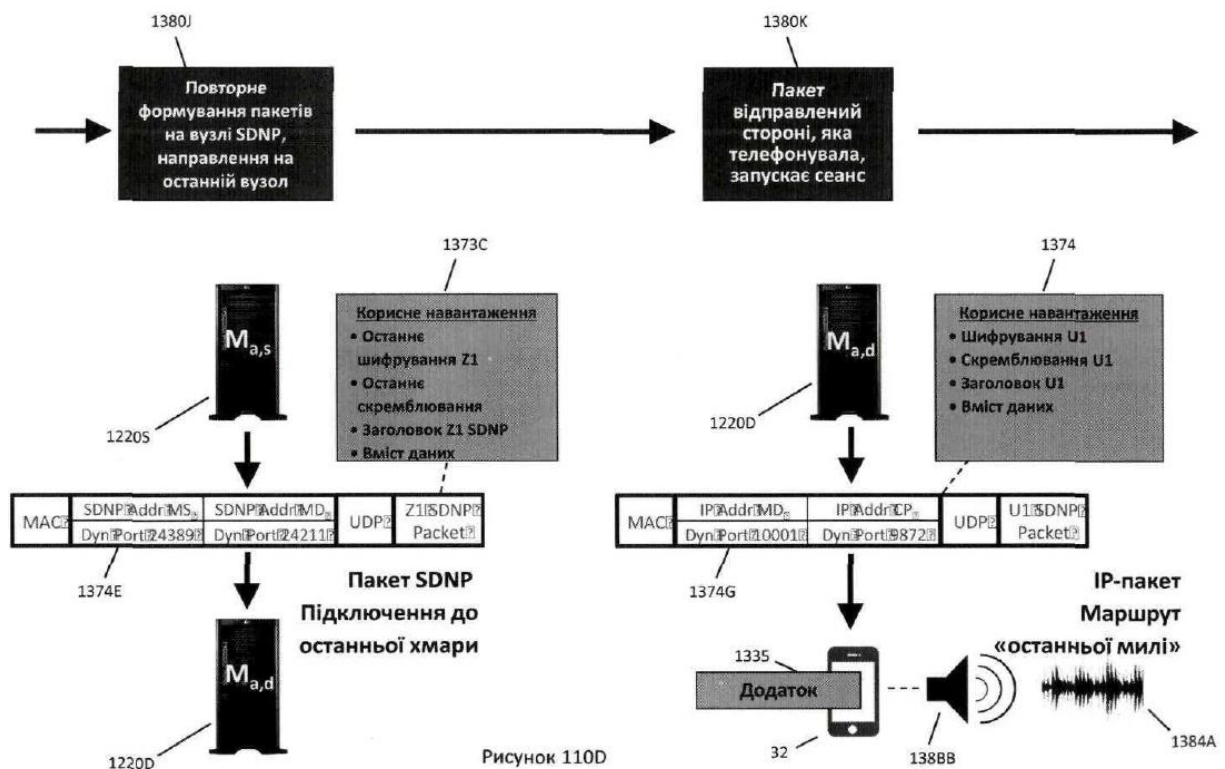
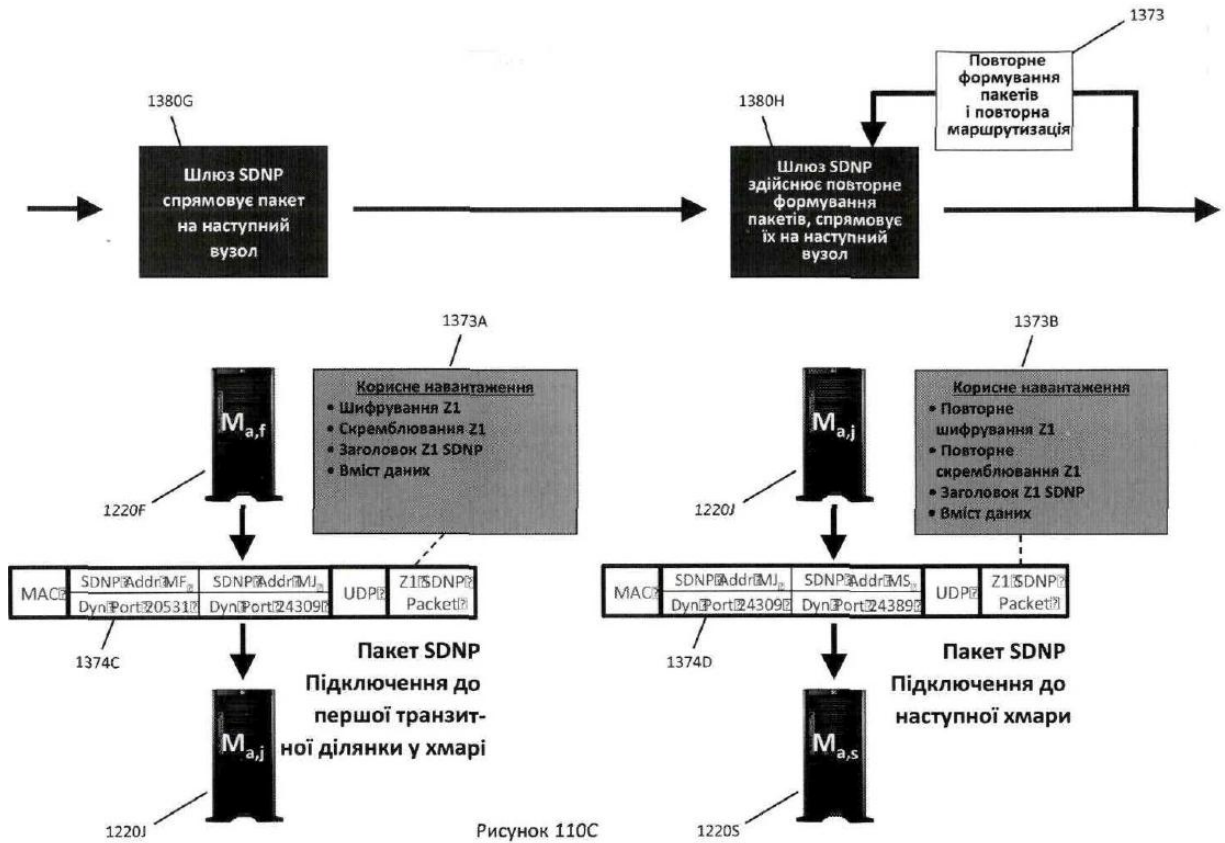


Рисунок 110B



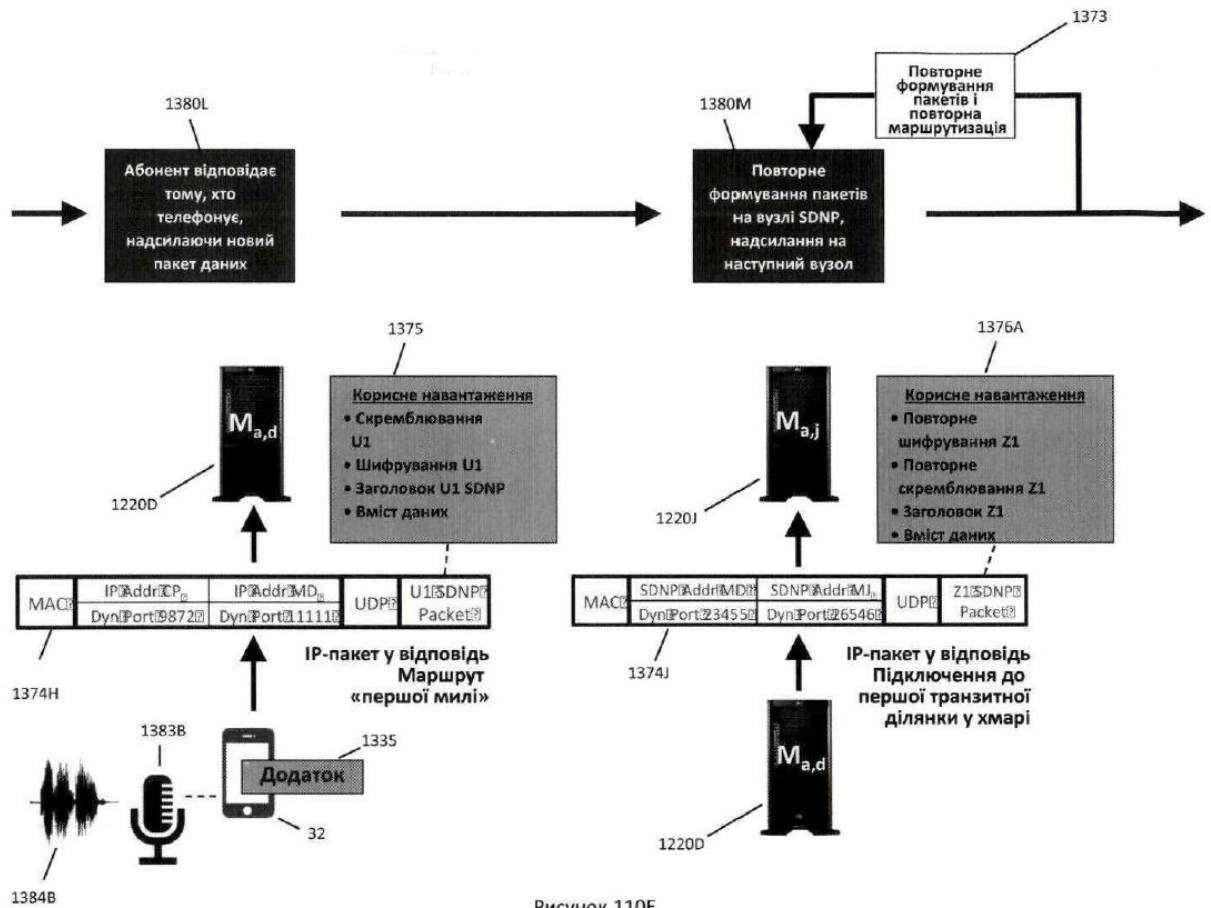


Рисунок 110E

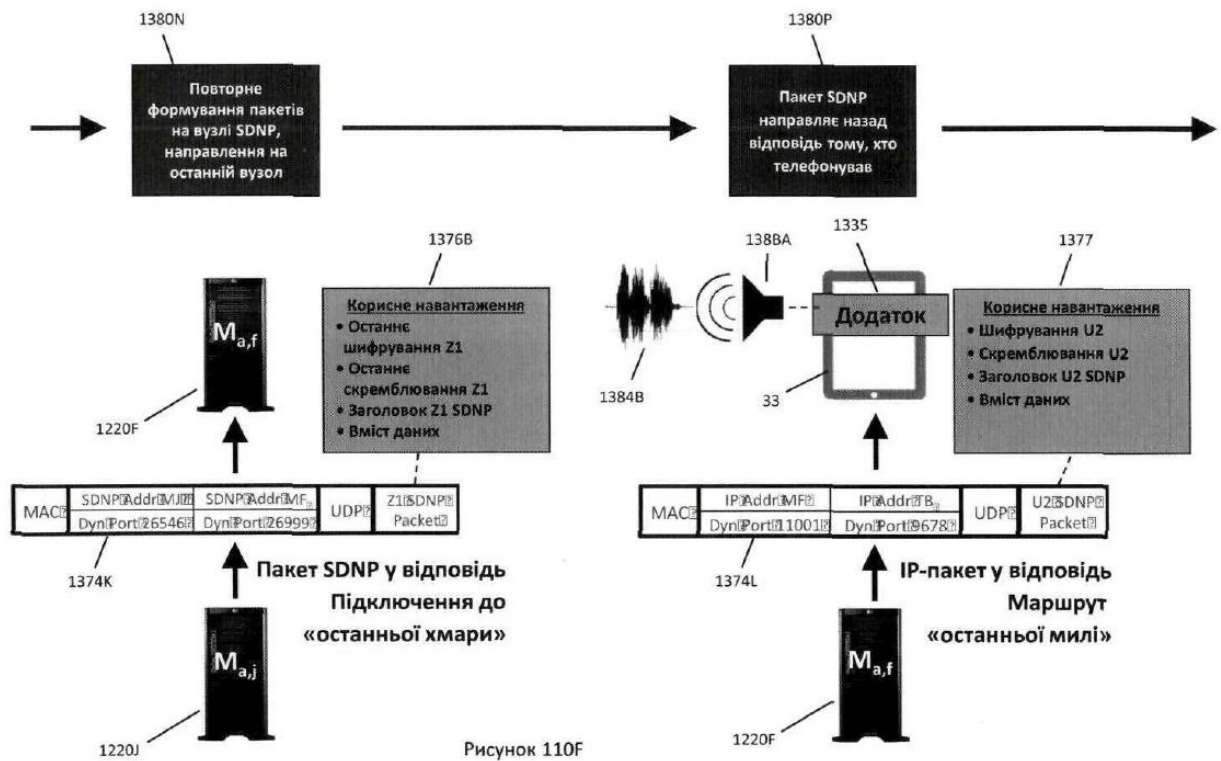


Рисунок 110F

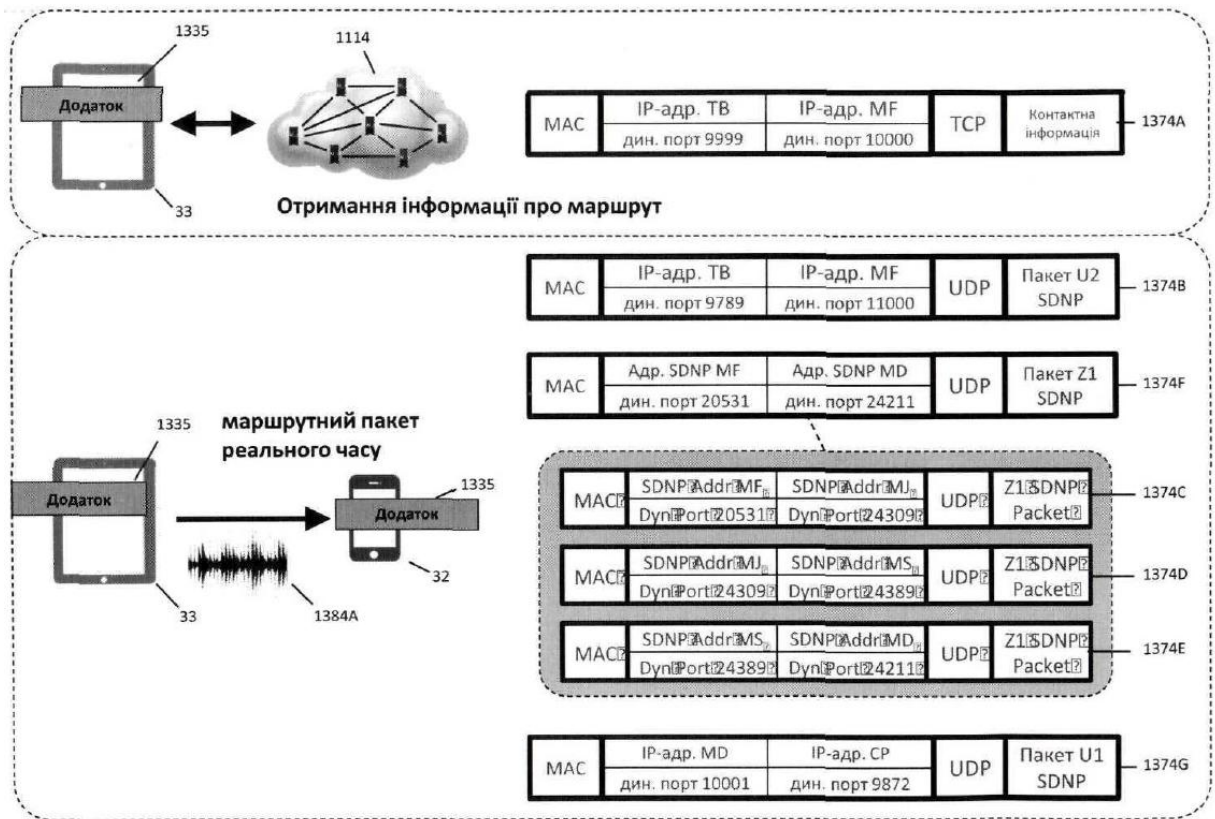


Рисунок 111A

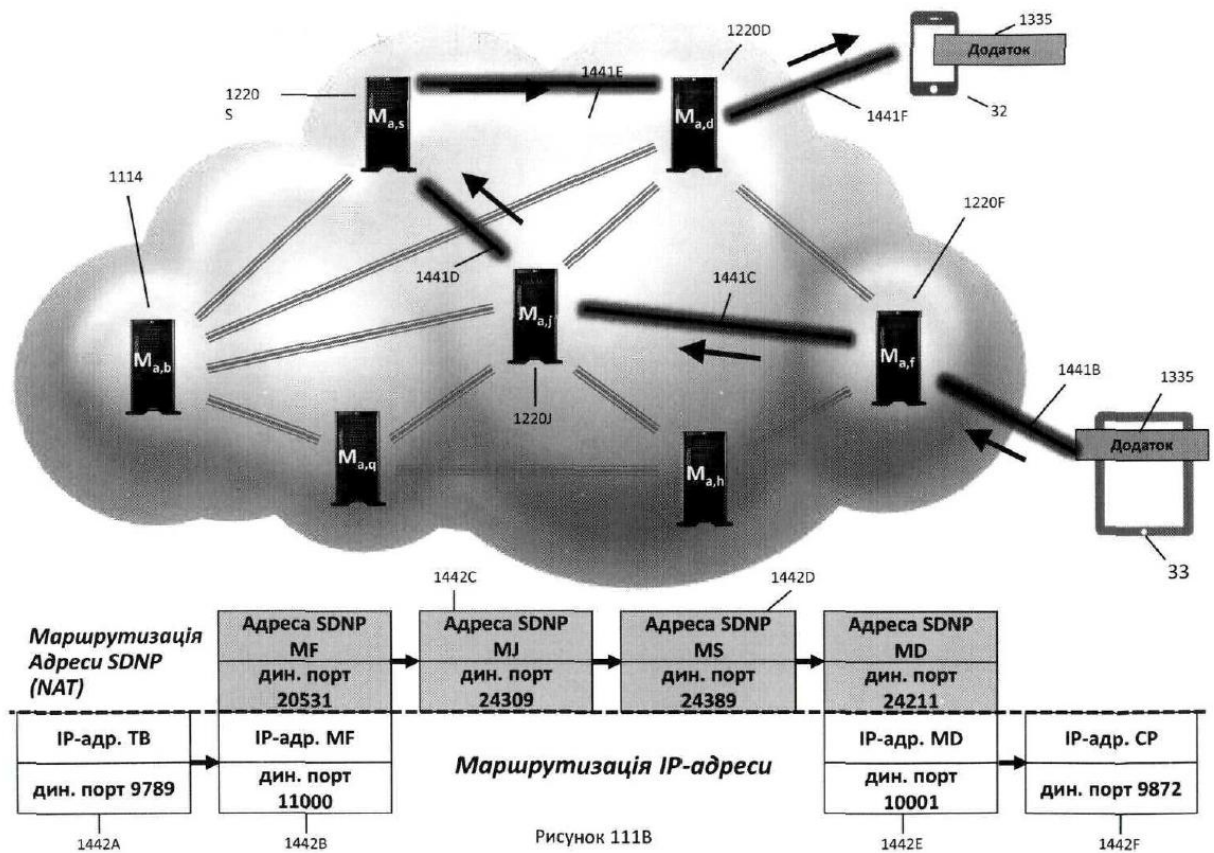


Рисунок 111B

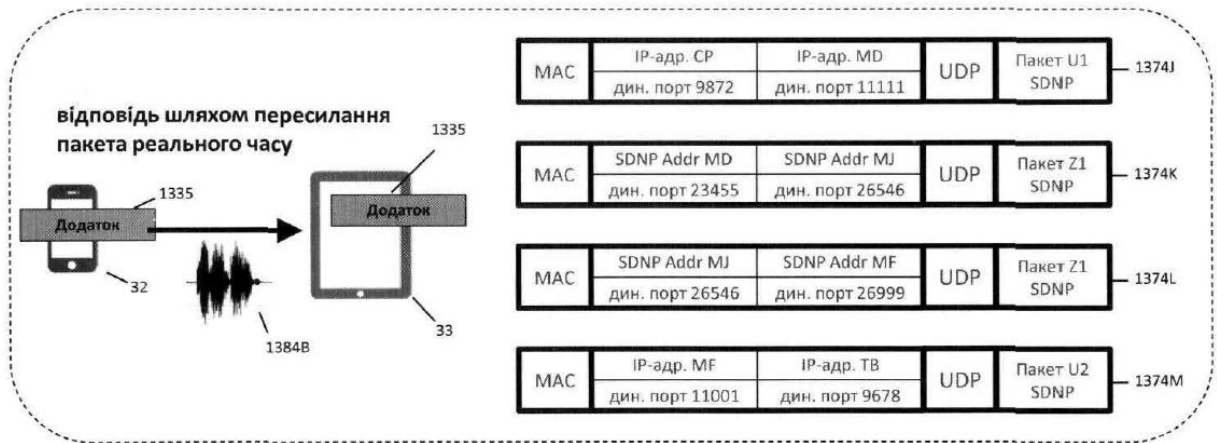


Рисунок 112A

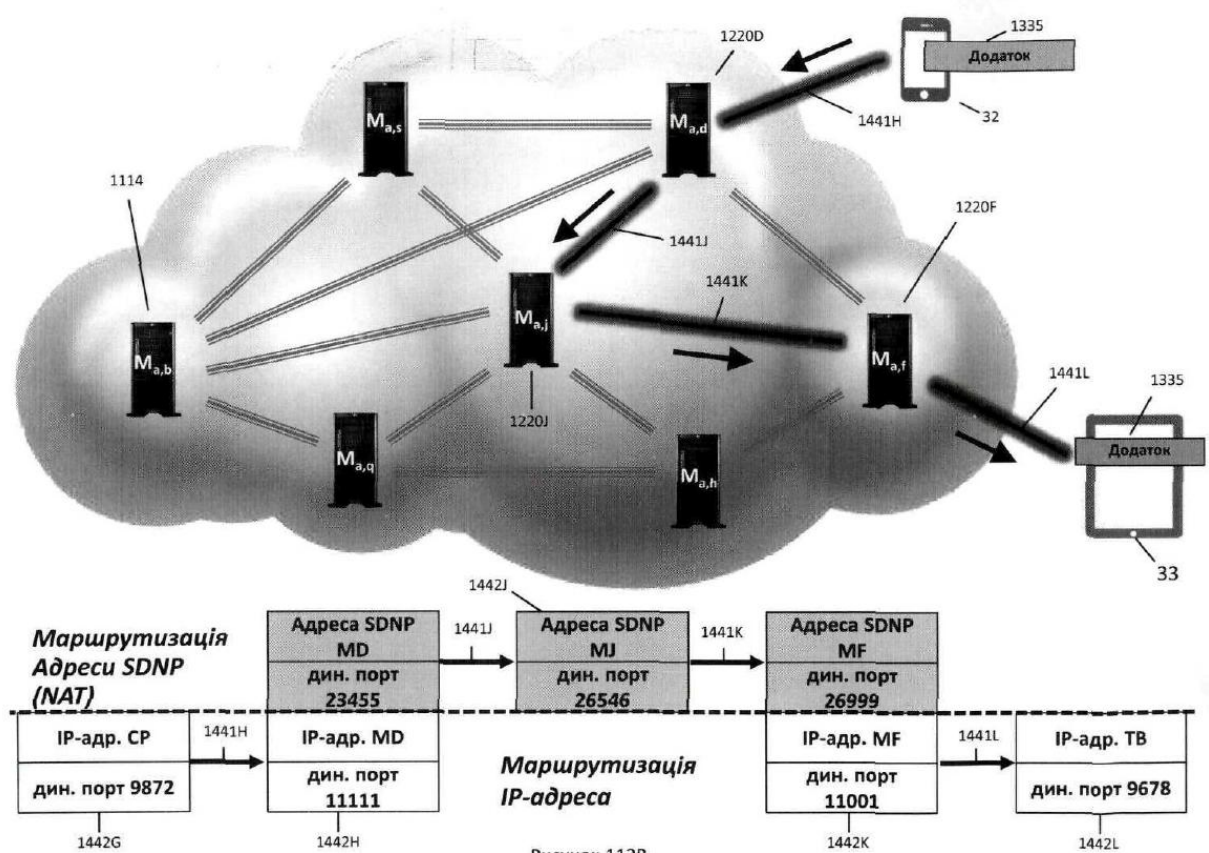


Рисунок 112B

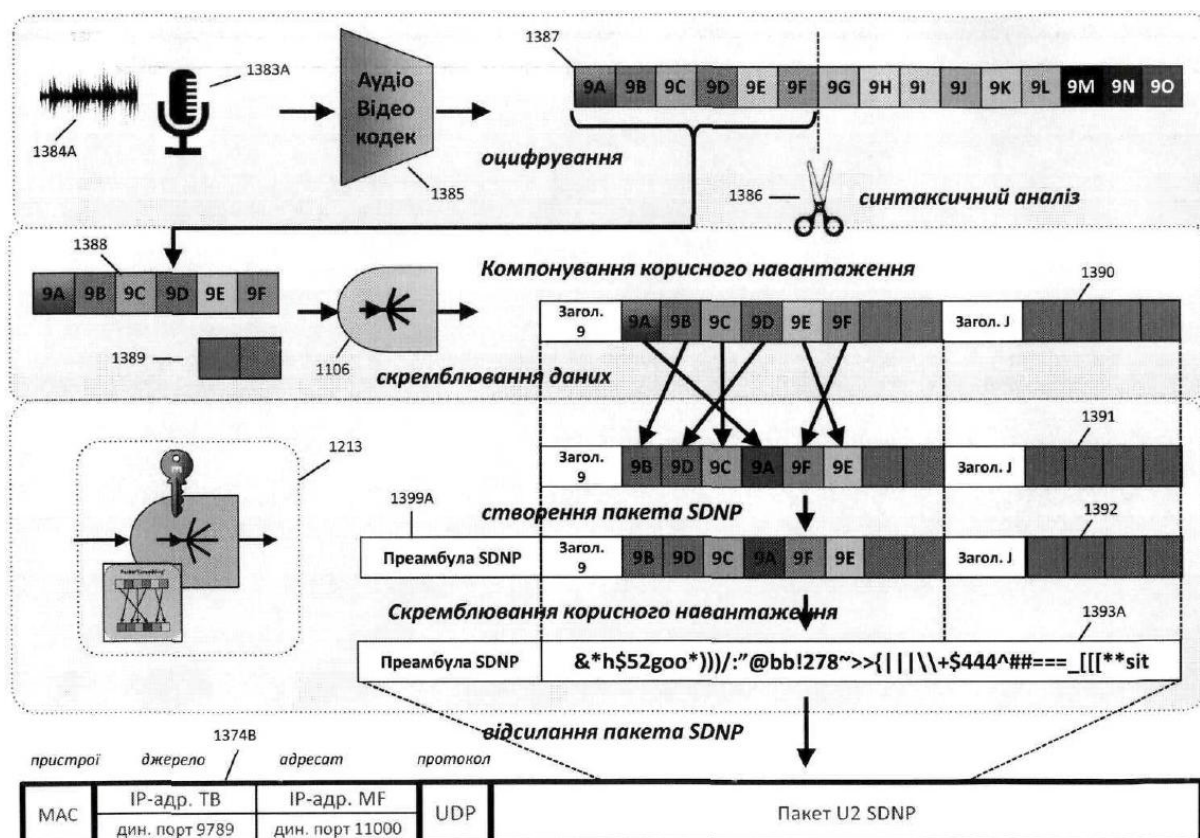


Рисунок 113А

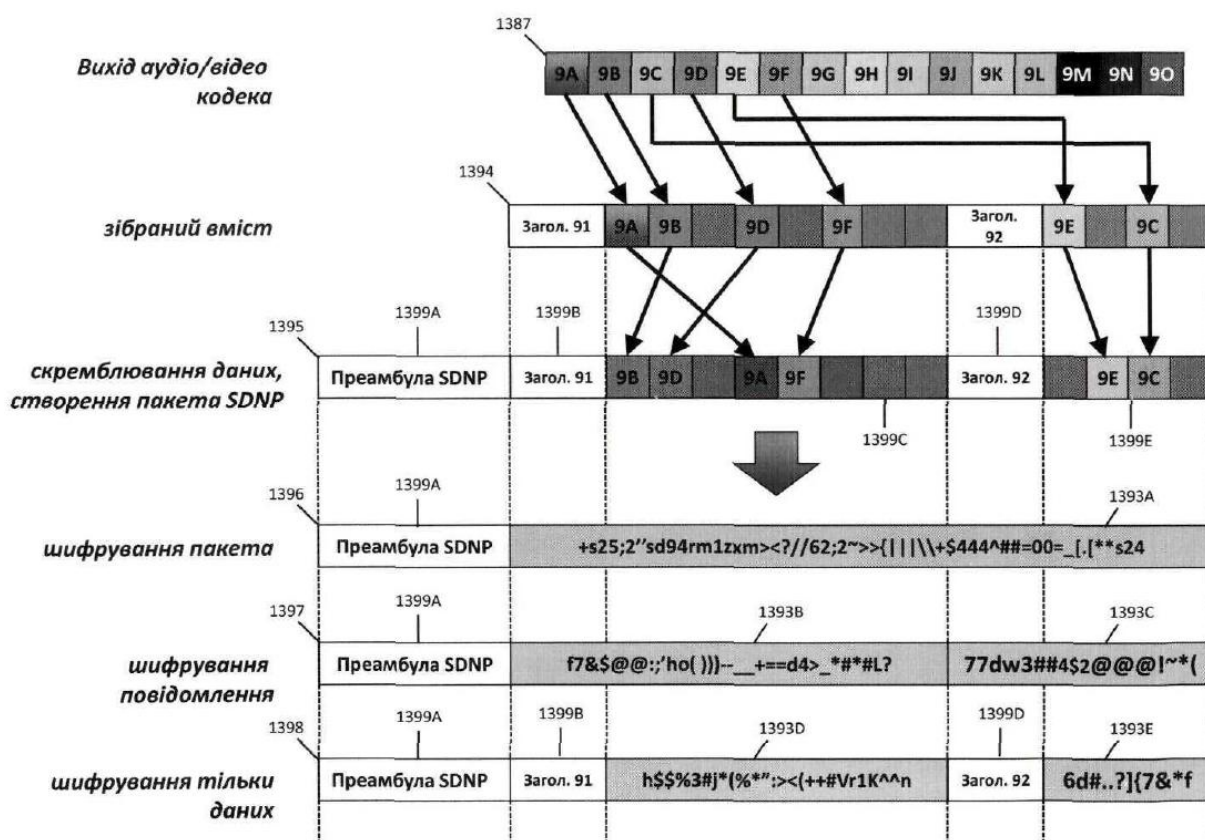


Рисунок 113В

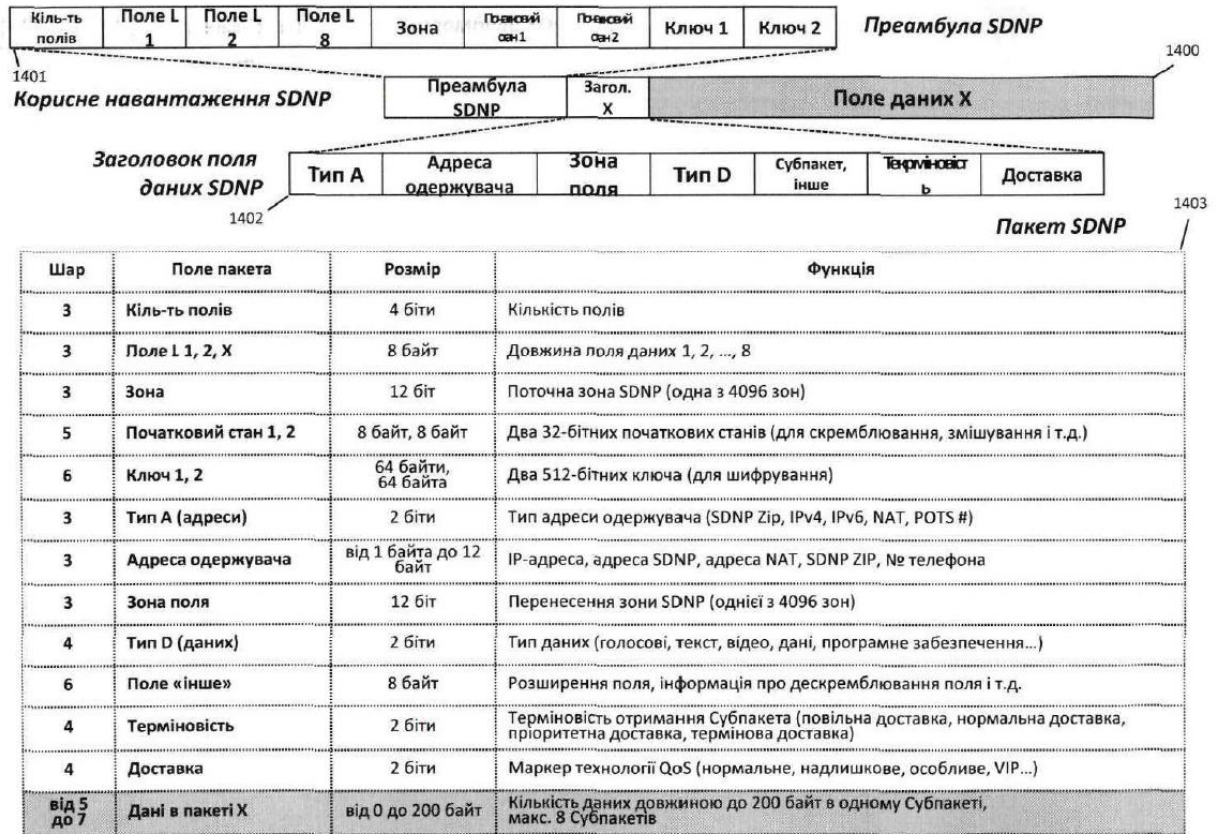
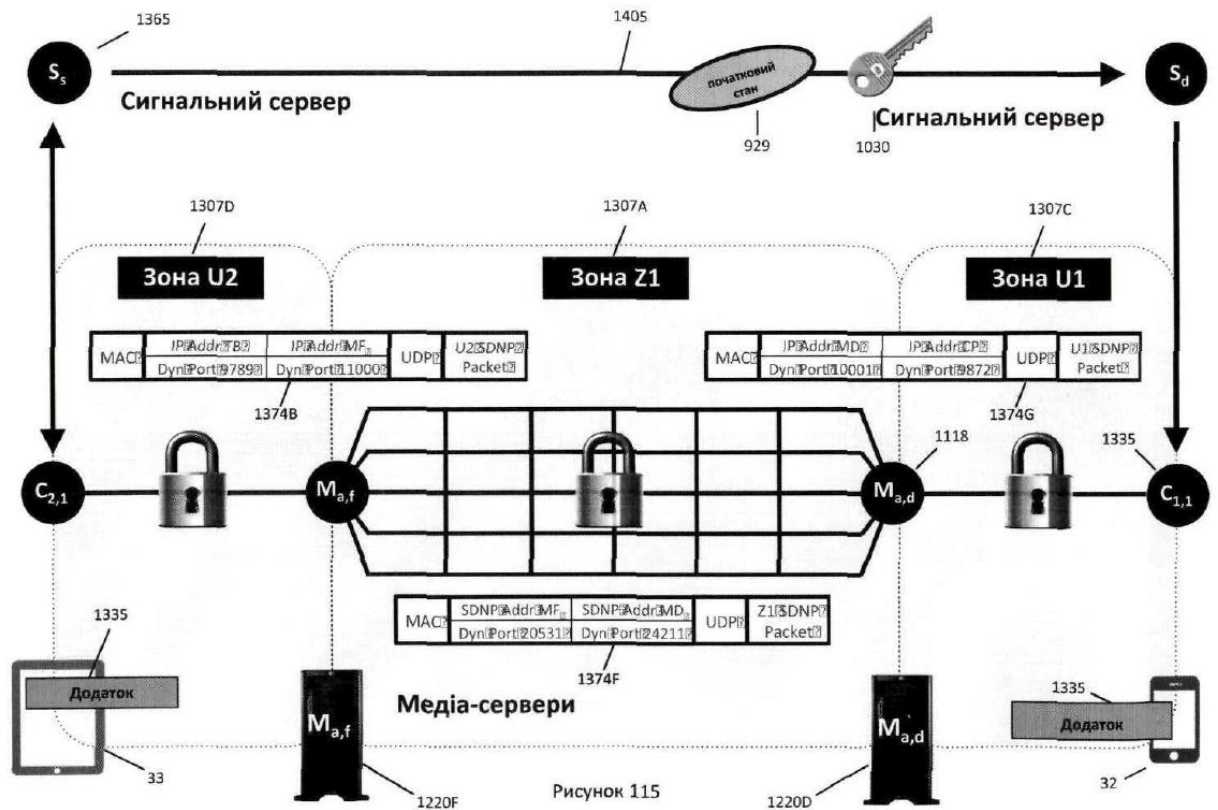


Рисунок 114



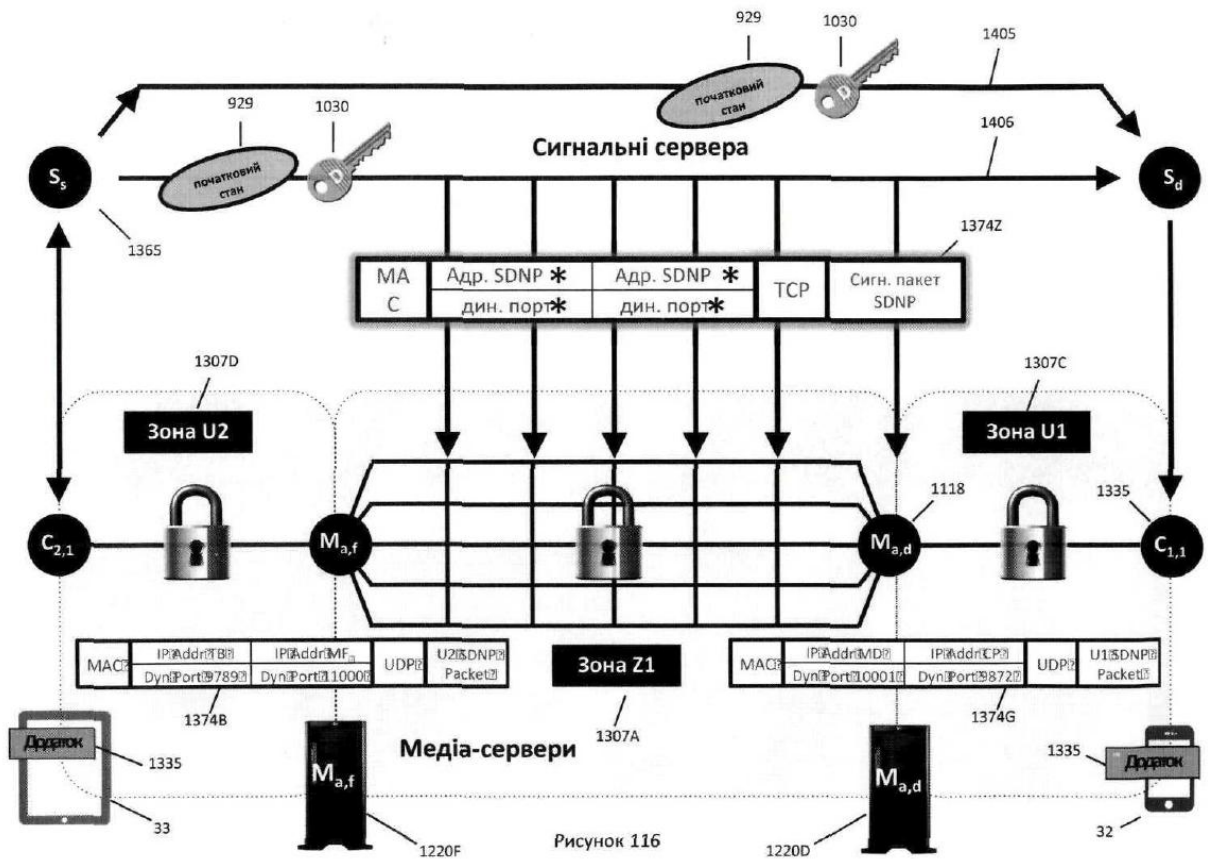


Рисунок 116

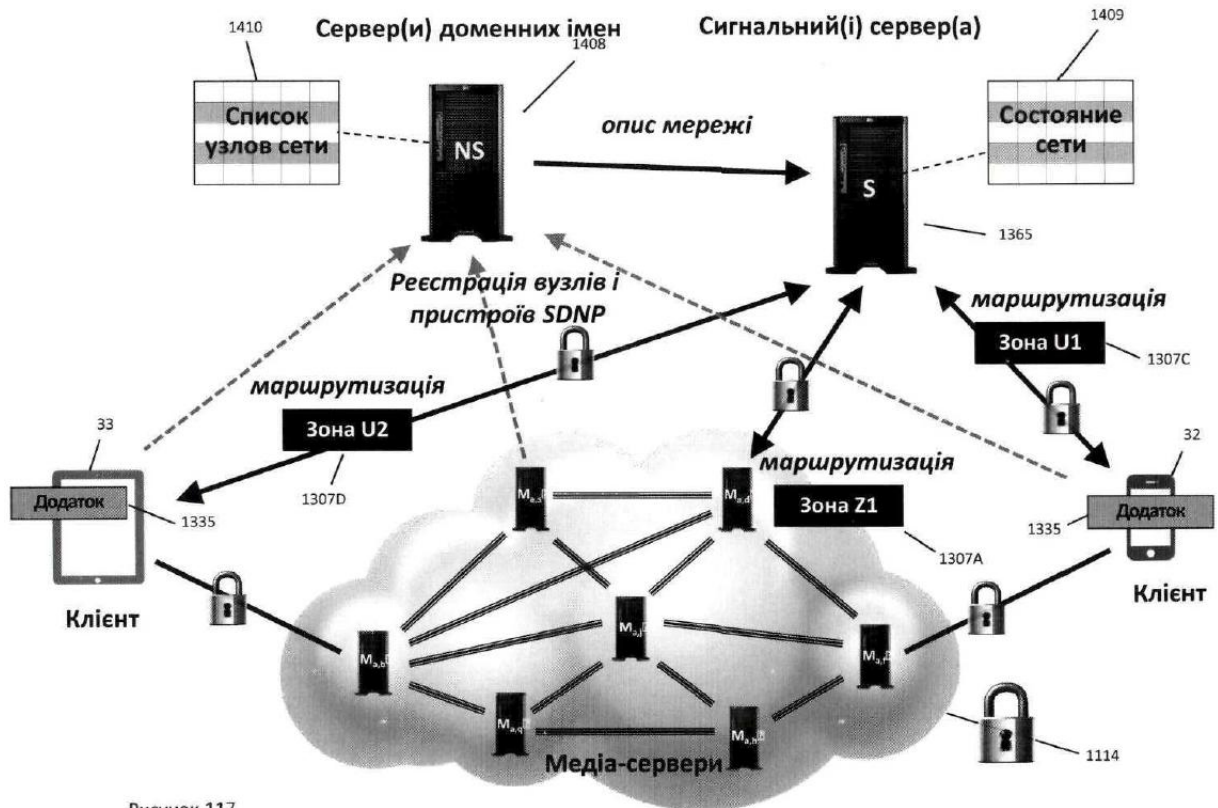


Рисунок 117

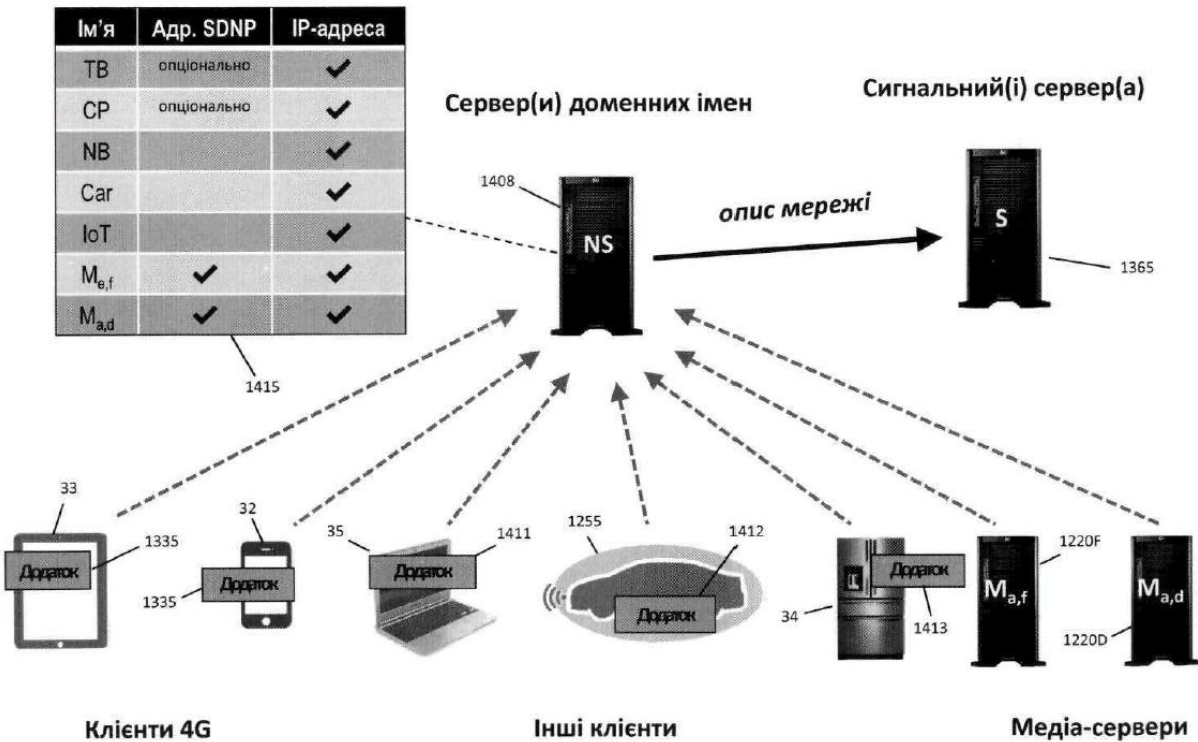
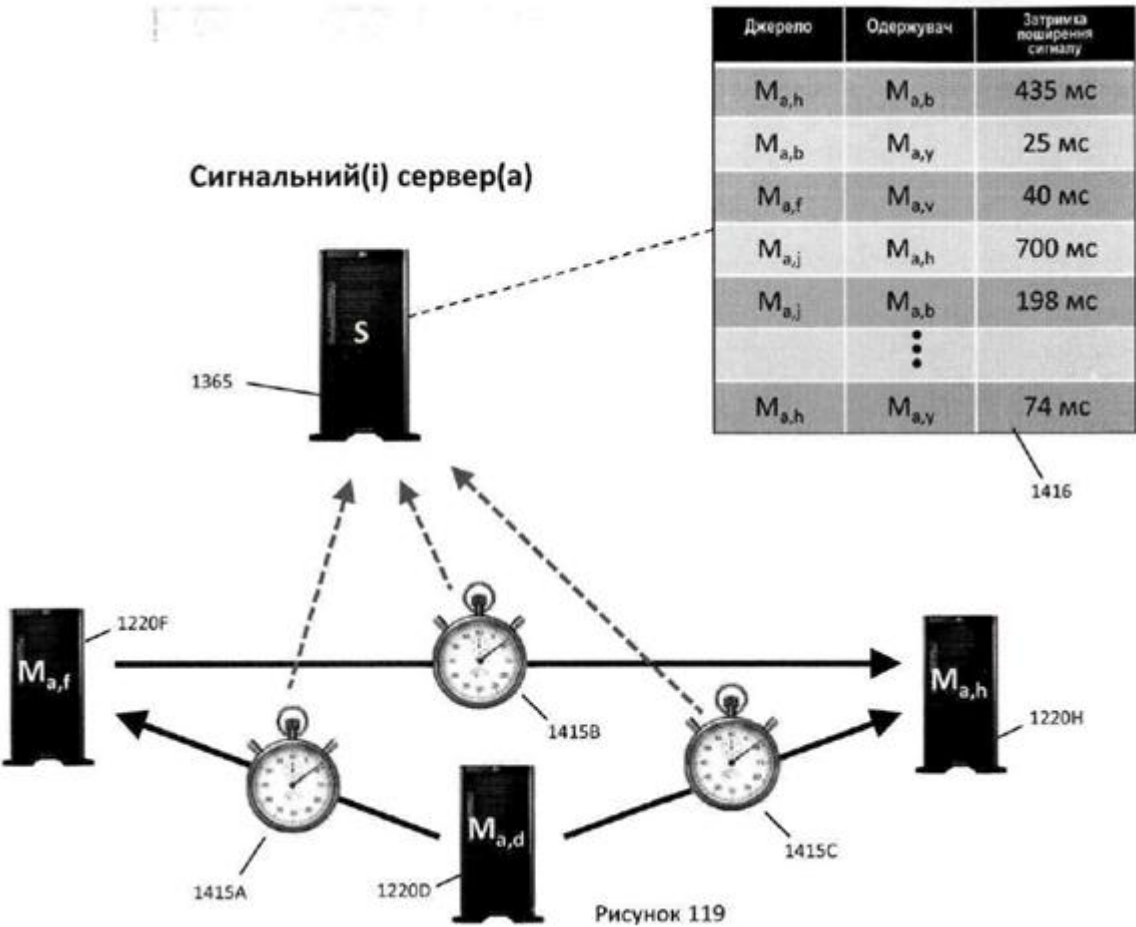
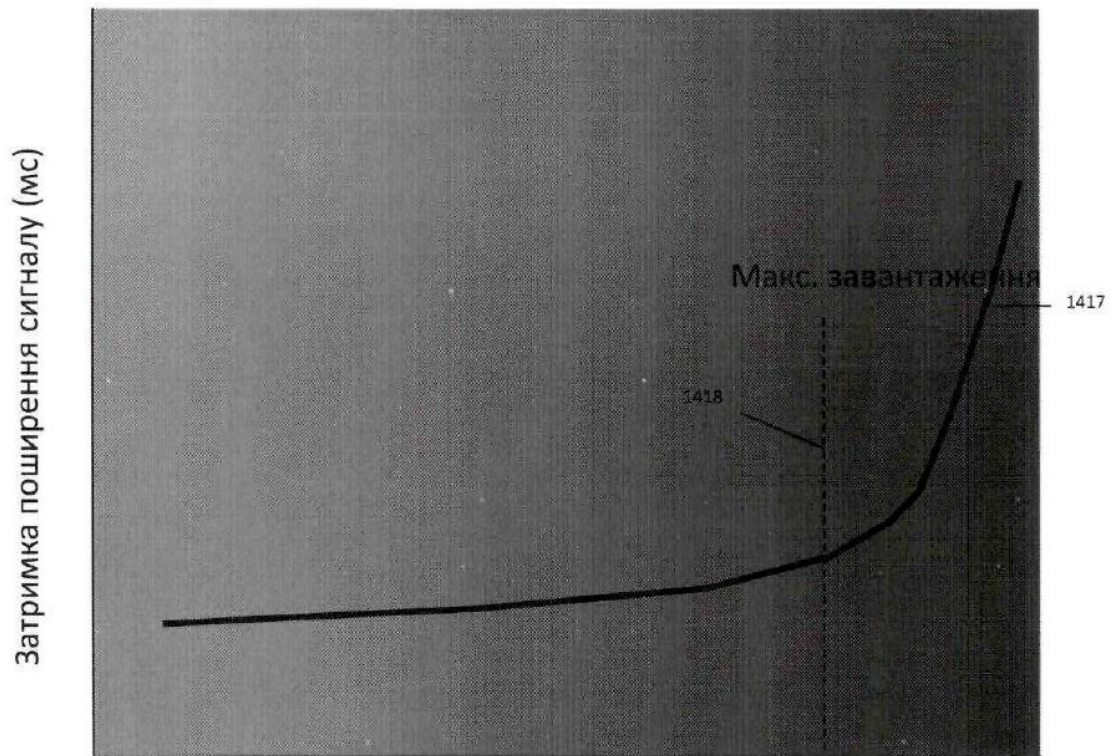


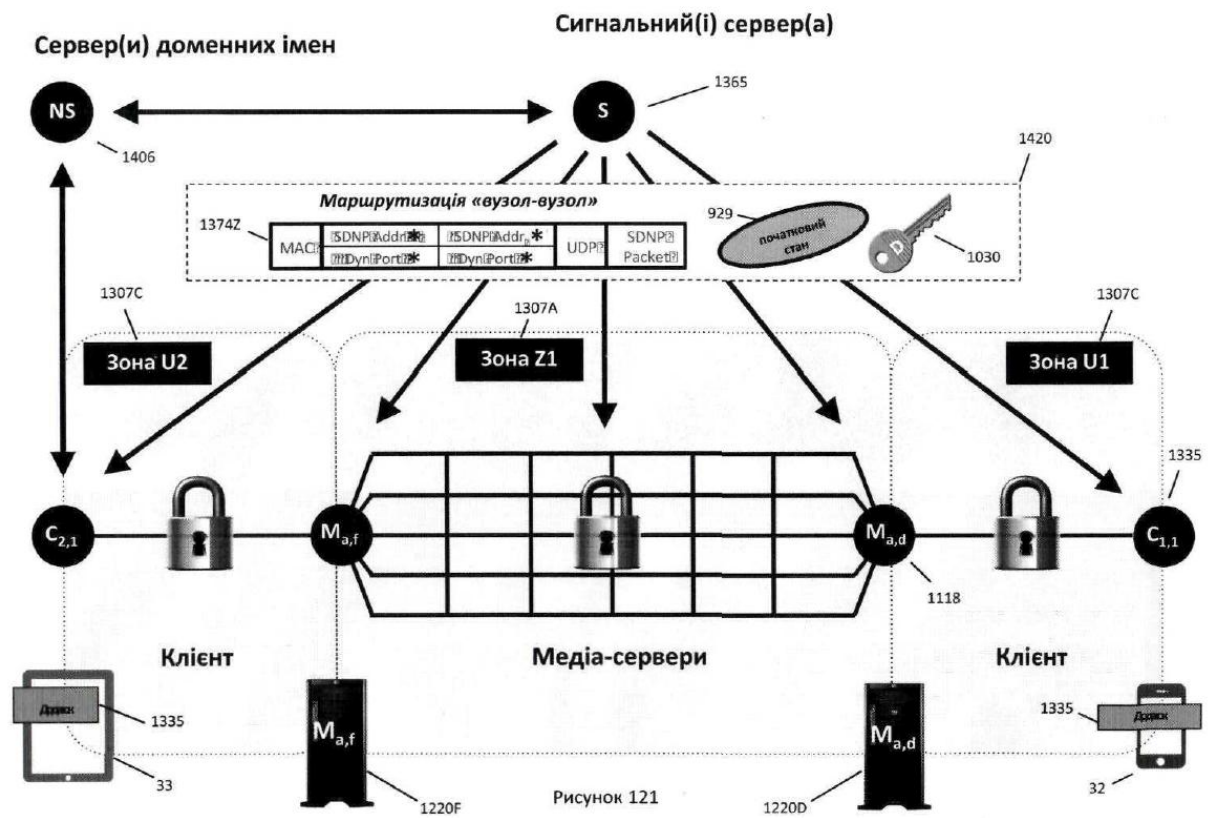
Рисунок 118





Кількість або розмір відправлених тестових пакетів

Рисунок 120



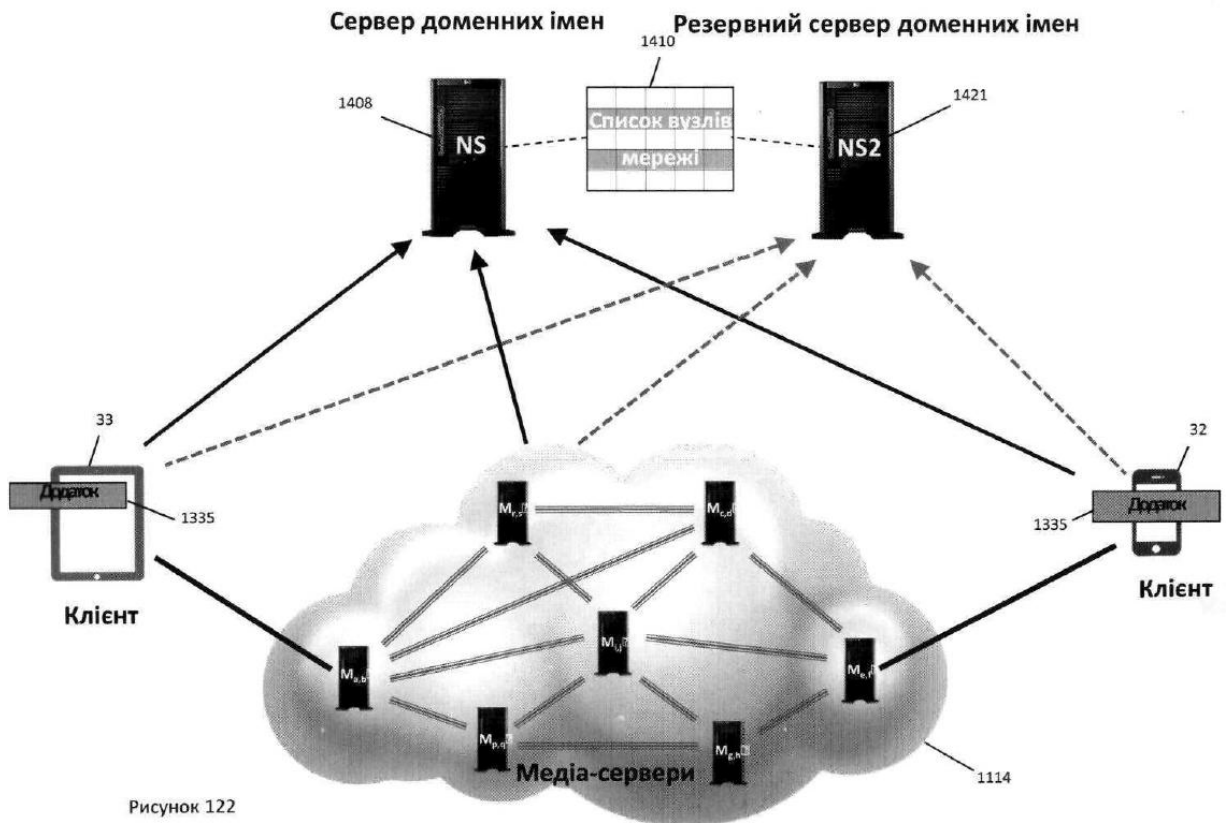


Рисунок 122

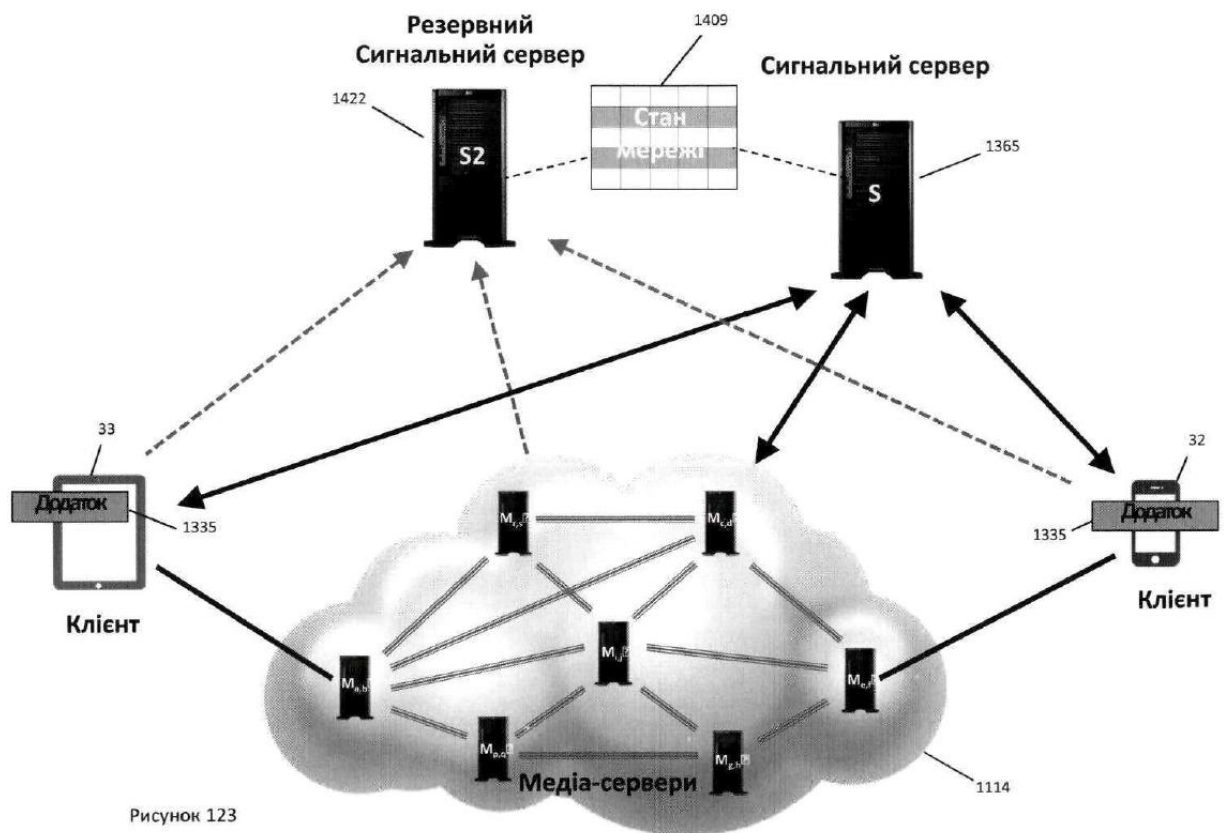


Рисунок 123

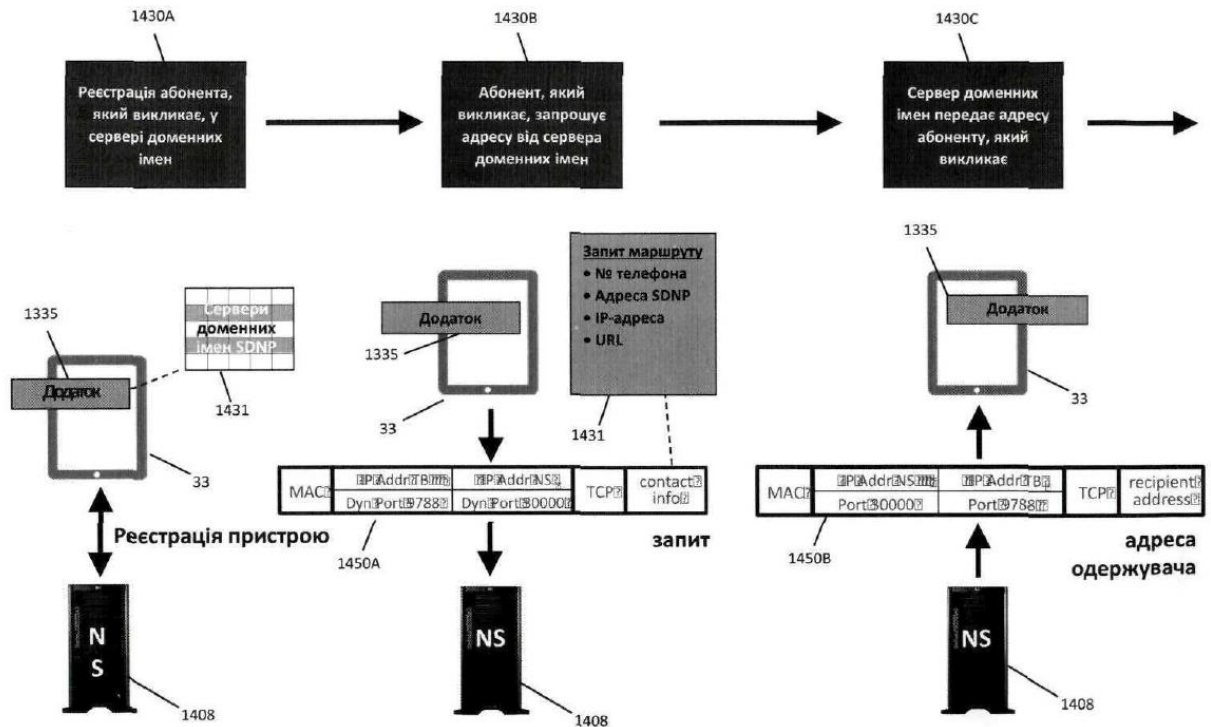


Рисунок 124А

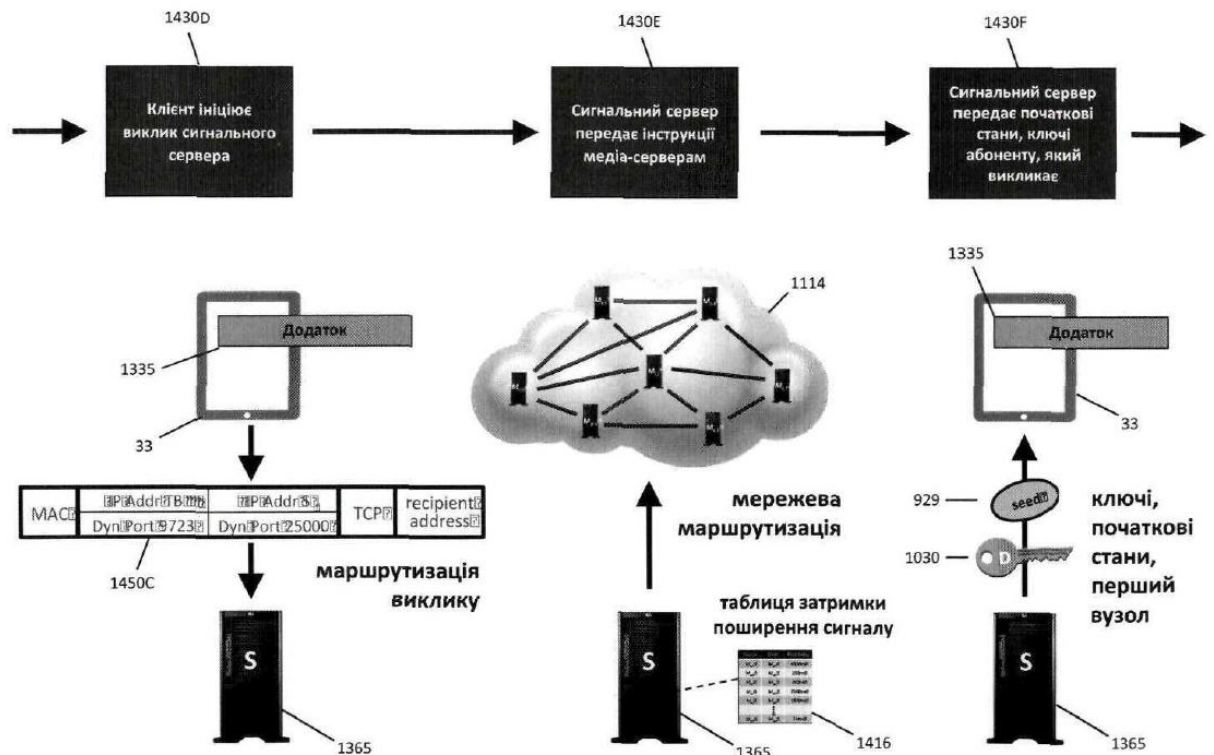
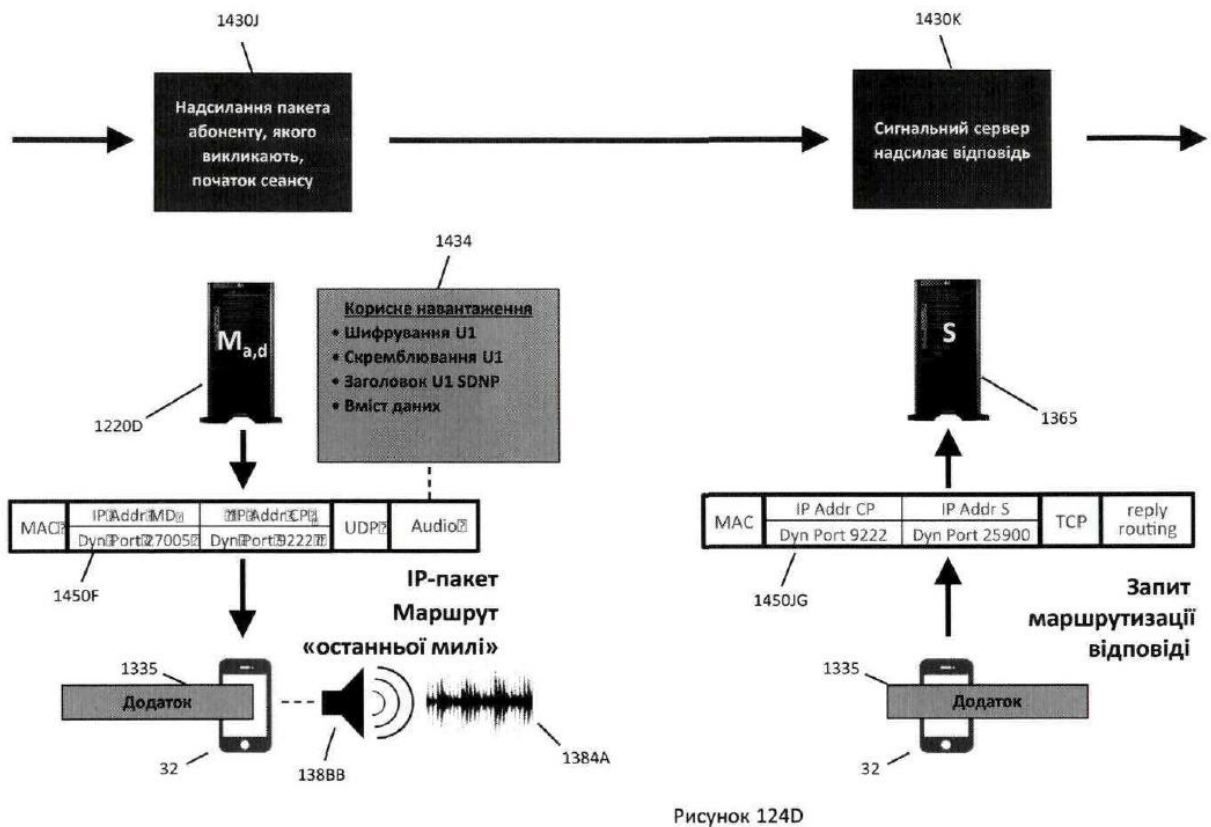
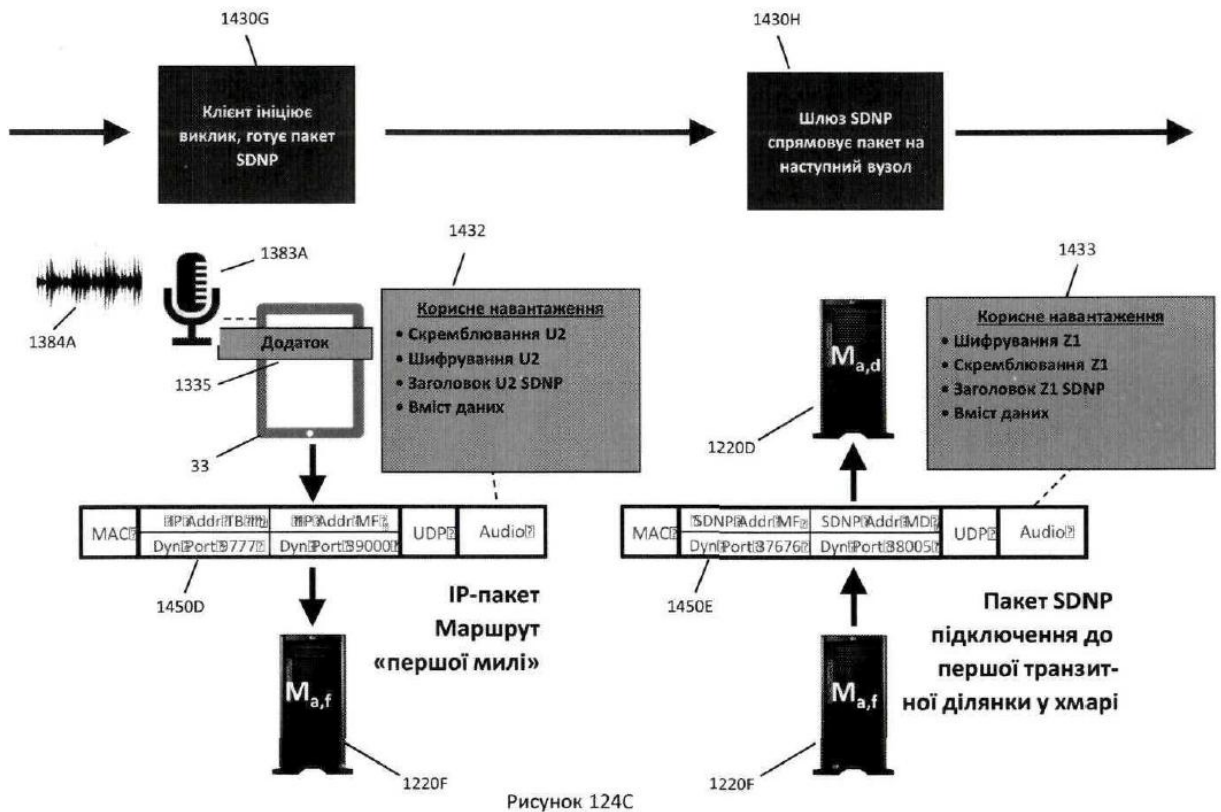


Рисунок 124В



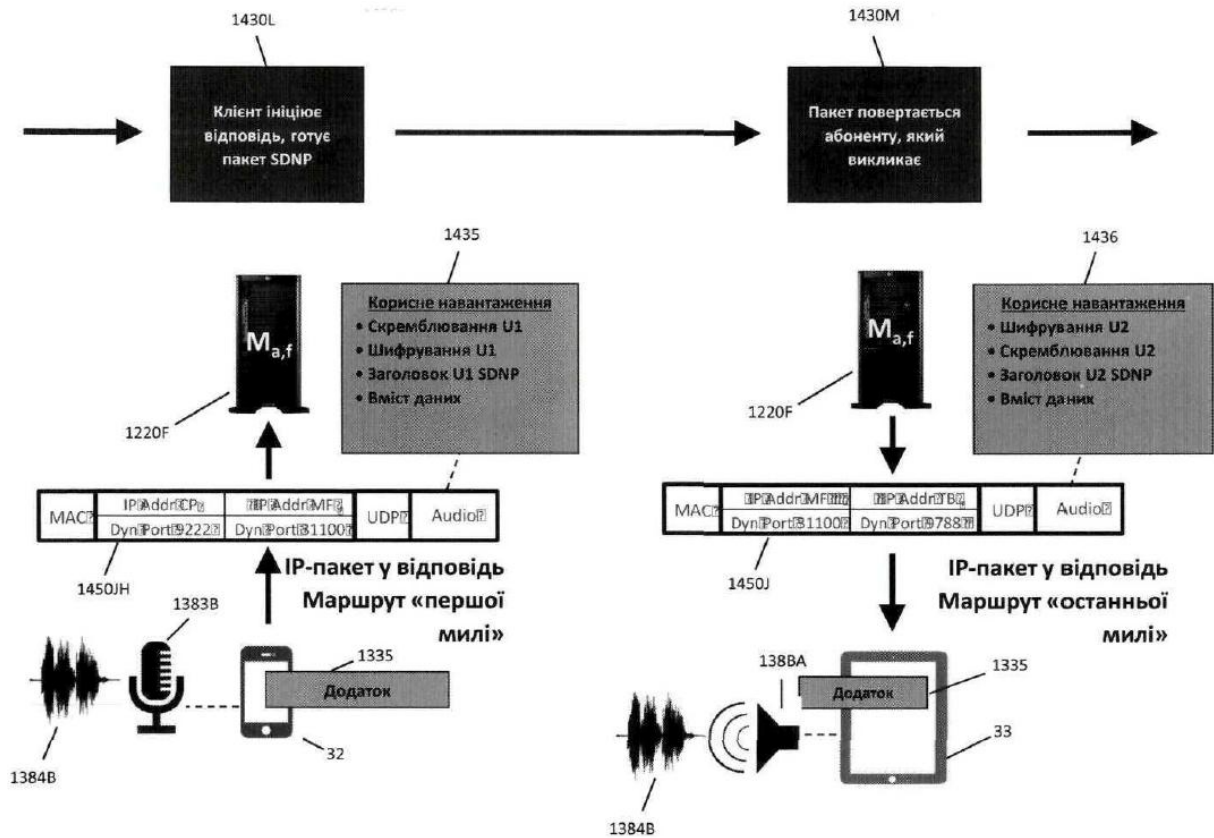


Рисунок 124E

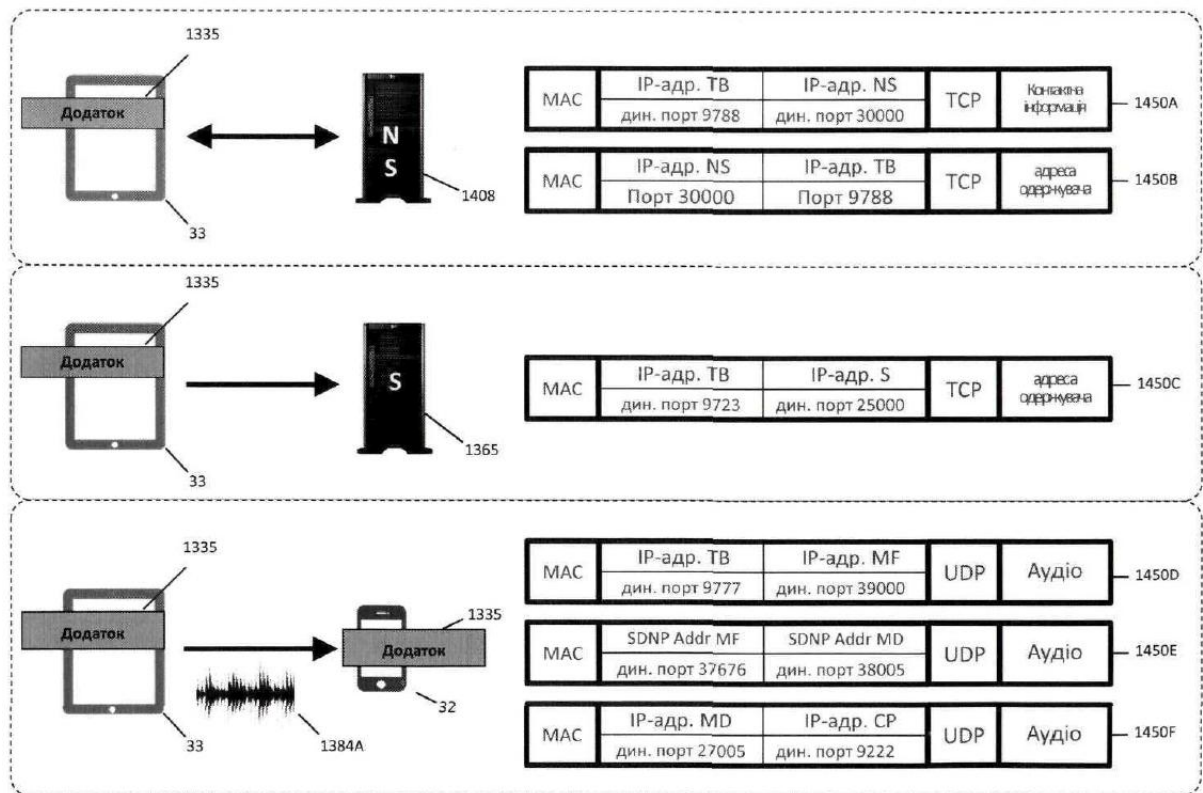
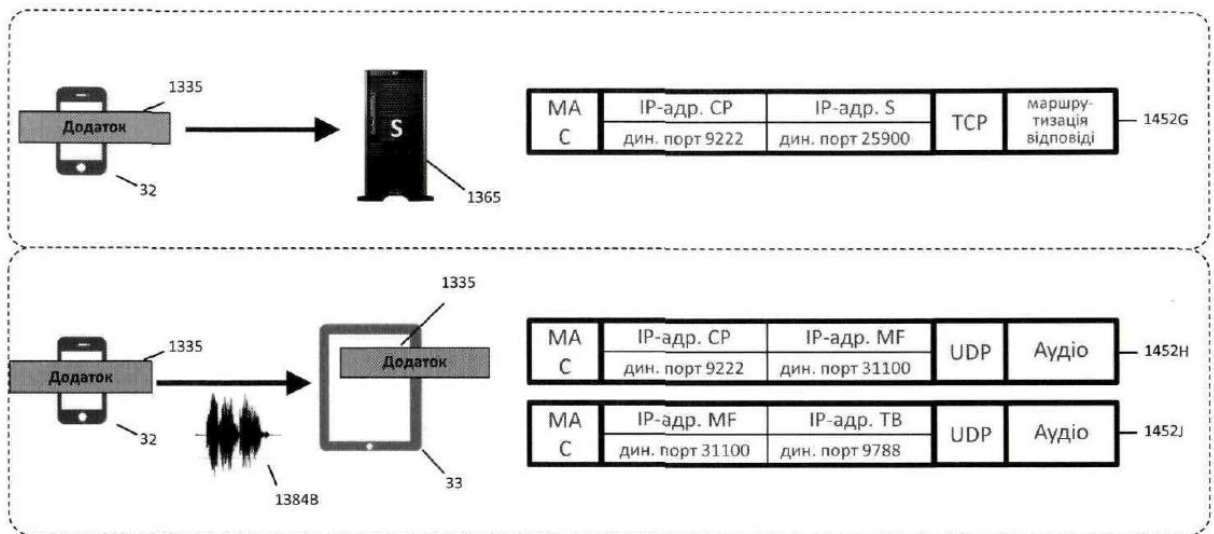
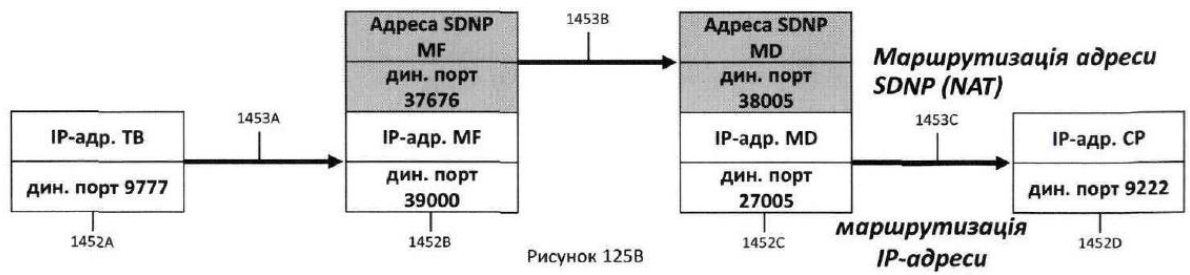
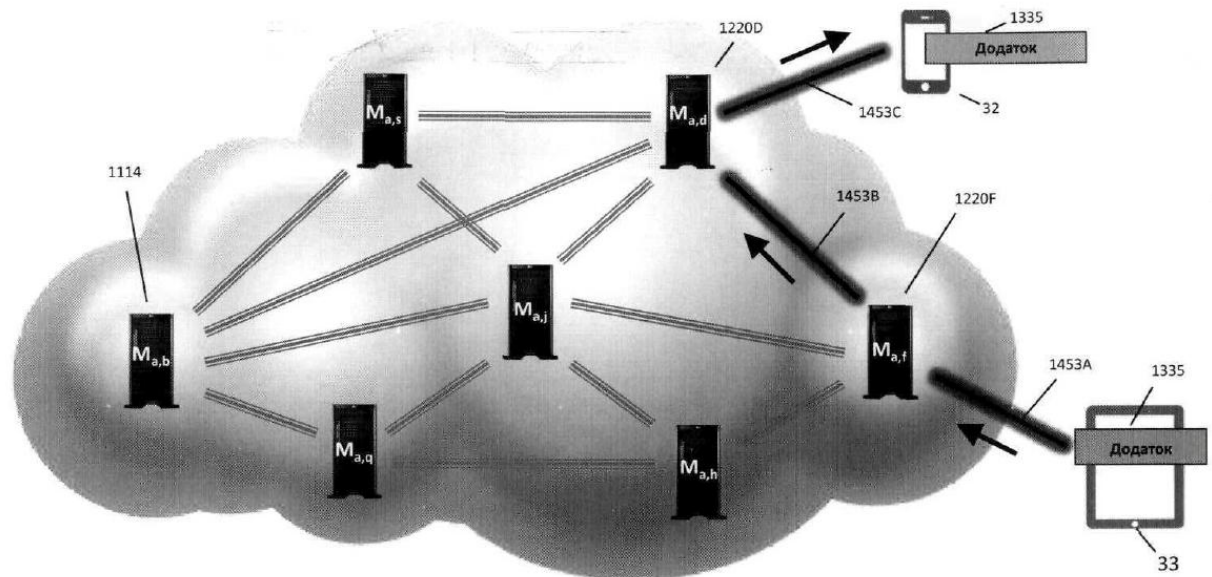
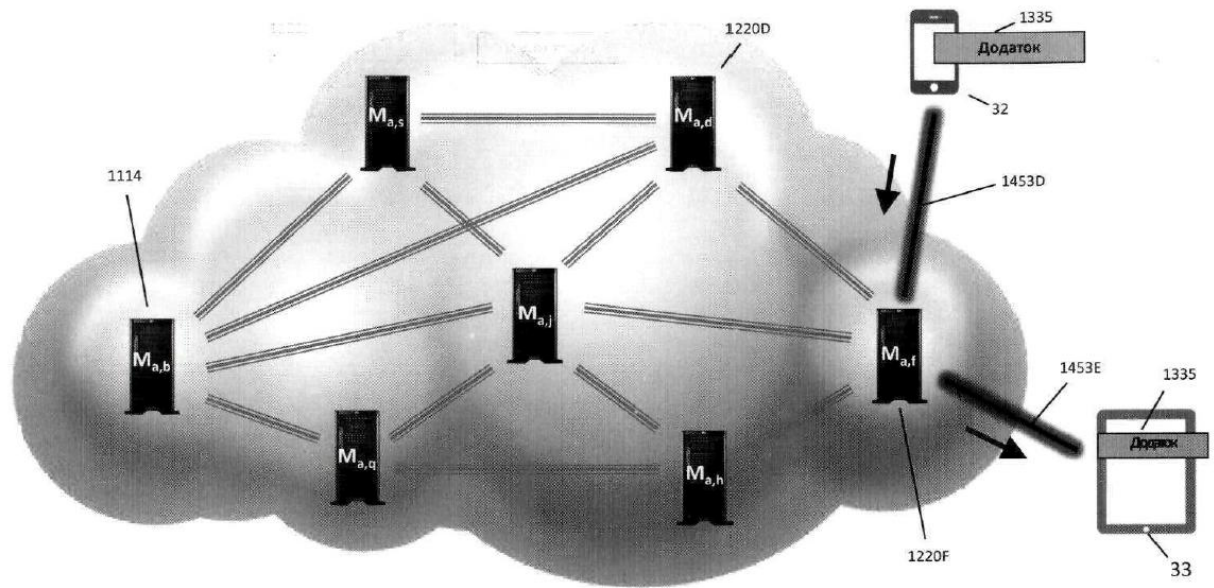


Рисунок 125A

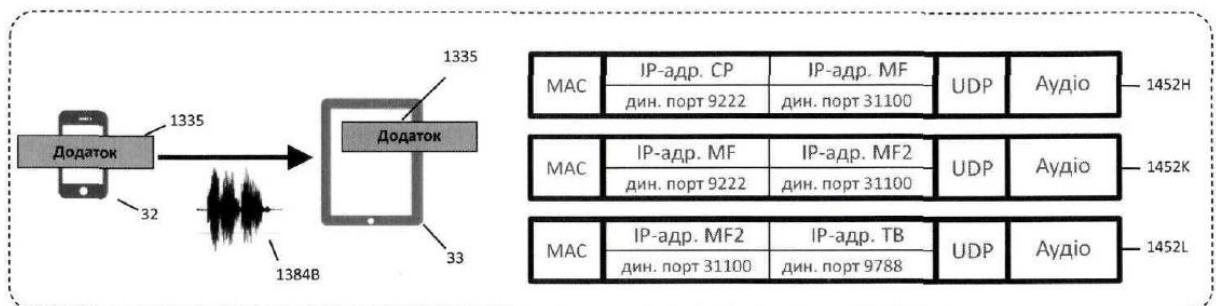




маршрутизація тільки IP-адреси



Рисунок 126В



маршрутизація трансльованої IP-адреси (з використанням резервного сервера)

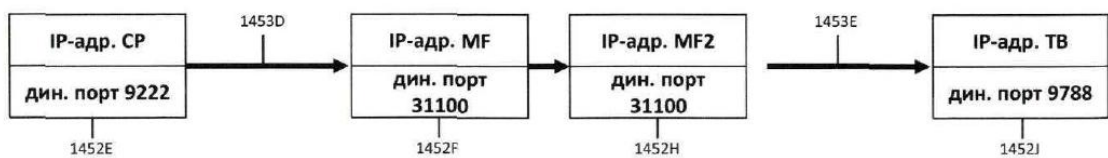
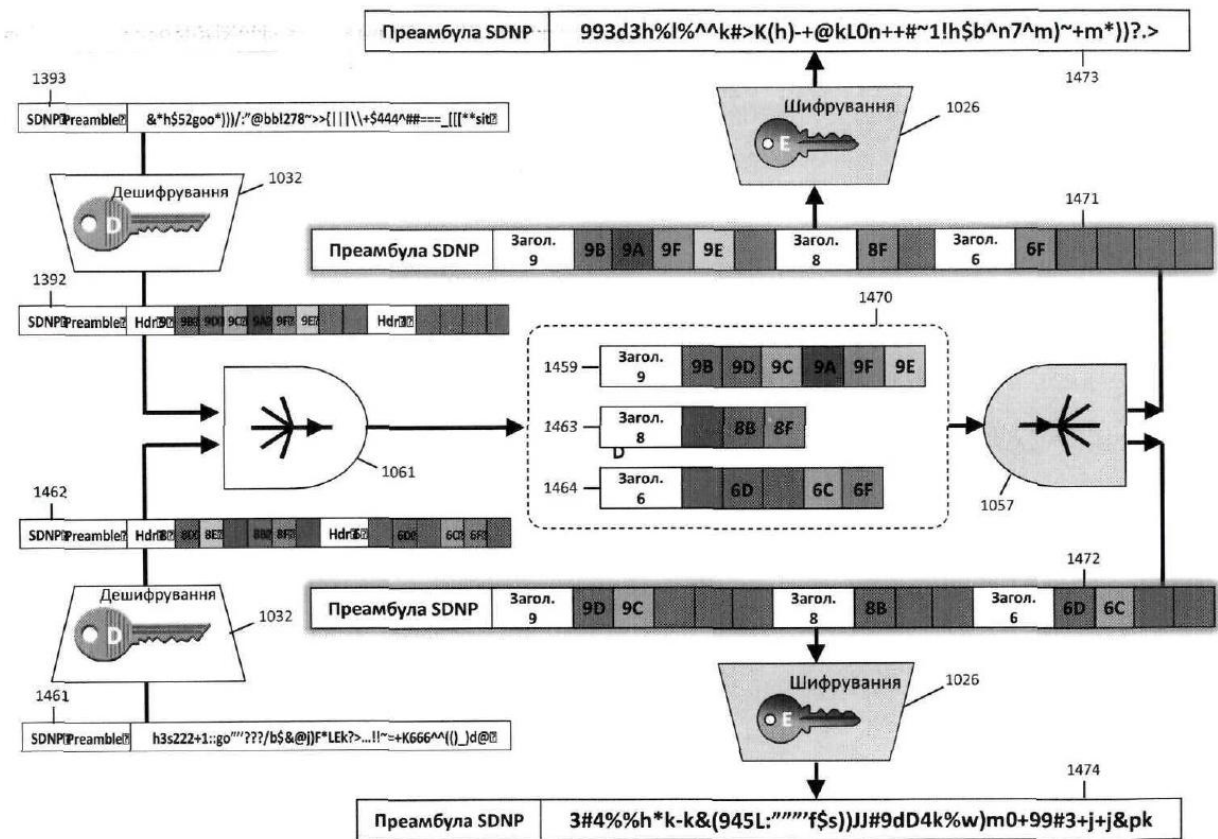
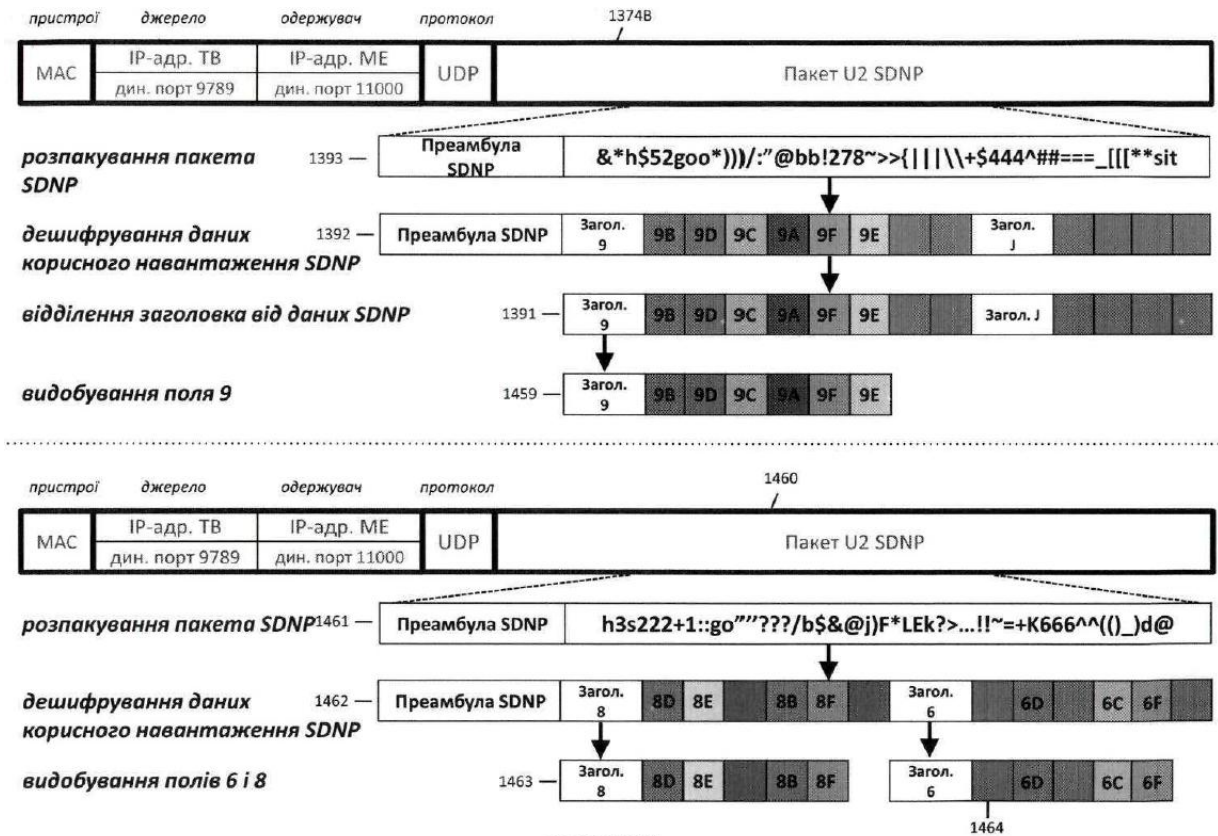


Рисунок 126С



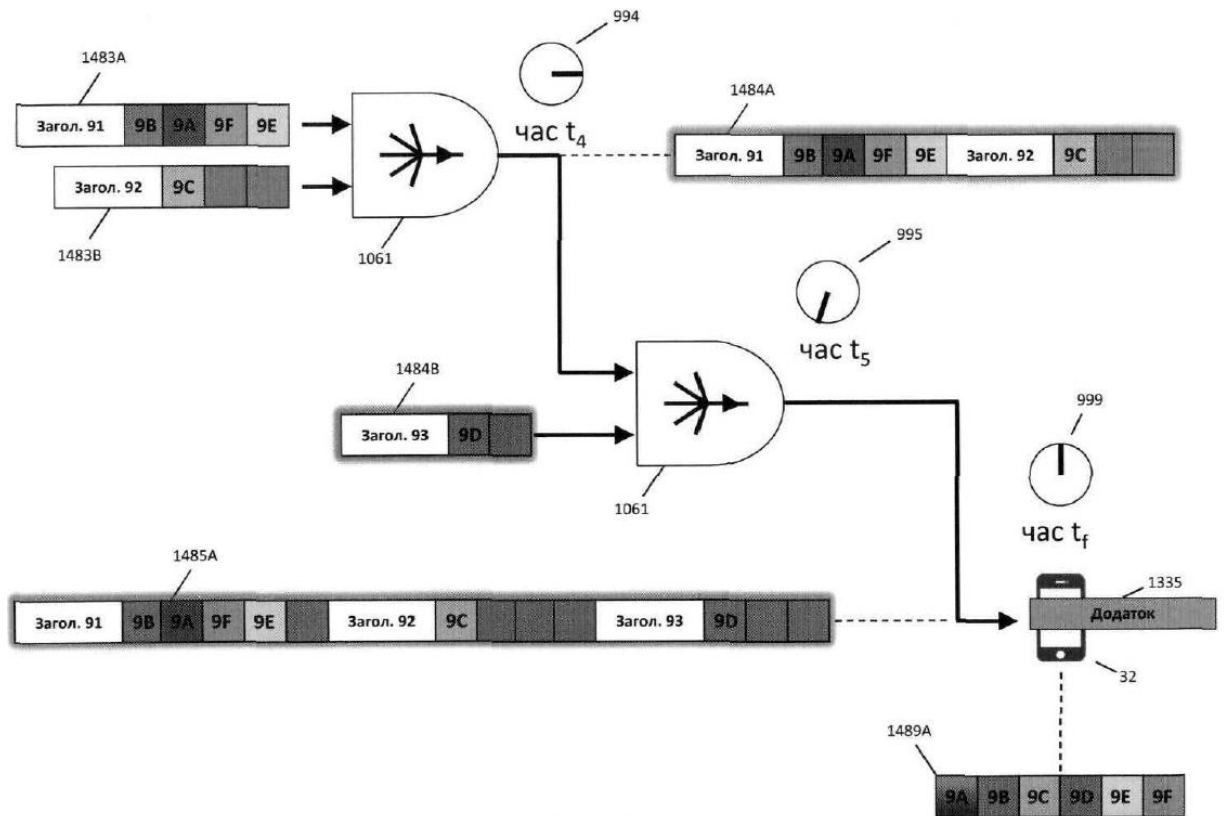


Рисунок 129А

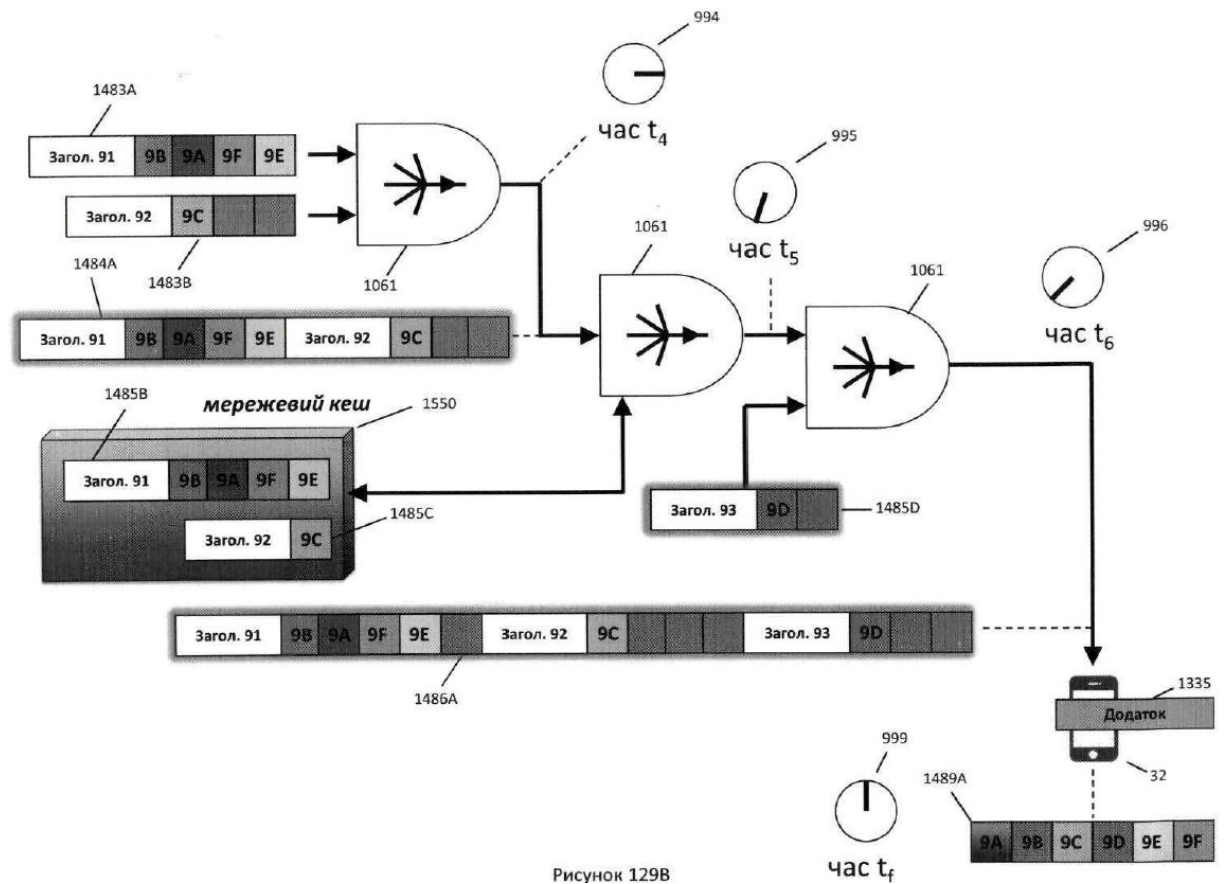
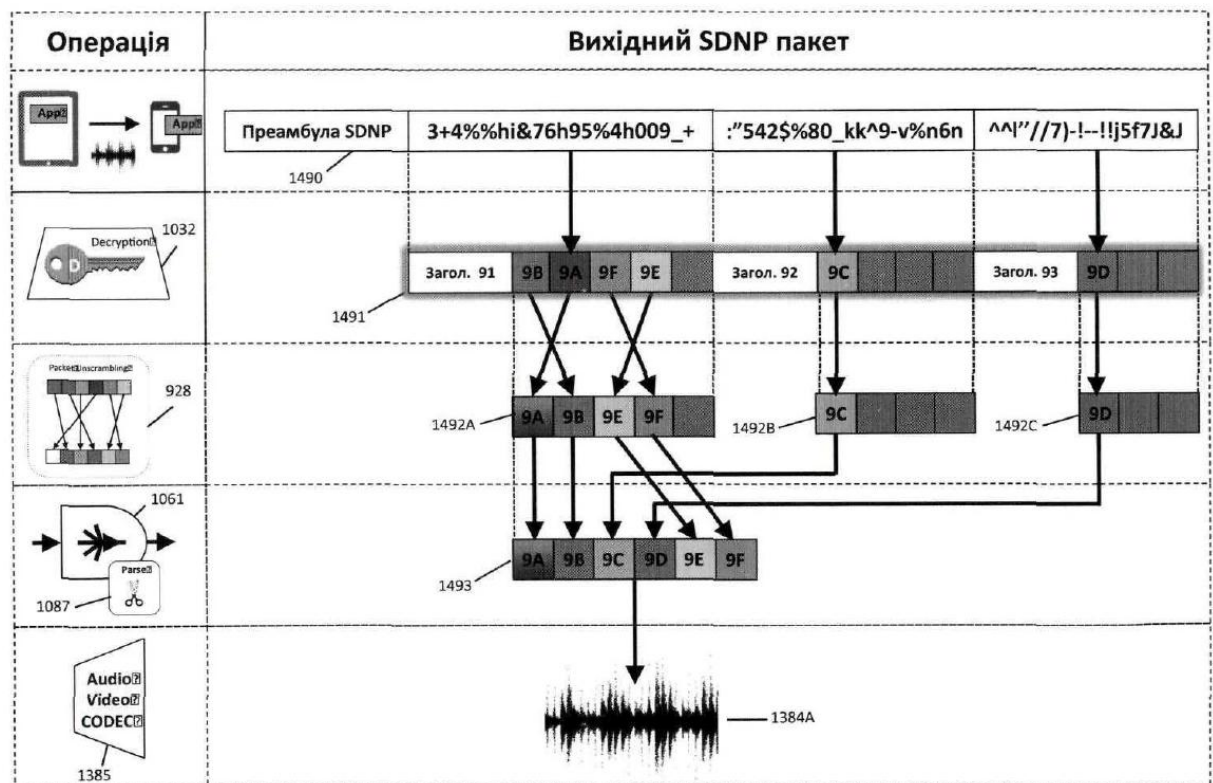
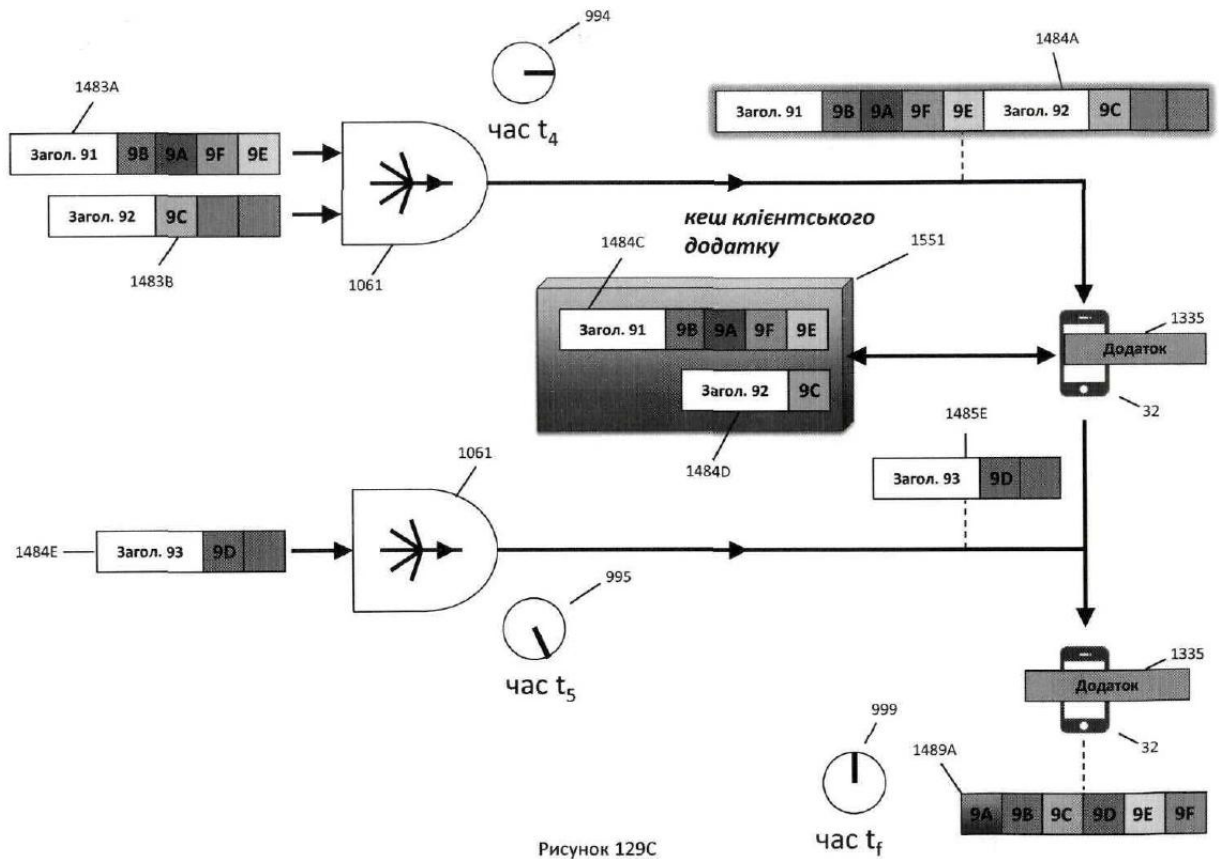
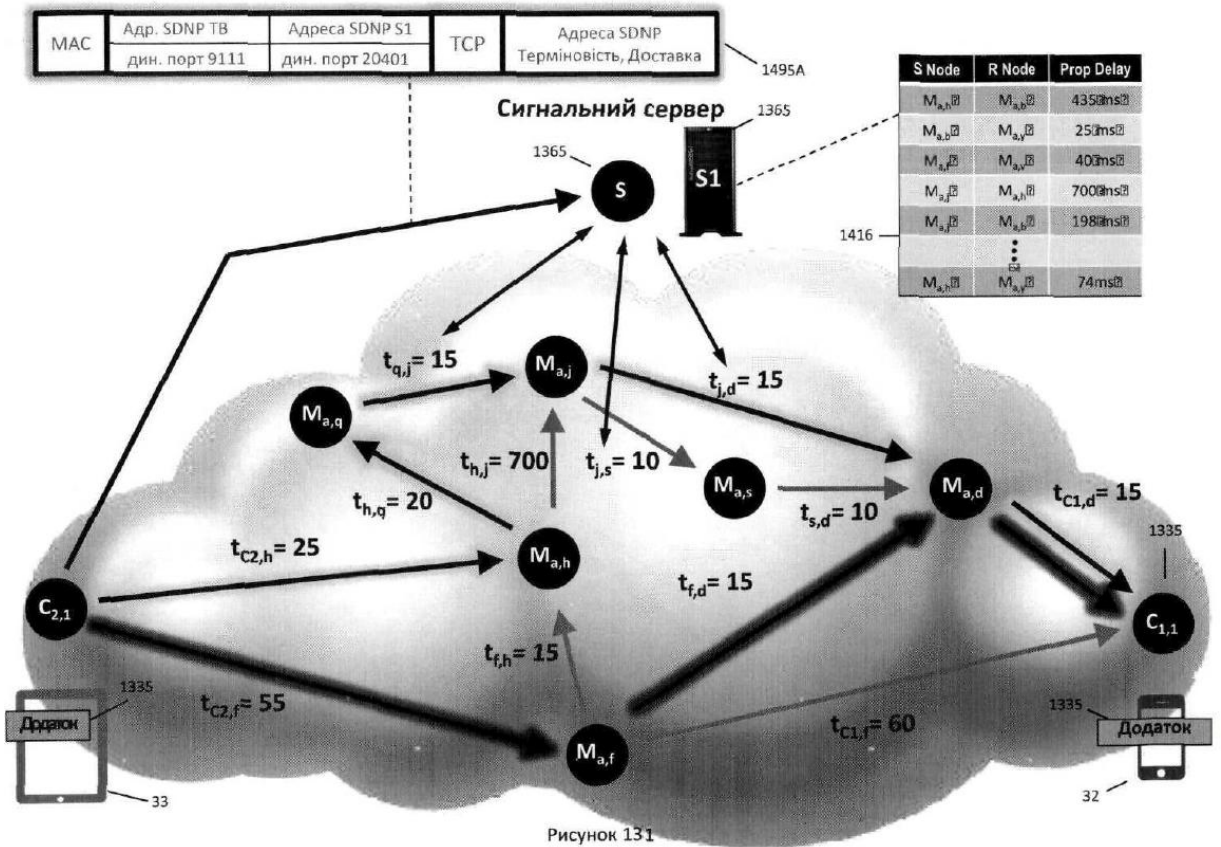
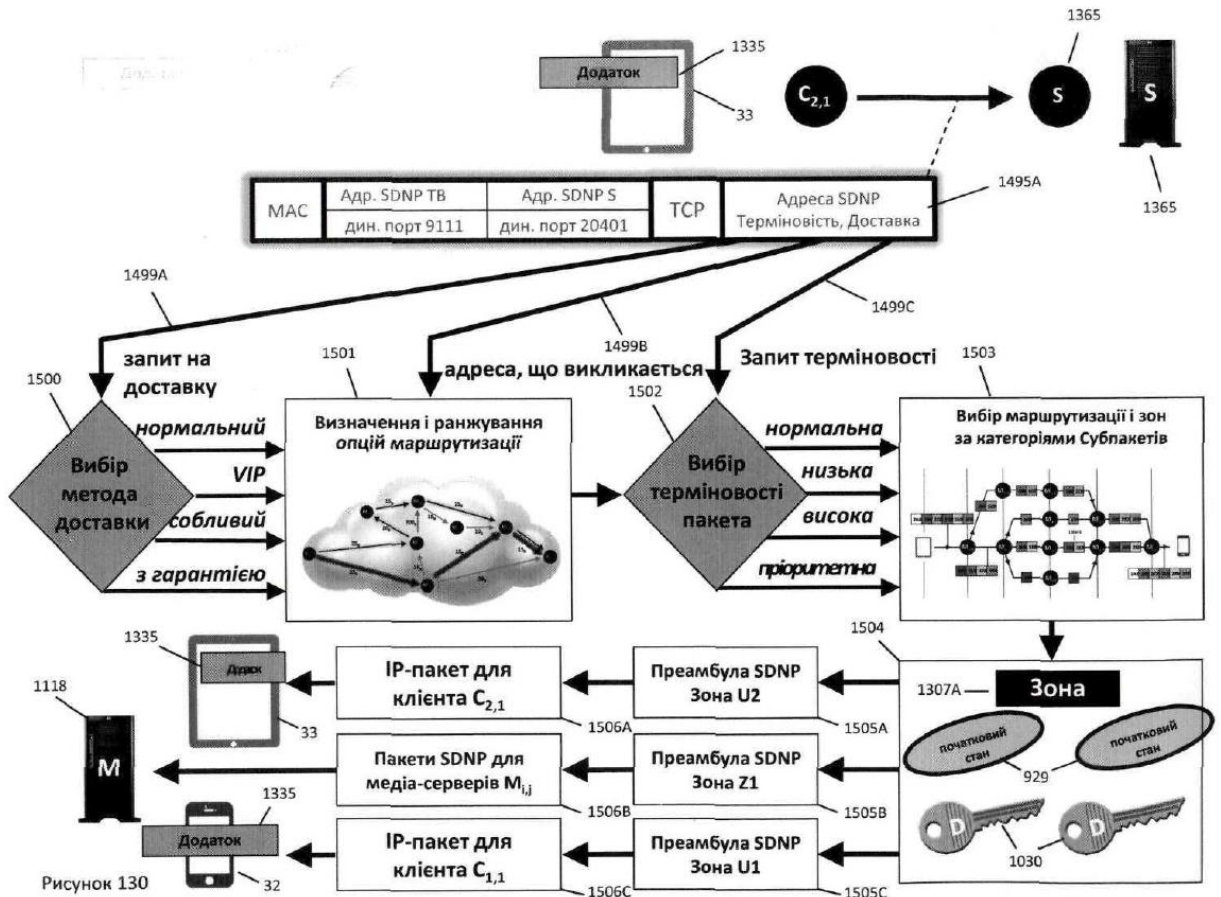


Рисунок 129В





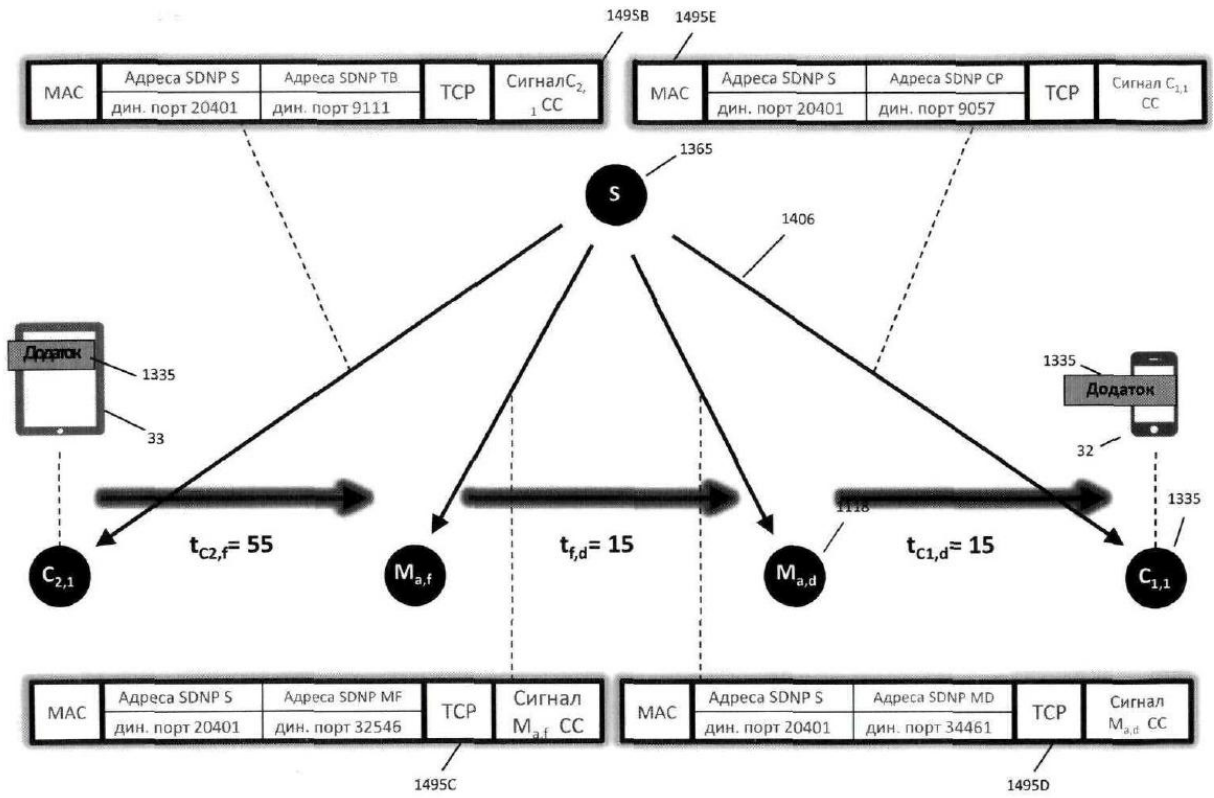


Рисунок 132А

Сигнальні пакети команд і керування (Маршрут 1-2)

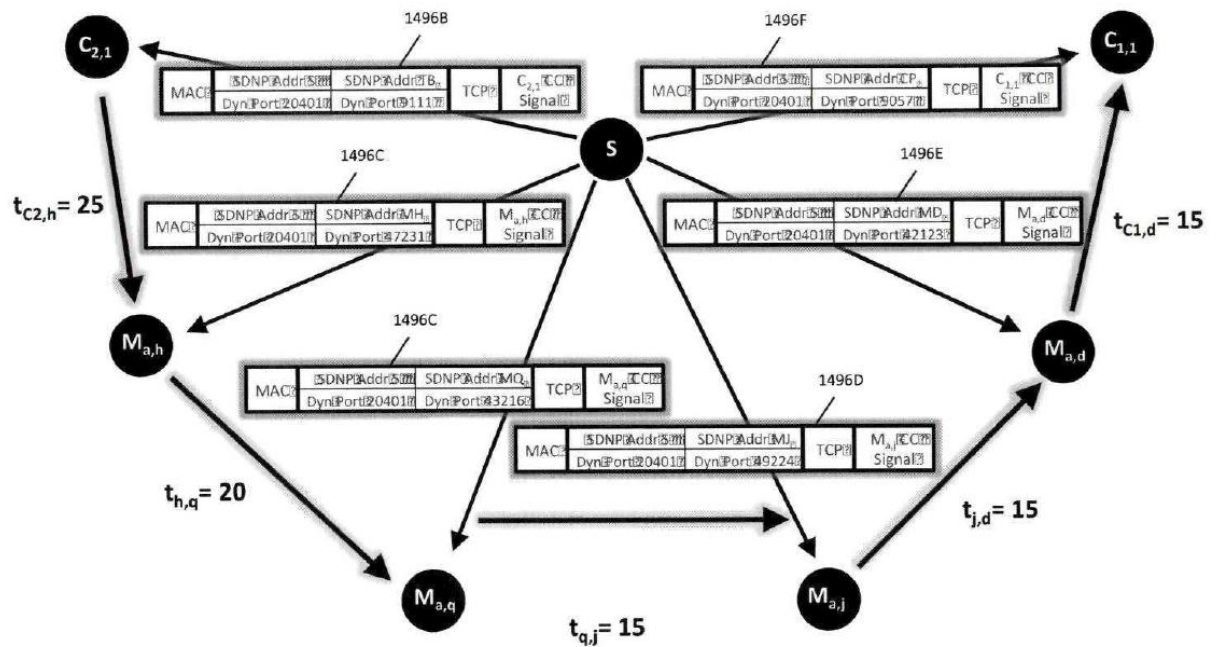


Рисунок 132В

Відновлення пакета

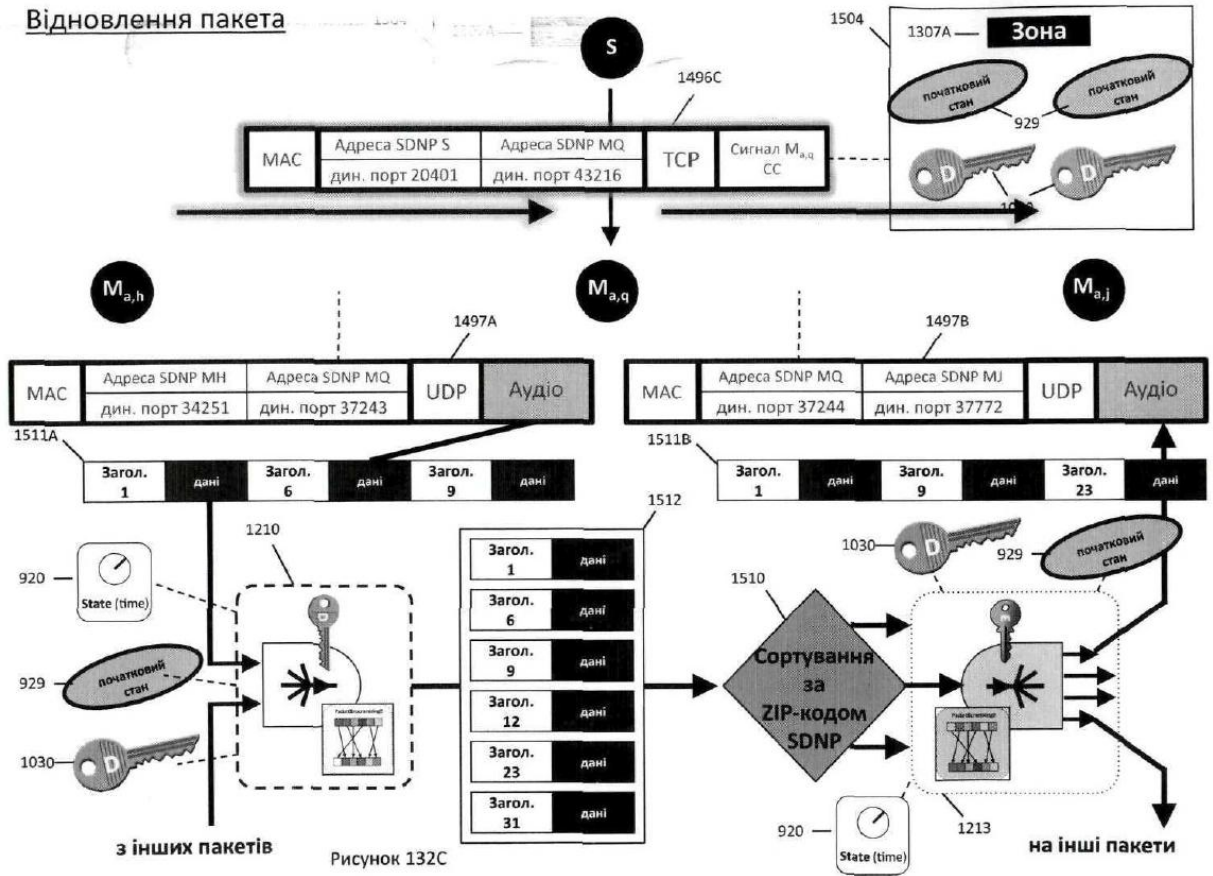


Рисунок 132C

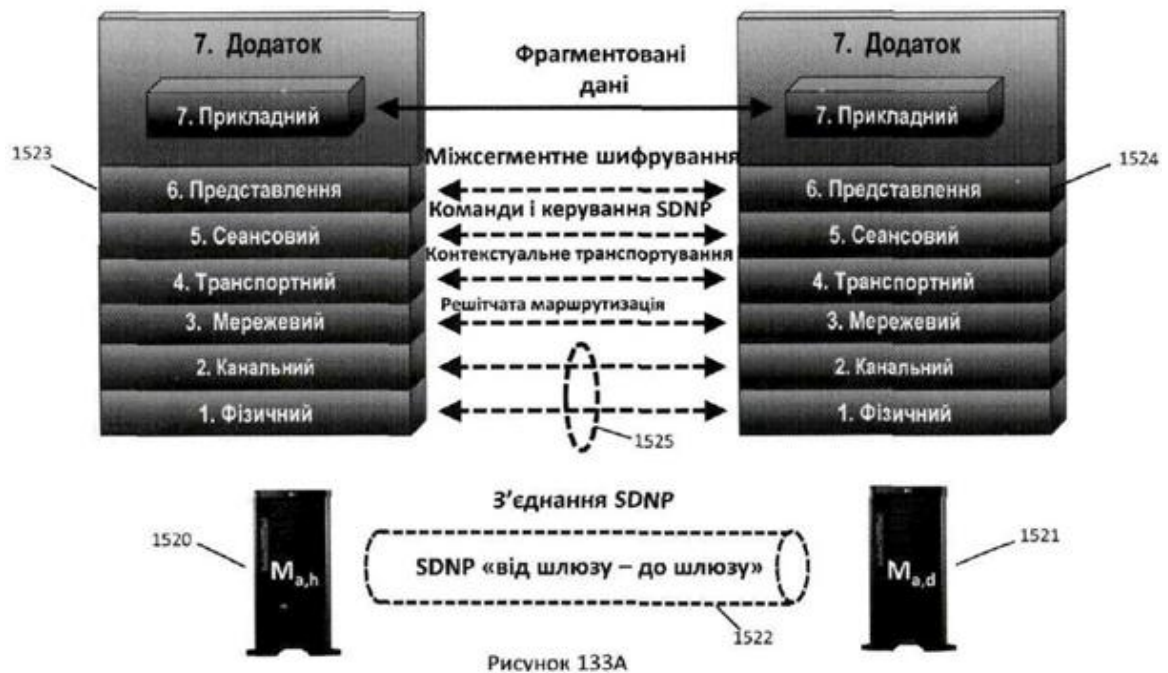


Рисунок 133A

Зашифрований тунельний сеанс SDNP

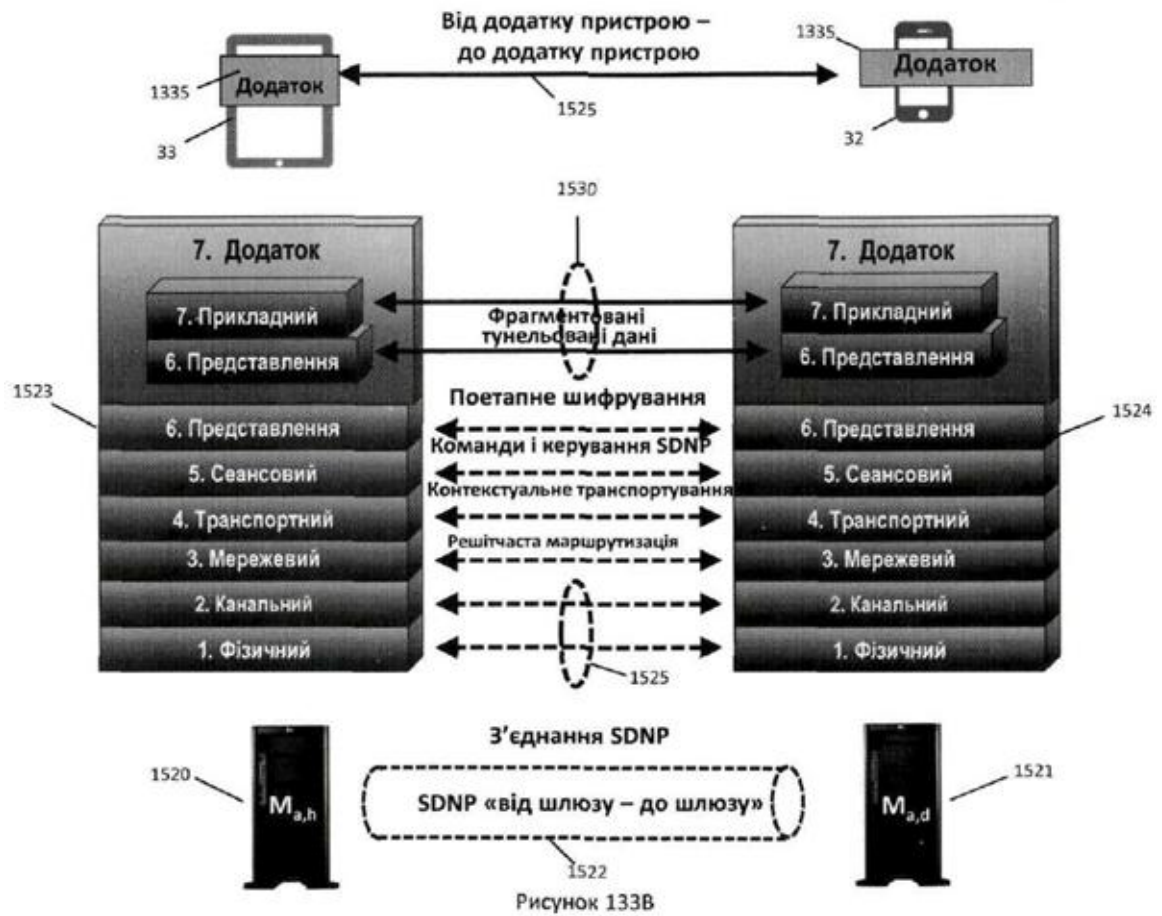


Рисунок 133В

Пересилання SDNP пакетів декількома маршрутами

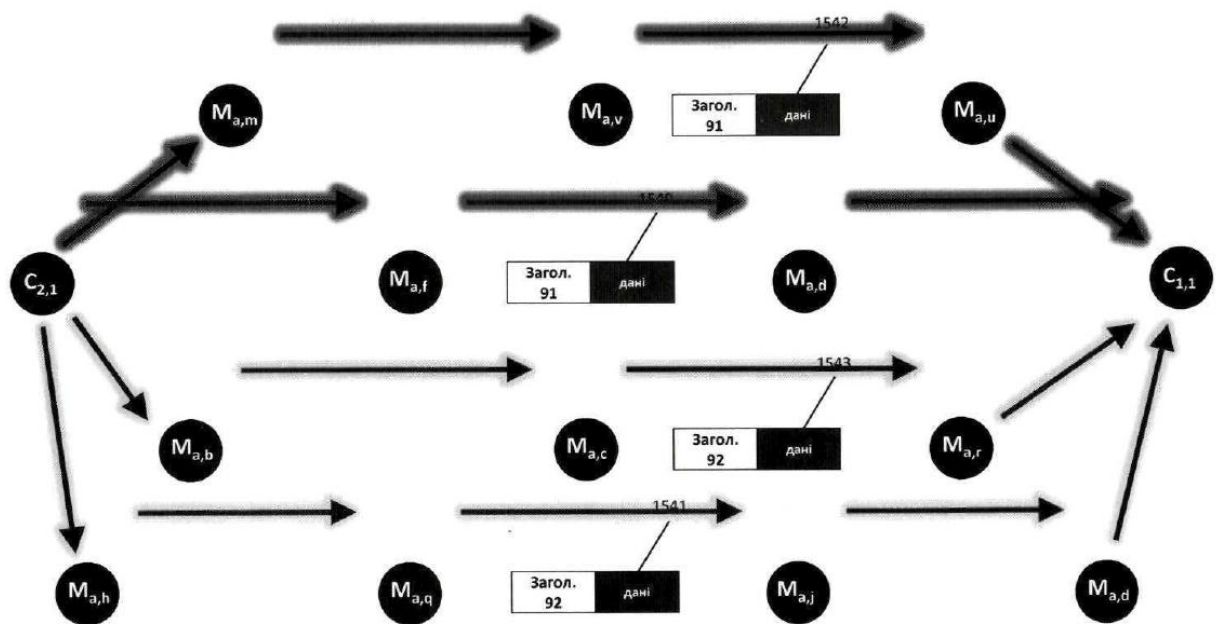


Рисунок 134

Параметр	OTT	VPN	PTP	SDNP
Адреса джерела	Видима	Зашифрована	Видима	Відсутня (NAT)
Адреса одержувача	Видима	Зашифрована	Видима	Відсутня (NAT)
Зашифроване корисне навантаження	Статичне (опціонально)	Статичне (опціонально)	Статичне (опціонально)	Динамічне
Скрембльоване корисне навантаження	Немає	Немає	Немає	Динамічне
Решітчаста маршрутизація	Немає	Немає	Немає	Динамічне
Вузли маршрутизації	Інтернет-сервери	Виділені сервери	Користувачі	Виділені сервери
Затримка	Неконтрольована	Єдиний маршрут	Мінімальне керування	Динамічна мінімальна

Рисунок 135