

**ПРОТОКОЛ ТУНЕЛЮВАННЯ ГОЛОСОВОГО НАВАНТАЖЕННЯ
З АГРЕГАЦІЄЮ СЕСІЙ ТА КОМПРЕСІЄЮ ЗАГОЛОВКІВ**

**ПРОТОКОЛ ТУННЕЛИРОВАНИЯ ГОЛОСОВОЙ НАГРУЗКИ
С АГРЕГАЦИЕЙ СЕССИЙ И КОМПРЕССИЕЙ ЗАГОЛОВКОВ**

**THE VOICE DATA TUNNELING WITH SESSIONS AGGREGATION
AND HEADERS COMPRESSION PROTOCOL**

Анотація. Розроблено специфікацію протоколу тунелювання голосового навантаження з агрегацією сесій та компресією заголовків. Розглянуто основні параметри роботи протоколу. Розроблені основні засади програмної реалізації протоколу та зазначено можливі шляхи оптимізації запатентованого авторами телекомунікаційного механізму.

Аннотация. Разработана спецификация протокола туннелирования голосовой нагрузки с агрегацией сессий и компрессией заголовков. Рассмотрены основные параметры работы протокола. Разработаны основные положения программной реализации протокола и указаны возможные пути оптимизации запатентованного авторами телекоммуникационного механизма.

Summary. Voice data tunneling with sessions aggregation and headers compression protocol are specified. The main parameters of the protocol. The main provisions of the protocol and software implementation are possible ways of optimizing the telecommunications mechanism patented by the authors.

В основу процесу еволюції телекомунікаційних мереж, як правило, покладено принцип економічної доцільності. Бажання максимально ефективно використати вкладені в будівництво мережі інвестиції змушує власників мереж постійно шукати шляхи більш оптимального використання наявних каналних ресурсів за умов дотримання необхідного рівня якості обслуговування. Внаслідок стрімкого розвитку IP-мереж в останні роки одним із найбільш популярних методів такої оптимізації стало застосування технологій, які дозволяють мінімізувати обсяги передавання службової інформації і тим самим зменшити розмір протокольної надлишковості. Особливої гостроти це питання набуло після значного розповсюдження технологій передавання мультимедійної (насамперед, голосової) інформації через IP-мережі під час якого частка службового навантаження може перевищувати частку корисного навантаження в декілька разів [1–2].

В якості одного із варіантів технології мінімізації службового навантаження при передаванні голосової інформації через IP-мережу авторами запропоновано спосіб [1,3,4], який базується на принципах збереження інформації про чинні контексти RTP-сесій із одночасним застосуванням процедури агрегації пакетів з паралельних RTP-сесій. У пропонуваному способі ефект мінімізації, насамперед, досягається за рахунок внесення додаткової затримки до частини RTP-сесій через проміжну буферизацію голосових фреймів на компресорі (вхідній стороні IP-тунелю) з подальшим пересиланням агрегованих пакетів через IP-мережу.

Проте невід'ємною частиною практичної реалізації будь-якого механізму обміну інформацією є специфікація алгоритмів його роботи та характеристик протоколів зв'язку. Отже, зрозуміло, що впровадження запропонованого способу потребує розробки відповідного протоколу та програмних модулів, що реалізують функціональність компресора та декомпресора тунелю [1].

Метою статті є розробка протоколу тунелювання голосового навантаження з агрегацією сесій та компресією заголовків.

Протокол тунелювання голосового навантаження з агрегацією сесій та компресією заголовків являє собою набір правил, що регламентують формат та процедури обміну інформацією між компресором та декомпресором тунелю й характеризується своїми логічними та процедурними характеристиками.

1. Логічні характеристики протоколу. Основною логічною характеристикою протоколу є формат агрегованих пакетів, що містять голосові фрейми або пакети, що ініціюють або поновлюють стан контексту. На рис. 1 зображено схему інкапсуляції, а також структуру службових заголовків

агрегованого пакета, що використовується тунелем для передавання інформації від компресора до декомпресора та в зворотному напрямку.

Як видно з рис. 1 агрегований пакет складається зі звичайного IP-заголовка (в якому адресою відправника є IP-адреса однієї сторони тунелю – компресора або декомпресора, а адресою отримується є IP-адреса іншої сторони – декомпресора або компресора), а також із серії мінізаголовків за кожним з яких може слідувати голосовий фрейм (корисні дані) або повний IP-пакет (корисні дані та службова інформація). Розмір мінізаголовків, відповідно до запропонованої специфікації, може становити 2 (пошкодження контексту та скорочений варіант), 5 (розширений варіант) або 8 (повний варіант) байтів. Повідомлення про пошкодження контексту передається виключно від декомпресора до компресора з метою інформування про необхідність термінового пересилання повного IP-пакета з метою відновлення інформації про чинний контекст RTP-сесії. Скорочений варіант мінізаголовка використовується у випадку, коли слідом за ним передається повний IP-пакет та є сигналом для декомпресора про створення або відновлення контексту. У свою чергу розширений варіант мінізаголовка формується компресором у випадку передавання лише корисних даних (голосового фрейма) з чергового пакета в сесії, а повний варіант використовується для періодичного поновлення стану змінних полів RTP-заголовка.

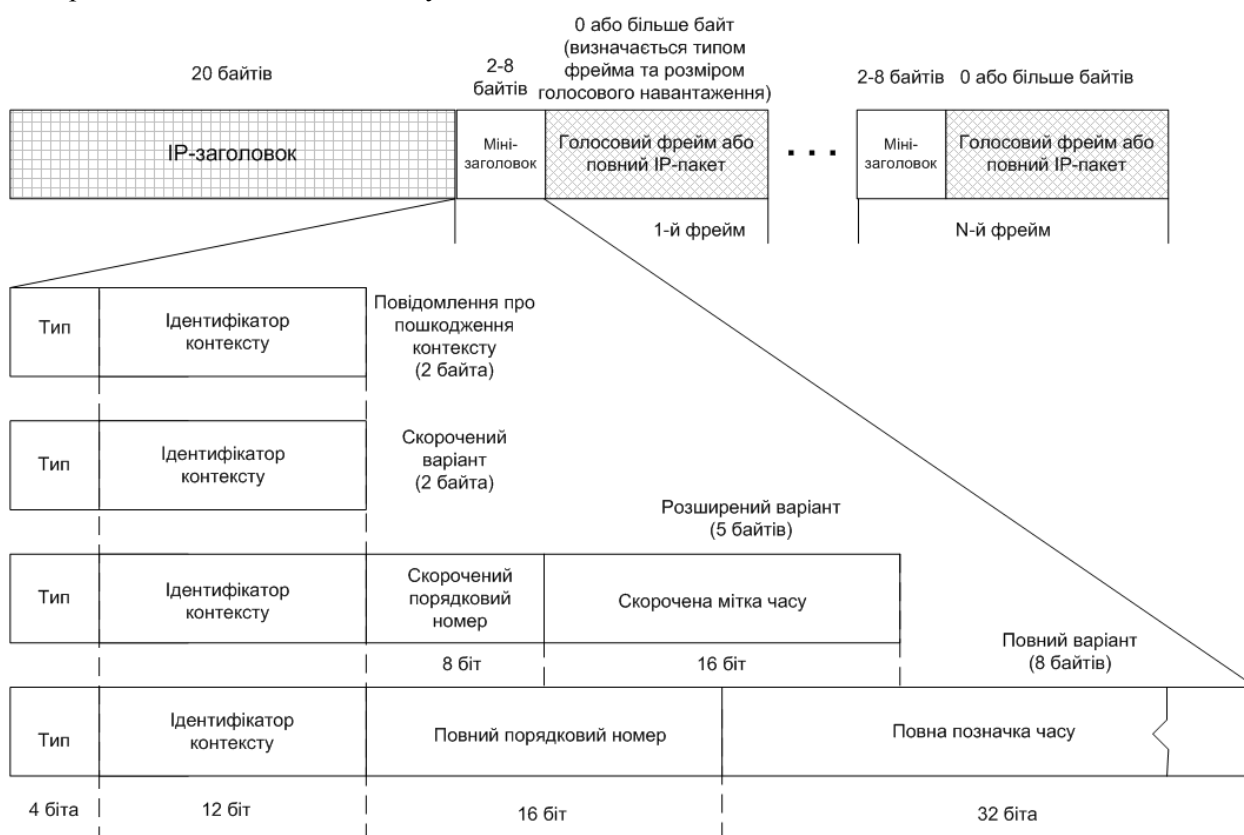


Рисунок 1 – Схема інкапсуляції та структура службових заголовків

Структурно мінізаголовок складається з двох (для повідомлення про пошкодження контексту та для скороченого варіанта) або чотирьох (для розширеного та повного варіантів) полів:

– «Тип». Поле розміром 4 біта визначає тип мінізаголовка та, як наслідок, тип даних, що слідує за ним. Перший біт цього поля завжди має дорівнювати «1», що дозволить відрізнити спеціалізований контент від класичних IP-пакетів 4 або 6-ї версій (відповідні значення поля «Версія» для цих протоколів починаються зі значення «0»). Таким чином, для ідентифікації типів мінізаголовка залишається 3 біта, два з яких позначають безпосередньо тип заголовка, а ще один може використовуватись для розширення змісту цих типів у майбутньому;

– «Ідентифікатор контексту». Поле розміром 12 бітів, використовується компресором та декомпресором для ідентифікації контексту RTP-сесії в межах одного тунелю. Такий розмір поля дозволяє ідентифікувати одночасно до 4096 контекстів та практично виключити колізії при створенні нових записів у таблиці контекстів;

– «Скорочений порядковий номер». Поле розміром 8 бітів, використовується лише в розширеному варіанті мінізаголовка для передавання чергового номера фрейма в межах сесії та застосовується декомпресором для визначення абсолютного значення порядкового номера (у термінах протоколу RTP) шляхом додавання до останнього актуального значення повного порядкового номера, яке було отримано декомпресором у складі повного IP-пакета. Слід зазначити, що такий розмір поля дозволяє передавати до 255 голосових фреймів без поновлення контексту (передавання повного IP-пакета);

– «Скорочена позначка часу». Поле розміром 16 бітів, яке також використовується лише в розширеному варіанті мінізаголовка для передавання зміщення позначки часу відносно останнього актуального значення повної позначки часу, яке було отримано декомпресором у складі повного IP-пакета. Як і відповідне поле RTP-заголовка [5] «Скорочена позначка часу» призначена для подолання проблем із варіацією затримки (джиттером) у межах однієї голосової сесії;

– «Повний порядковий номер» та «Повна позначка часу». Поля розміром 16 та 32 біти відповідно, які за призначенням повністю відповідають полям «Порядковий номер» та «Позначка часу» RTP-заголовка [5]. Періодичне передавання цієї інформації дозволяє синхронізувати відповідні значення контексту на компресорі та декомпресорі. Слід зазначити, що періодичність з якою має здійснюватись часткове поновлення контексту має визначитись окремим алгоритмом, наприклад, на підставі аналізу кількості пошкоджень контексту за певний період часу.

2. Процедурні характеристики протоколу. Основними процедурними характеристиками пропонованого протоколу є:

– процедура оброблення компресором голосових пакетів та процедура оброблення декомпресором агрегованих пакетів. Ці процедури детально розглянуто в [1, 4];

– процедура забезпечення гарантованої якості обслуговування. Рекомендації щодо реалізації модуля контролю за забезпеченням гарантованої якості обслуговування при використанні тунелю надано в [3, 6];

– процедури перетворення повних полів RTP-заголовка до скорочених варіантів.

Розглянемо більш детально останню з наведених вище характеристик. Як показано на рис. 1 мінізаголовок у розширеному варіанті містить три основні поля, процедури генерації яких не є очевидними: «ID контексту», «Скорочений порядковий номер» та «Скорочена позначка часу». При цьому якщо процедура генерації ідентифікатора контексту є досить тривіальною та може являти собою вибір першого незайнятого ідентифікатора в діапазоні від 0 (бітове значення «0000») до 31-го (бітове значення «1111»), то процедури перетворення значень полів RTP-заголовка до скорочених варіантів є дещо складнішими.

На рис. 2 наведено алгоритми формування відповідних полів мінізаголовка із визначенням способу передавання голосової інформації від компресора до декомпресора (у скороченому вигляді, або із передаванням повного IP-пакета).

На початку роботи алгоритму (при обробленні кожного голосового фрейма) здійснюється пошук контексту RTP-сесії в таблиці контекстів. У разі, якщо контекст не знайдено (рис. 2) здійснюється створення нового контексту та збереження опорних значень порядкового номера та позначки часу (з відповідних полів RTP-заголовка) в рамках створеного контексту. У подальшому до агрегованого пакета вноситься повний IP-пакет разом зі скороченим варіантом мінізаголовка, що дозволяє створити або поновити пошкоджений контекст на стороні декомпресора і тим самим проінформувати іншу сторону тунелю про зміну опорних значень полів RTP-заголовка.

У випадку, коли контекст знайдено здійснюється обчислення зміщення чинного значення полів «Порядковий номер» та «Позначка часу» RTP-заголовка по відношенню до опорних значень збережених під час створення або останнього оновлення контексту. Якщо отриманий результат перевищує максимально припустиме значення хоча б для одного з полів (2^8-1 для «Порядкового номера» та $2^{16}-1$ для «Позначки часу») здійснюється оновлення існуючого контексту та збереження нових опорних значень. У подальшому до агрегованого пакета вноситься голосовий фрейм з повним варіантом мінізаголовка, який містить повні значення зазначених вище полів.

У випадку, коли зміщення значення полів «Порядковий номер» та «Позначка часу» не перевищило максимально припустиме значення до агрегованого пакета вноситься голосовий фрейм із розширеним варіантом мінізаголовка. До речі максимальні значення зміщення визначаються розміром відповідних полів саме цього варіанта мінізаголовка.

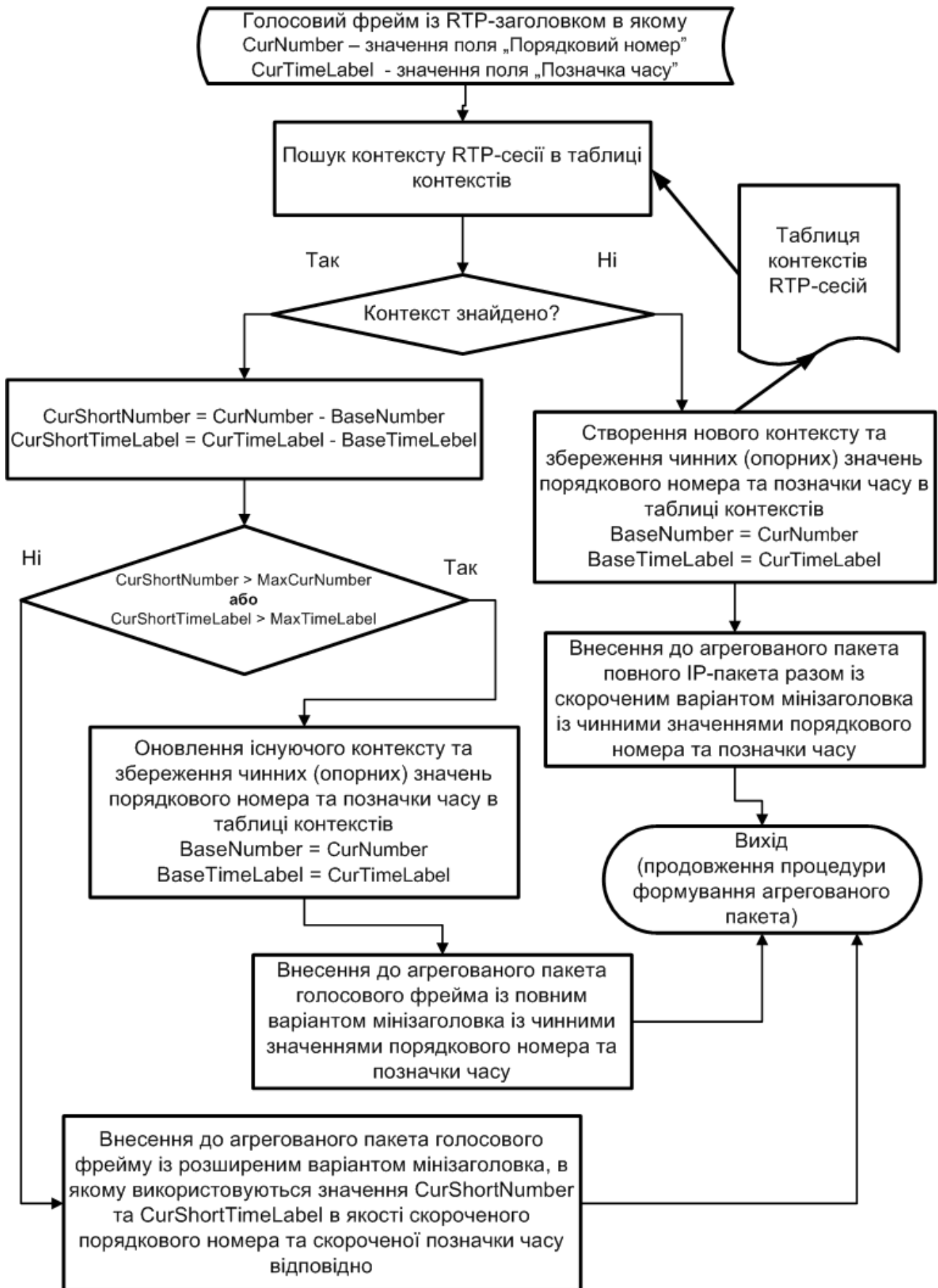


Рисунок 2 – Алгоритми формування полів «Скорочений порядковий номер» та «Скорочена позначка часу» мінізаголовка

По завершенню процедури здійснюється перехід до основного алгоритму роботи компресора, який детально розглянуто в [1].

Таким чином, розроблений алгоритм дозволяє значно скоротити обсяг службової інформації, що передається із кожним голосовим фреймом (в тому числі, за рахунок використання скорочених версій змінних полів RTP-заголовка) при цьому не порушуючи принципи роботи протоколу RTP, яким передбачено механізми розміщення голосових фреймів у межах мультимедійного потоку в тому самому порядку, в якому вони надходять від голосового кодека.

3. Програмна реалізація. Протокол тунелювання голосового навантаження з агрегацією сесій та компресією заголовків реалізовано у вигляді модуля ядра операційної системи Linux із використанням середовища xfrm. Модуль створює тунельний інтерфейс, який можна конфігурувати стандартними засобами операційної системи.

Для реєстрації та видалення тунелю з чинного контексту операційної системи використовуються функції `xfrm4_tunnel_register` та `xfrm4_tunnel_deregister` відповідно. Функції обробки пакетів та допоміжні процедури додаються до адресного простору ядра.

На рис. 3 наведено діаграми проходження пакета через мережний стек операційної системи.

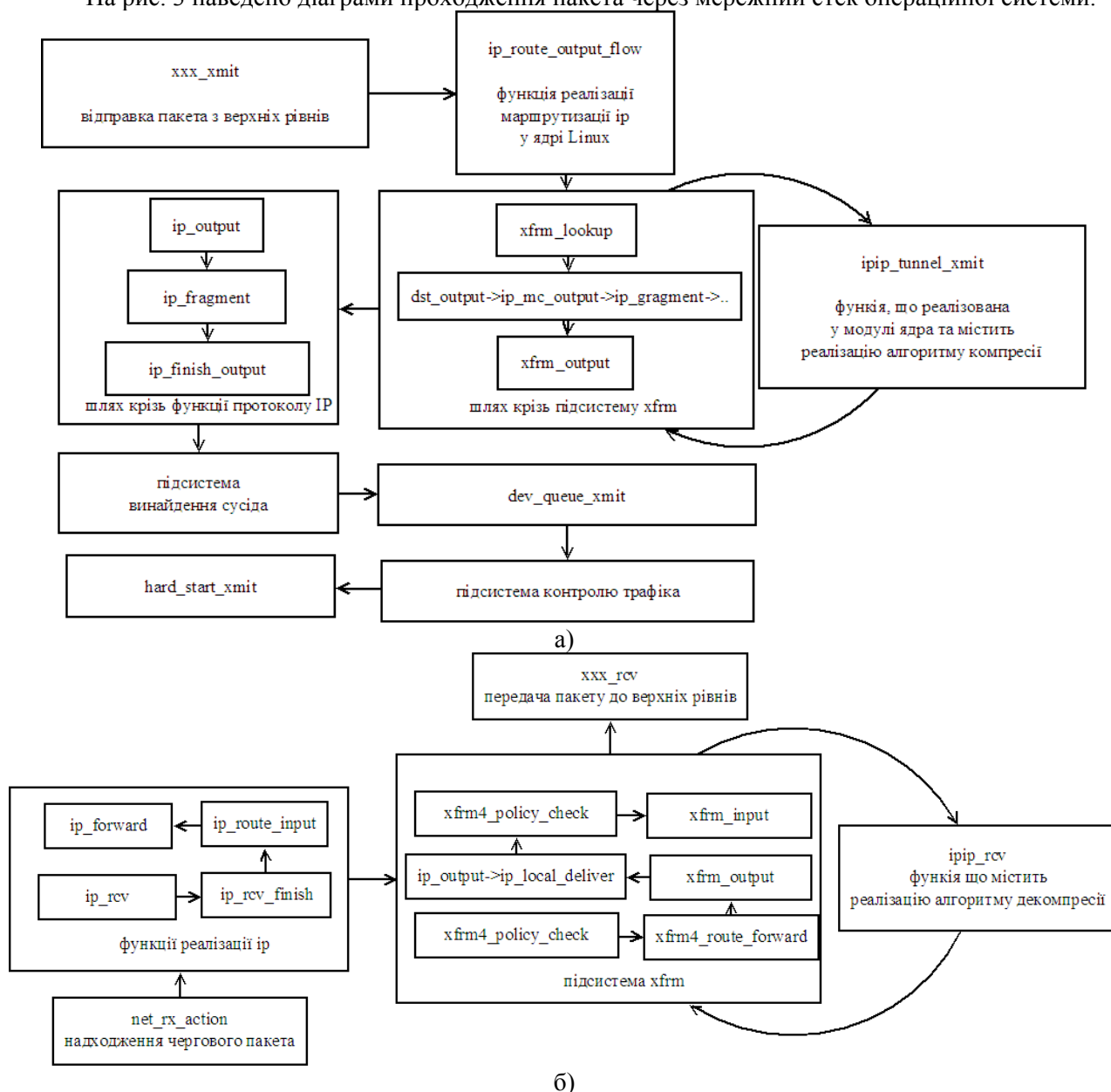


Рисунок 3 – Діаграми проходження пакета через мережний стек операційної системи:

а) проходження пакетів від прикладних процесів до тунелю;

б) надходження пакетів до верхніх рівнів стеку протоколів

Алгоритми компресора та декомпресора реалізовано у функціях `rip_tunnel_xmit` та `rip_rcv` відповідно (рис. 3). Таблиця контекстів має вигляд зв'язаного списку та містить у собі записи типу `ctx_entry`. Кожний запис типу `ctx_entry` містить ідентифікатор контексту та поля для збереження службової інформації заголовків протоколів IP, UDP та RTP.

Допоміжні функції керування контекстами, збереженням до контексту та відтворенням з контексту наведені в табл. 1.

Таблиця 1 – Допоміжні функції керування контекстами

Визначення функції	Призначення
<code>static void save_iur_to_ctxe(struct ctx_entry *ctxe, const struct iurhdr *iur)</code>	Зберігає службову інформацію протоколів IP,UDP,RTP до контексту
<code>static void restore_iur_from_ctxe(struct iurhdr *iur, const struct ctx_entry *ctxe, __be16 seq, __be32 ts)</code>	Відтворює заголовки протоколів IP,UDP,RTP з таблиці контекстів
<code>static inline struct ctx_entry *get_ctxe_by_ctxid(ctxid_t ctxid)</code>	Повертає структуру <code>ctx_entry</code> з таблиці контекстів за ідентифікатором контексту
<code>static inline struct ctx_entry *get_ctxe_by_iur(const struct iurhdr *iur)</code>	Реалізує пошук у таблиці контекстів за існуючими полями протоколів IP,UDP,RTP
<code>static struct ctx_entry *create_ctx(const struct iurhdr *iur)</code>	Створює новий запис у таблиці контекстів
<code>static inline void delete_ctx(ctxid_t ctxid)</code>	Видалення запису з таблиці контекстів за ідентифікатором контексту
<code>static void flush_ctxtbl(void)</code>	Очищення таблиці контекстів

Висновки та результати:

1. Невід'ємною частиною практичної реалізації будь-якого механізму обміну інформацією є специфікація алгоритмів його роботи та характеристик протоколів зв'язку.
2. Пропонована специфікація логічних та процедурних характеристик протоколу дозволяє реалізувати тунелювання голосового навантаження в межах понад чотирьох тисяч одночасних сесій.
3. Розроблений алгоритм формування скорочених варіантів полів «Порядковий номер» та «Позначка часу» мінізаголовка дозволяє досягти додаткового скорочення обсягу службової інформації, що передається із кожним голосовим фреймом.
4. Програмна реалізація протоколу може бути використана для поєднання серверів IP-телефонії з метою скорочення загального обсягу навантаження на канал зв'язку між ними.
5. Подальший розвиток протоколу та програмної реалізації способу може здійснюватись у напрямку удосконалення методів визначення контекстів за змістом службових заголовків, а також шляхом додаткового стиснення корисного голосового навантаження враховуючи особливості кодеків.

Література

1. Каптур В.А. Метод мінімізації службової інформації при тунелюванні IP-навантаження / В.А. Каптур, Є.В. Добровольський, О.О. Яніна // Радіоелектронні і комп'ютерні системи. – 2009. – № 4. – С. 91 – 98.
2. Таненбаум Э. Компьютерные сети / Таненбаум Э. – СПб.: Питер, 2005. – 992 с.
3. Каптур В.А. Оцінка ефективності процесу тунелювання голосового навантаження з агрегацією сесій та компресією заголовків / В.А. Каптур, Є.В. Добровольський, О.О. Яніна // Радіоелектронні і комп'ютерні системи. – 2010. – № 3. – С. 92 – 97.
4. Декларативний патент на корисну модель 57700 Україна, Н 04 L 12/46, Н 04 L 12/56, Н 04 L 29/02, Н 04 L 99/00. Спосіб мінімізації службової інформації при тунелюванні RTP-навантаження / Каптур В.А., Добровольський Є.В., Яніна О.О.; заявлено 02.08.2010. опубл. 10.03.11. Бюл. № 5.
5. RTP: A Transport Protocol for Real-Time Applications. [Електронний ресурс] / H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson // RFC 3550 – Режим доступу: <http://www.ietf.org/rfc/rfc3550.txt>.
6. Яніна О.О. Математическое моделирование процесса туннелирования голосового трафика с агрегацией сессий и компрессией заголовков / О.О. Яніна // 66-я наук.-техніч. конф. проф.-виклад. складу, науковців, аспірантів та студентів, (Одеса, 5 – 8 грудня 2011 р.) / Одеск. націон. акад. зв'язку. – Одеса: ОНАЗ, 2011.