МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

П. В. Кучернюк

ТЕХНОЛОГІЇ МОНІТОРИНГУ ТА ТРАФІК-ІНЖИНІРИНГУ В ТЕЛЕКОМУНІКАЦІЙНИХ МЕРЕЖАХ

Затверджено Вченою радою КПІ ім. Ігоря Сікорського
як підручник для здобувачів ступеня магістра
за спеціальністю 172 Телекомунікації та радіотехніка

Київ
КПІ ім. Ігоря Сікорського
2021

Рецензенти: *Забара С.С.*, д-р техн. наук, проф.,
завідувач кафедри інформаційних технологій та програмування Інституту комп'ютерних технологій Відкритого міжнародного університету розвитку людини «Україна»

Чупринка В.І., д-р техн. наук, проф.,
проф. кафедри комп'ютерних наук та технологій факультету мехатроніки та комп'ютерних технологій Київського національного університету технологій та дизайну

Відповідальний редактор *Корнєв В.П.*, канд. техн. наук, доц.,
доц. кафедри конструювання електронно-обчислювальної апаратури факультету електроніки КПІ ім. Ігоря Сікорського

Гриф надано Вченою радою КПІ ім. Ігоря Сікорського (протокол № 3 від 15.03.2021 р.)

Електронне мережне навчальне видання

Кучернюк Павло Валентинович, канд. техн. наук, доц.

ТЕХНОЛОГІЇ МОНІТОРИНГУ ТА ТРАФІК-ІНЖИНІРИНГУ В ТЕЛЕКОМУНІКАЦІЙНИХ МЕРЕЖАХ

Технології моніторингу та трафік-інжинірингу в телекомунікаційних мережах [Електронний ресурс] : підручник для студ. спеціальності 172 «Телекомунікації та радіотехніка» / П. В. Кучернюк; КПІ ім. Ігоря Сікорського. – Електронні текстові данні (1 файл: 5,2 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2021. – 257 с.

Підручник містить матеріали, що використовуються як безпосередньо у теоретичному курсі, так і для самостійної роботи студентів та підготовки до лабораторних робіт. В підручнику розглянуто міжнародні стандарти і концепції побудови систем управління, протоколи управління мережним обладнанням, програмні та апаратні засоби аналізу та управління інформаційними мережами, поняття якості обслуговування та характеристики й механізми його забезпечення, архітектурні особливості диференційованих і інтегрованих послуг, механізми забезпечення QoS канального рівня таких технологій як ATM, Frame Relay, набору стандартів IEEE 802, особливості технології мультипротокольної комутації міток MPLS, інструменти класифікації та маркування пакетів і рекомендації щодо забезпечення QoS для різних типів трафіку згідно документу «Базові Основи QoS» фірми Cisco.

Призначений для студентів спеціальності 172 «Телекомунікації та радіотехніка», інженерно-технічних спеціалістів та аспірантів.

ЗМІСТ

ВСТУП

Сучасний етап розвитку телекомунікаційних і інформаційних технологій характеризується конвергенцією різних телекомунікаційних мереж, мереж передачі даних і IP-мереж в єдину транспортну інформаційну мережу, що здійснює інтегровану передачу голосових, відео та інших даних. Мережним адміністраторам у своїй роботі доводиться мати справу з мережами, що складаються з багатьох типів устаткування різних виробників і різноманітногопрограмного забезпечення для реалізації і підтримки тих або інших інформаційних сервісів і технологічних систем в мережі. Системи управління такими мережами, окрім типових завдань моніторингу мережних пристроїв, повинні вирішувати завдання по аналізу завантаження каналів передачі і мережних пристроїв та забезпеченню якості обслуговування при передачі різних типів даних, контролю доступу до інформаційних ресурсів, завдання обліку використання інформаційних ресурсів користувачами. Для підвищення ефективності управління необхідно створювати інтегровані системи управління мережею і її компонентами, які реалізовані на основі міжнародних стандартів і здатні працювати в гетерогенних мережах. Такі системи дозволять мережним адміністраторам контролювати і організовувати роботу мереж різного масштабу. Архітектура таких систем має бути такою, що масштабується, достатньо недорогою для невеликих систем і, в той же час, достатньо гнучкою для вирішення завдань, які виникатимуть при подальшому розвитку мережі.

Підручник з дисципліни «Технології та засоби керування в інформаційних мережах» є третім з групи підручників/навчальних посібників з дисциплін інформаційно-телекомунікаційного напрямку спеціальності 172 «Телекомунікації та радіотехніка». Дисципліна «Технології та засоби керування в інформаційних мережах» викладається здобувачам ступеня магістра за освітньою програмою підготовки «Інформаційно-обчислювальні засоби радіоелектронних систем» і направлена на вивчення специфічних питань, пов’язаних зі створенням

інтегрованих систем управління мережами та їх компонентами, що реалізовані на основі міжнародних стандартів, побудовою архітектурних рішень та їх реалізації у вигляді механізмів і протоколів канального і мережного рівнів для забезпечення якості обслуговування при передачі трафіку різних мережних додатків як у корпоративних мережах, так і у мережах операторів телекомунікаційних послуг.

Підручник містить матеріали, що використовуються як безпосередньо в теоретичному курсі дисципліни «Технології та засоби керування в інформаційних мережах», так і для самостійної роботи студентів, підготовки до лабораторних робіт. Ці матеріали поділено на десять розділів, у яких розглянуто міжнародні стандарти і концепції побудови систем управління (OSI Management FrameWork та Telecommunication Management Network – TMN), протоколи управління мережним обладнанням (SNMP та CMIP), програмні та апаратні засоби аналізу та управління інформаційними мережами, поняття якості обслуговування та характеристики й механізми його забезпечення, базові алгоритми обслуговування черг, які використовуються для вирішення задача забезпечення QoS, архітектурні особливості диференційованих і інтегрованих послуг та механізми і протоколи для їх забезпечення, механізми забезпечення QoS канального рівня таких технологій як ATM, Frame Relay, набору стандартів IEEE 802, особливості технології мультипротокольної комутації міток MPLS, інструменти класифікації та маркування пакетів і рекомендації щодо забезпечення QoS для різних типів трафіку згідно документу «Базові Основи QoS» фірми Cisco.

Підручник призначений для студентів спеціальності 172 «Телекомунікації та радіотехніка», інженерно-технічних спеціалістів та аспірантів.

1. СТАНДАРТИ УПРАВЛІННЯ OSI

1.1. Загальні відомості про модель мережного управління OSI Management FrameWork

Модель мережного управління OSI – OSI Management FrameWork – визначена в документі ISO/IEC 7498-4: Basic Reference Model, Part 4, Management FrameWork [Д:1]. Вона є розвитком загальної семирівневої моделі взаємодії відкритих систем для випадку, коли одна система управляє іншою.

Документ ISO/IEC 7498-4 складається з наступних основних розділів: терміни і загальні концепції, модель управління системами, інформаційна модель, функціональні області управління системами, структура стандартів управління системами.

В рамках даної моделі обмін управляючою інформацією з використанням протоколу управління (Management Protocol) відбувається між суб'єктами додатків управління системами (Systems Management Application Entities, SMAE) [О:1, Д:1]. Суб'єкти SMAE розташовані на прикладному рівні семирівневої моделі OSI і є елементами служби управління. Суб'єкт в моделі OSI – це активний в даний момент елемент протоколу якого-небудь рівня, що бере участь у взаємодії. Прикладами SMAE є агенти і менеджери систем управління.

Архітектуру системи мережного управляння можна описати за допомогою наступної схеми взаємодії [О:1, Д:1]. На пристроях, що керуються, (мережні пристрої, комп'ютерні системи тощо) працюють спеціальні програми (агенти), які надсилають попереджувальні сигнали при розпізнавання мережної проблеми. Керуючі програми (менеджери) приймають ці повідомлення та, у відповідь, виконують визначені дії (наприклад, сповіщають системного адміністратора, намагаються автоматично поновити систему, вимикають систему, тощо). Менеджер звертається до агентів для того, щоб отримати або перевірити значення деяких змінних. Такі запити можуть надсилатися автоматично, або адміністратором. Агент спочатку збирає інформацію про пристрій, що керується, зберігає цю інформацію в локальній базі даних керування, і,

наприкінці, надсилає її менеджеру системи управління, використовуючи для цього протокол управління (наприклад, SNMP – Simple Network Management Protocol).

Визначення функцій агентів і менеджерів в стандартах OSI досить добре узгоджуються з визначеннями систем SNMP (Simple Network Management Protocol), за деякими виключеннями в термінології [О:1, Д:1]. Повідомлення, які агент посилає менеджеру за своєю ініціативою, називаються **повідомленнями – notifications.**

Менеджер не тільки збирає і зіставляє дані, що отримуються від агентів, на основі цих даних він може також виконувати адміністративні функції, управляючи операціями віддалених агентів.

У стандартах OSI межі між менеджерами і агентами не дуже чіткі. Суб'єкт SMAE, що виконує в одному взаємозв'язку роль менеджера, може в іншій взаємодії виконувати роль агента, і навпаки.

Щоб менеджер і агент змогли взаємодіяти, кожен з них повинен мати певні дані про іншого. Ці знання модель OSI називає контекстом додатку (Application Context, AC) [О:1, Д:1]. AC описує елементи прикладного рівня стека OSI, які використовуються агентами і менеджерами.

Прикладний рівень стека OSI включає декілька допоміжних служб загального призначення, які використовуються прикладними протоколами і додатками користувача (у тому числі і додатками управління) для автоматизації найбільш часто виконуваних дій. На прикладному рівні стека OSI існують наступні допоміжні служби [О:1, Д:1].

- **ACSE** (Association Service Element) – сервісний елемент управління асоціацією. Відповідає за встановлення з'єднань між додатками різних систем. З'єднання (сесія, сеанс) на прикладному рівні OSI носить назву асоціації. Асоціації бувають індивідуальними і груповими (shared).
- **RTSE** (Reliable Transfer Service Element) – сервісний елемент надійної передачі. Займається підтримкою в рамках асоціації відновлення діалогу, викликаного розривом комунікаційних служб, які розташовані нижче.

- **ROSE** (Remote Operations Service Element) – сервісний елемент віддаленої взаємодії. Організовує виконання програмних функцій на віддалених машинах (аналог служби виклику віддалених процедур RPC).

Основна модель управління OSI включає [О:1, Д:1]:

– управління системами,

– управління N-рівнем,

– операції N-рівня.

Таке розбиття на три області зроблене для того, щоб врахувати всі можливі ситуації, що виникають при управлінні.

Управління системами має справу з керованими об’єктами на всіх семи рівнях OSI, включаючи прикладний рівень. Воно базується на надійній передачі управляючої інформації між кінцевими системами зі встановленням з’єднання. Необхідно підкреслити, що модель управління OSI не дозволяє використання служб без встановлення з’єднання.

Управління N-рівнем обмежене керованими об’єктами якогось певного рівня семирівневої моделі. Протокол управління використовує при цьому комунікаційні протоколи рівнів, що розташовані нижче. Управління N-рівнем корисне, коли немає можливості використовувати всі сім рівнів OSI. В цьому випадку дозволяється користуватися протоколом управління N-рівня, який призначений чітко для даного рівня. Прикладами рівневого протоколу управління є протоколи управління для локальних мереж, розроблені інститутом IEEE, які обмежені рівнями 1 і 2.

Операції N-рівня зводяться до моніторингу і управління на основі керуючої інформації, що міститься в комунікаційних протоколах тільки даного рівня. Наприклад, дані моніторингу мережі, що містяться в кадрах STM-n технології SDH, відносяться до операцій 1-рівня, а саме фізичного рівня.

Стандарти на управління N-рівнем і операції N-рівня не входять в набір стандартів управління OSI. Стандарти OSI розглядають тільки управління системами за допомогою повного семирівневого стека.

1.2. Інформаційна модель управління

Керований об’єкт – це уявлення OSI про ресурс, який підлягає управлінню [О:1, Д:1]. Конкретний керований об’єкт – це екземпляр (instance) деякого класу керованих об’єктів. Модель управління OSI широко використовує об’єктно-орієнтований підхід. Клас керованих об’єктів – це сукупність об’єктів, які характеризуються набором властивостей, які можуть бути обов’язковими або умовними. За допомогою опису одного класу керованих об’єктів, наприклад комутаторів, можна створити інший клас керованих об’єктів, наприклад комутаторів, що підтримують техніку VLAN, успадкувавши всі властивості класу комутаторів, але додавши нові атрибути.

Для управління ресурсами менеджер і агент мають бути обізнані про деталі цих ресурсів. Деталізація представлення керованих об’єктів, які потрібні для виконання функцій управління, зберігається в базі даних, відомій як Management Information Base (MIB) [О:1, Д:1]. Бази MIB OSI зберігають не тільки описи класів керованих об’єктів, але і характеристики мережі і її елементів. Бази MIB містять характеристики кожної частини керованого устаткування і ресурсів. MIB також включає опис дій, які можуть виконуватися на основі зібраних даних або ж викликаються зовнішніми командами. Бази MIB дозволяють зовнішнім системам опитувати, змінювати, створювати і видаляти керовані об’єкти (реальні ресурси мережі при цьому, звісно, продовжують працювати).

MIB – це концептуальна модель, і вона не має ніякого зв’язку зі способом фізичного або логічного зберігання даних про ресурс. Стандарти не визначають аспекти власне зберігання даних. Протоколи OSI визначають синтаксис інформації, що зберігається в MIB, і семантику обміну даними.

Велика система управління зазвичай складається з великої кількості агентів і менеджерів. Для організації автоматичної взаємодії між менеджерами і агентами необхідно якимсь чином задати дані, що містять характеристики агентів і менеджерів. Менеджеру необхідно знати про те, які агенти працюють в системі управління, їх імена і мережні адреси, підтримувані ними класи

керованих об’єктів і тому подібне. Агенту також необхідна аналогічна інформація про менеджерів, оскільки йому потрібно відправляти за своєю ініціативою повідомлення і відповідати на запити менеджерів.

Такі дані називаються в моделі OSI керуючими знаннями, що розділяються, **(shared management knowledge)** між менеджером і агентом [О:1, Д:1]. У системах SNMP організація цих даних не стандартизована, і в кожній конкретній системі управління ці дані зберігаються в індивідуальній формі.

Керуючі знання, що розділяються, мають бути відомі до встановлення асоціації між агентом і менеджером. Зазвичай вони зберігаються в якому-небудь файлі або розподіленій базі даних і викликаються кожного разу, коли встановлюється асоціація. Під час встановлення асоціації відбувається обмін керуючими знаннями що розділяються.

Стандарт ISO 10164-16.2 визначає модель об’єктів керуючих знань і класи таких об’єктів. Крім того, визначені функції роботи з керуючими знаннями.

Є три типи керуючих знань і відповідно, три типи об’єктів, які описують ці знання [О:1].

- **Знання репертуару (Repertoire Knowledge)** описують можливості керованої системи, що включають перелік підтримуваних класів керованих об’єктів, підтримувані функції управління й іменування. Знання репертуару допомагають менеджеру ідентифікувати можливості керованих систем без доступу до них.
- **Знання визначень (Definition Knowledge)** включають формальні описи класів керованих об’єктів, категорії тестів, класів взаємозв’язків і визначення керуючої інформації, яка підтримується керованою системою.
- **Знання про екземпляри (Instance Knowledge)** забезпечують інформацію про конкретні екземпляри керованих об’єктів, наявних в керованій системі.

У системі управління знання про підтримувані класи об’єктів і про породжені екземпляри об’єктів повинні зберігатися в якій-небудь формі, зручній для надання модулям системи управління доступу до цієї інформації.

Архітектура управління OSI передбачає декілька схем представлення інформації в базі даних про керовані об’єкти та їх класи. Ці схеми зазвичай називають деревами через ієрархічну організацію інформації. Існують такі дерева [O:1].

- **Дерево спадкоємства (Inheritance Tree)**, зване також деревом реєстрації. Описує стосунки між базовими і похідними класами. Підлеглий клас успадковує всі характеристики суперкласу і доповнює їх специфічними розширеннями (додатковими атрибутами, поведінками і діями). Дерево спадкоємства може бути глобальним, тобто починатися з кореня, що представляє весь світ, або локальним, таким, що має корінь, відповідний верхньому рівню об’єктів даної організації або мережі. Всі керовані об’єкти OSI мають бути зареєстровані в глобальному дереві ISO (у якому зареєстровані об’єкти MIB-I, MIB-II, RMON MIB стандарту SNMP). Об’єкти, що представляють міжнародні стандарти, реєструються в міжнародній гілці дерева, а приватні моделі, розроблені виробниками систем управління, реєструються в гілках дерева, що починаються з гілки private.
- **Дерево включень (Containment Tree)**. Описує стосунки включення керованих об’єктів реальної системи.
- **Дерево імен (naming tree)** визначає спосіб іменування об’єктів в системі управління. Об’єкти OSI можуть мати імена декількох типів: відносне відмітне ім’я (Relative Distinguished Name, RDN), відмітне ім’я (Distinguished Name, DN), іноді зване повним відмітним ім’ям (Full Distinguished Name, FDN), і локальне відмітне ім’я (Local Distinguished Name, LDN). Ці імена пов’язані з деревом включень, оскільки визначають імена об’єктів щодо об’єктів, що включають їх. Відносне ім’я (RDN) відповідає короткому імені, яке однозначно визначає об’єкт серед безлічі інших об’єктів, підлеглих тому ж батьківському об’єкту. Наприклад, ім’я interface_a є RDN-іменем, що унікально характеризує об’єкт серед об’єктів, підлеглих об’єкту node_a. Повне відмітне ім’я (FDN) є послідовністю RDN-імен, що починається у вершині глобального дерева імен, тобто дерева, що описує деяку глобальну мережу. Нарешті, локальне відмітне ім’я – це послідовність RDN-імен, яке починається не в глобальному корені,

а в корені дерева імен локальної системи управління, що відповідає за частину глобального дерева імен даної мережі.

Дерево імен зазвичай поєднується з деревом включень.

Приклад дерева включень показаний на рис. 1.1 [О:1]. Екземпляр керованого об’єкта класу «core-switch» (комутатор ядра) має ім’я SB1, а також атрибут «max-slotes», що описує максимальну кількість слотів даного класу комутаторів, рівний, в даному випадку, 8. У цей об’єкт включено ряд інших об’єктів: об’єкти класів «switch» і «router», які у свою чергу включають об’єкти типу «interface», що описують порти модулів комутатора.

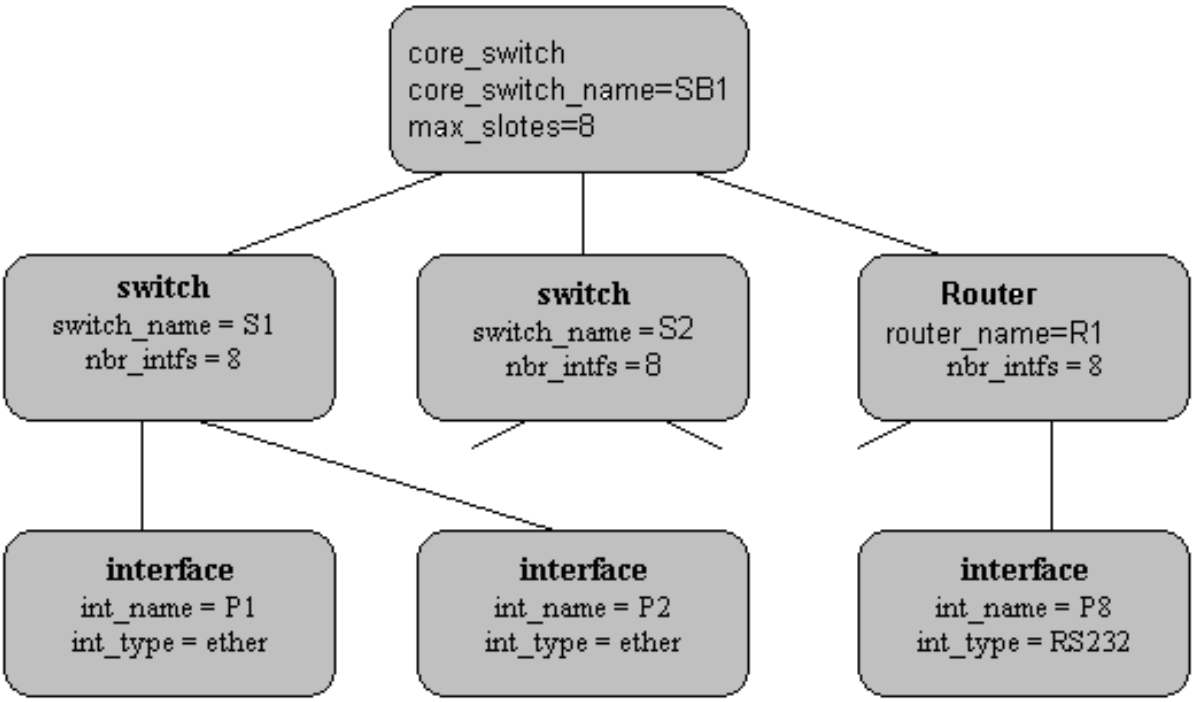

Рисунок 1.1 – Приклад дерева включень

Ім’я класу об’єкта дозволяє звернутися до опису класу і дізнатися повний список атрибутів цього класу або посилання на батьківський клас, у якого успадковуються всі або деякі атрибути. Ім’я екземпляра об’єкта дає інформацію про приналежність конкретного модуля або інтерфейсу певному комунікаційному пристрою. Наприклад, ім’я SB1.S1.P2 визначає другий порт модуля комутатора S1, що входить до складу корпоративного комутатора SB1.

1.3. Правила визначення керованих об’єктів

Класи керованих об’єктів OSI повинні визначатися відповідно до стандарту GDMO (Guidelines for the Definition of Managed Objects – Правила визначення керованих об’єктів – стандарт ISO 10165-4).

У GDMO визначається декілька шаблонів (templates) – порожніх форм, які заповнюються для опису певного класу керованих об’єктів [O:1]. У шаблоні класу перераховуються комплекти властивостей (PACKAGES), які складають клас. Шаблон комплекту властивостей «PACKAGE» включає «Атрибути», «Групи атрибутів», «Дії», «Поведінку» і «Повідомлення», тобто властивості, згруповані для зручності опису класу об’єктів. Стосунки спадкоємства між класами описуються за допомогою шаблону «Зв’язування імен».

«Атрибути» і «Групи атрибутів» визначають параметри об’єкта, які можна читати і дізнаватися з них про стан об’єкта. Властивості «Дії» описують можливі керуючі дії, які допускається застосовувати до даного об’єкта – наприклад, мультиплексувати декілька вхідних потоків в один вихідний. Властивості «Поведінка» описують реакцію об’єкта на застосовану до нього дію. «Повідомлення» складають набір повідомлень, які генерує об’єкт за своєю ініціативою.

Заповнені шаблони GDMO визначають представлення класу і його властивостей.

Заповнення шаблонів виконується відповідно до нотації ASN.1. На відміну від стандартів SNMP, що використовують тільки підмножину типів даних ASN.1, при використанні протоколу CMIP (Common Management Information Protocol) в GDMO застосовується повна версія ASN.1.

На підставі правил GDMO визначено декілька міжнародних стандартів на класи керованих об’єктів. Документи Definition of Management Information (DMI, ISO/IEC 10165-2:1991) і Generic Management Information (GMI, ISO/IEC CD 10165-5:1992) є першими визначеннями MIB на основі остаточної версії GDMO. Ці MIB можуть розглядатися як ISO-еквівалент для Internet MIB II, оскільки вони створюють основу для побудови більш специфічних MIB.

Наприклад, DMI визначає клас об’єктів, який є верхнім суперкласом і називається «Top». Він містить атрибути, які успадковуються всіма іншими класами керованих об’єктів. Визначені також класи об’єктів «System» і «Network», що займають верхні позиції в дереві спадкоємства, і будь-який агент повинен розуміти їх атрибути. У 1992 році була завершена робота і над більш специфічними класами об’єктів – об’єктами мережного і транспортного рівнів (ISO/IEC 10737-1 і ISO/ IEC 10733).

Сьогодні багато організацій працюють над створенням класів об’єктів на основі GDMO. Це і міжнародні організації по стандартизації – ISO, ITU-T, ANSI, ETSI, X/Open, і організації, які розробляють платформи та інструментальні засоби для систем управління, такі як: SunSoft, Hewlett-Packard, Vertel, ISR Global. Для телекомунікаційних мереж в рамках архітектури TMN розроблений стандарт M.3100, який описує ряд специфічних для телекомунікаційних мереж класів об’єктів. Описи класів керованих об’єктів OSI реєструються як у приватних гілках дерева ISO – гілках компаній Sun, Hewlett-Packard, IBM і т. п., так і в публічних гілках, контрольованих ISO або іншими міжнародними органами стандартизації.

Запитання для самоперевірки та контролю засвоєння знань

1. Яка модель визначає концептуальні основи побудови систем керування?
2. Як визначено обмін управляючою інформацією в рамках даної моделі?
3. Що розуміють під суб’єктом SMAE?
4. На якому рівні моделі OSI розташовані суб’єкти SMAE?
5. Які суб’єкти SMAE відповідають за збір інформації про керований об’єкт?
6. Які суб’єкти SMAE виконують дії над керованим об’єктом?
7. За допомогою якого протоколу здійснюється обмін інформацією між суб’єктами SMAE?
8. Що необхідно для організації взаємодії між агентами та менеджерами?
9. Що розуміють під асоціацією в системах управління?
10. Які функції виконує сервісний елемент управління асоціацією (ACSE)?
11. Які функції виконує сервісний елемент надійної передачі (RTSE)?

12. Які функції виконує сервісний елемент віддаленої взаємодії (ROSE)?

13. Які рівні включає модель управління OSI ?

14. Що передбачає управління системами?

15. Що передбачає управляння N-рівня?

16. Що передбачають операції N-рівня?

17. Операції якого рівня стандартизовані в рамках моделі OSI ?

18. Що розуміють під керованим об’єктом в рамках моделі управління OSI ?

19. Що розуміють під класом керованих об’єктів?

20. Яка інформація міститься у базах MIB?

21. Що розуміють в моделі управління OSI під керуючими знаннями?

22. Що описують знання репертуару?

23. Що описують знання визначень?

24. Що описують знання про екземпляри?

25. Як називають схеми представлення інформації в базі даних про керовані об’єкти?

26. Що описує дерево спадкоємства?

27. Що описує дерево включень?

28. Що описує дерево імен?

29. Які типи імен об’єктів визначені в дереві імен?

30. Що визначає відносне відмітне ім’я (RDN)?

31. Що визначає повне відмітне ім’я (FDN)?

32. Що визначає локальне відмітне ім’я (LDN)?

33. Згідно яких правил здійснюється опис керованих об’єктів в моделі управління OSI?

34. За допомогою чого здійснюється опис керованих об’єктів?

35. Що включає шаблон комплекту властивостей?

36. Згідно яких правил виконується заповнення шаблонів?

2. КОНЦЕПЦІЯ МЕРЕЖІ УПРАВЛІННЯ ТЕЛЕКОМУНІКАЦІЯМИ (TMN)

2.1. Загальні принципи концепції TMN

Як було зазначено у попередньому розділі, питаннями стандартизації систем управління займається безліч міжнародних організацій. За основу при розробці рекомендацій і стандартів зазвичай беруться найбільш універсальні й ефективні підходи і рішення, реалізовані в продуктах різних крупних фірм. В процесі такої стандартизації була створена концепція побудови системи управління телекомунікаційними мережами [О:1, 2, 3, Д:2, 3, 4]. Мета розробки даної концепції – стандартизація підходу до управління мережами з різними типами устаткування і створення єдиної системи управління, незалежної від виробників окремих мережних елементів. Розроблений набір рекомендацій отримав назву концепції мережі управління телекомунікаціями (Telecommunication Management Network – TMN). Концепція TMN описана в рекомендації М.3010 міжнародного союзу електрозв'язку (International Telecommunication Union – ITU). TMN тісно пов'язана з моделлю мережного управління OSI, розглянутою у попередньому розділі.

У рекомендації ITU-T M.3010 визначено загальні принципи планування, функціонування і технічного обслуговування системи управління телекомунікаціями (TMN – Telecommunications Management Network). Основним принципом TMN є забезпечення можливостей керування телекомунікаційними мережами, які складаються з різних типів операційних систем і телекомунікаційних пристроїв і використовують стандартні протоколи і інтерфейси для передачі даних. На рис. 2.1 представлений взаємозв'язок між TMN і телекомунікаційною мережею. Операційні системи здійснюють обробку всієї інформації, необхідної для виконання функцій по управлінню. Робочі станції забезпечують інтерфейс користувача, за допомогою якого обслуговуючий персонал взаємодіє з мережею управління.

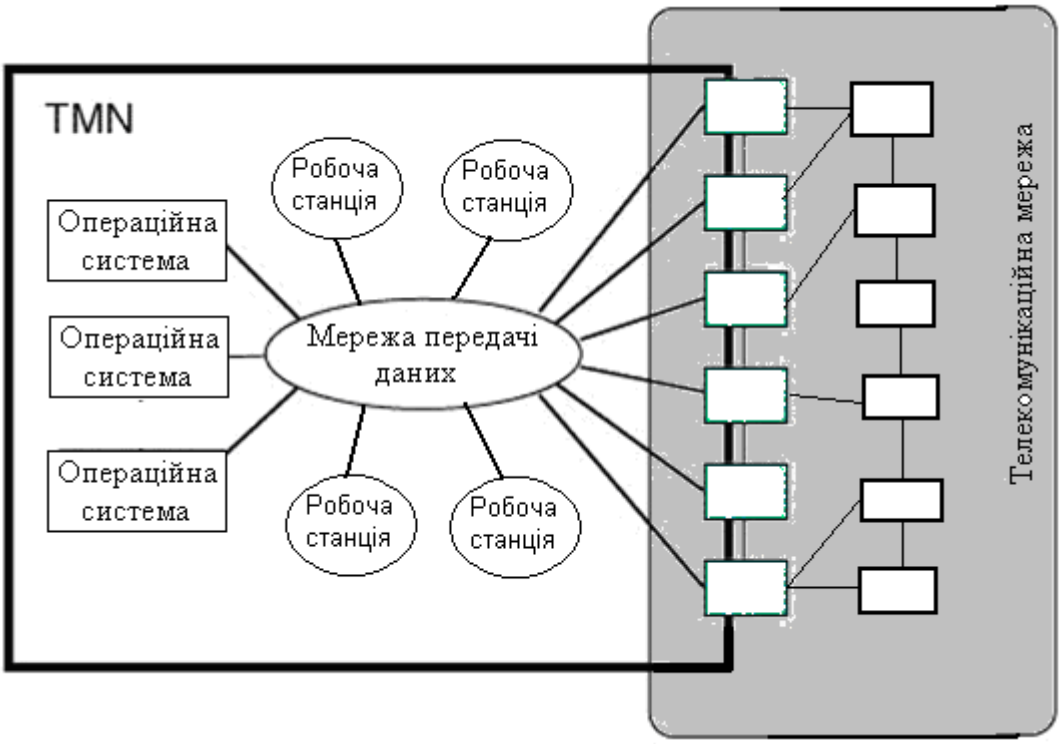

Рисунок 2.1 – Взаємозв'язок між системою управління і телекомунікаційною мережею

Мережа передачі даних призначена для зв'язку між мережними елементами, операційними системами та іншими компонентами TMN. TMN може змінюватися від достатньо простого з'єднання між операційною системою і окремим телекомунікаційним пристроєм до великої мережі, що з'єднує велику кількість операційних систем і телекомунікаційну апаратуру різних типів. Необхідно відзначити, що TMN принципово є самостійною системою, яка забезпечує інтерфейси з телекомунікаційною мережею в декількох різних точках для отримання інформації і управління її роботою.

2.2. Рівні управління мережею

Згідно моделі ISO система управління мережею будується ієрархічно і має наступні рівні (від низу до верху) (рис. 2.2) [О:1, 2, 3, Д:2, 3, 4]:

- мережних елементів;
- управління елементами;
- управління мережею;

- управління обслуговуванням;
- адміністративного управління.

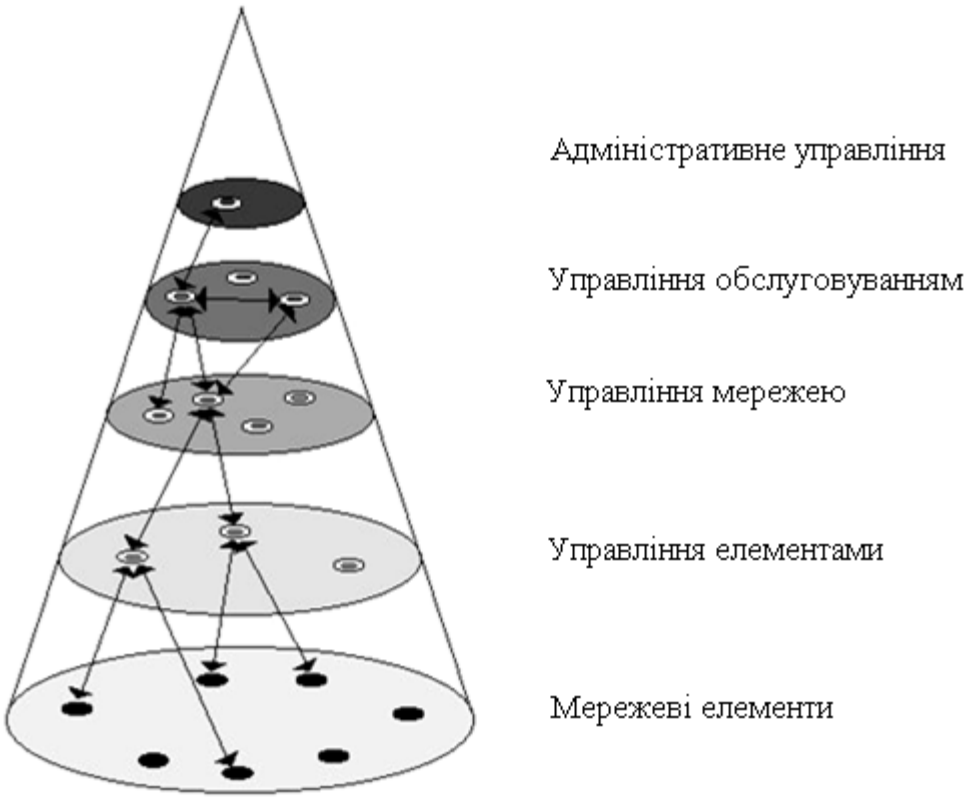

Рисунок 2.2 – Рівні управління телекомунікаційною мережею

Найнижній рівень (**рівень мережних елементів**) є самою телекомунікаційною мережею, яка являє собою об’єкт управління. У якості мережних елементів можуть розглядатися маршрутизаційні та комутаційні станції, системи передачі, мультиплексори, комплекти тестового устаткування тощо. Цей рівень містить об’єкти управління, на яких розташовані програмні або програмно-апаратні модулі (агенти), які відповідають за безпосередній збір інформації про об’єкт.

Рівень управління мережними елементами. Система управління цього рівня дозволяє управляти кожним мережним елементом окремо. На даному рівні системі управління з боку мережного елементу представляються найдокладніші технічні характеристики. Рівень управління мережними елементами реалізує функції конфігурування пристроїв, обробки відмов і управління продуктивністю.

Рівень управління мережею. Даний рівень повинен надавати послуги інтегрованого моніторингу і управління мережею в цілому, безвідносно до технологій, використовуваних в мережі і кількості мережних елементів. Рівень управління мережею повинен надавати можливість організації телекомунікаційної послуги (наприклад, з'єднання), управління маршрутами передачі даних, управління топологією мережі. Даний рівень також повинен реалізовувати функції обробки інформації про аварії з метою виявлення джерела аварії, а також прогнозувати виникнення тих або інших аварій за наслідками робочих характеристик мережі. Відмітною особливістю даного рівня є високий ступінь автоматизації всіх технічних функцій управління, інтегрованого обліку всіх мережних елементів і спостереження робочих характеристик мережі.

Рівень управління послугами абстрагований від більшості технічних завдань управління телекомунікаційними мережами. Завданнями даного рівня є контроль якості послуг, що надаються, ведення біллінгу, прийом заявок від споживачів на надання послуг і ведення реєстру послуг і споживачів. Моніторинг якості на даному рівні може представляти собою постійний автоматизований контроль за двома-трьома основними технічними параметрами послуги за спрощеною схемою: в межах норми, поза нормою, аварія. Рівень управління послугами реалізує механізми контролю за виконанням угоди про рівень сервісу (SLA) і забезпеченням якості обслуговування (QoS).

Рівень управління бізнесом – найвищий рівень логічної ієрархії. Реалізація управління на цьому рівні передбачає управління всім підприємством оператора (провайдера) в цілому. Управління підприємством передбачає управління доходами від послуг, що надаються, управління нерухомістю, фінансами, людськими ресурсами і т. і.

Кожен наступний рівень має вищий ступінь узагальнення, ніж попередній. Інформація про стан рівня поступає вгору, а зверху вниз йдуть керуючі дії. Ступінь автоматизації управління може бути різним, і, зазвичай, має місце

поєднання автоматизованих і ручних процедур. Як правило, чим вище рівень ієрархії управління, тим нижче його ступінь автоматизації. Рівень управління елементами охоплює контроль, відображення параметрів роботи, технічне обслуговування, тестування, конфігурацію окремих елементів або деяких їх підмножин. Рівень мережного управління дозволяє охопити єдиним поглядом всю мережу, контролюючи підмножини мережних елементів та їх взаємозв'язки між собою й управляючи всіма мережними ресурсами. Рівень управління обслуговуванням, на відміну від всіх рівнів, які розташовані нижче і безпосередньо пов'язані з мережею та технічними засобами, орієнтований на користувача. Тут ухвалюються рішення по наданню і припиненню послуг, здійснюється ведення відповідного планування й обліку і тому подібне. Ключовим чинником тут є забезпечення якості обслуговування. Рівень адміністративного управління забезпечує функціонування компанії-оператора телекомунікаційної мережі. Тут вирішуються організаційні і фінансові питання, здійснюється взаємодія з компаніями-операторами інших телекомунікаційних мереж.

2.3. Функції мережного управління

Всі функції, пов'язані з управлінням, можна розбити на дві частини: загальні і прикладні. Загальні функції забезпечують підтримку прикладних і включають, наприклад, переміщення інформації між елементами телекомунікаційної мережі і системи управління, зберігання інформації, її відображення, сортування, пошук і тому подібне [О:1, 2, 3, Д:2, 3, 4].

Додатково до концепції TMN, міжнародним союзом електрозв'язку була запропонована функціональна модель системи мережного управління (рекомендації М.3400), яка отримала назву FCAPS по заголовних літерах функцій системи. У даній моделі виділені наступні ключові функції адміністрування й управління мережами [О:1, 2, 3, Д:2, 3, 4]:

- (F) Fault management / управління відмовами;
- (C) Configuration management / управління конфігурацією;
- (A) Accounting management / облік;

– (P) Performance management / управління продуктивністю;

– (S) Security management / управління безпекою.

Компоненти управління відмовами вирішують задачу виявлення і усунення мережних проблем, ведуть обробку аварійних повідомлень і системних переривань, опит елементів мережі, тестування і діагностику. Засобами управління конфігурацією здійснюються моніторинг і контроль апаратного і програмного забезпечення мережі та будь-яких їх модифікацій. Функції обліку відповідають за розподіл і належне використання мережних ресурсів. Призначення функцій управління продуктивністю – представлення статистики роботи мережі в режимі реального часу, мінімізація заторів і вузьких місць, виявлення тенденцій, що складаються, і планування ресурсів для майбутніх застосувань. Компоненти управління безпеки забезпечують контроль доступу, ведення журналів доступу, захист від зовнішніх і внутрішніх порушників.

Модель FCAPS є розширенням концепції TMN. На кожному з рівнів TMN виконуються деякі або всі функції моделі FCAPS. Взаємозв'язок рівнів TMN і функцій моделі FCAPS може бути представлена в наступному вигляді (рис. 2.3) [О:1, 2, 3, Д:2, 3, 4].

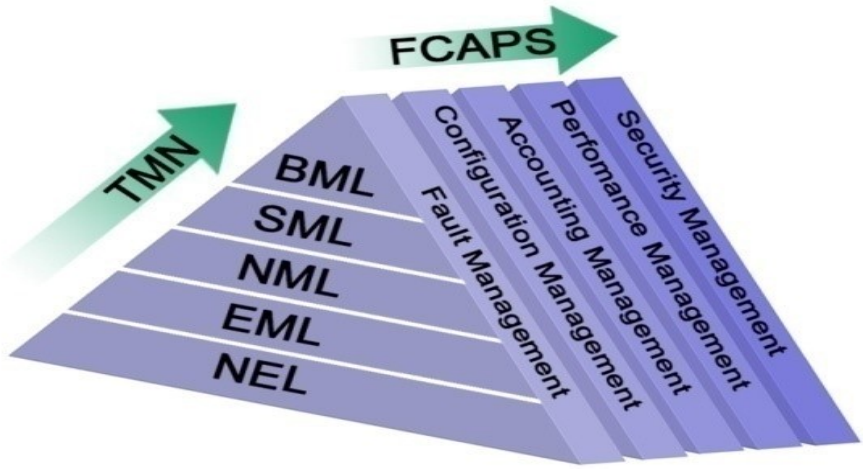

Рисунок 2.3 – Взаємозв'язок між рівнями управління концепції TMN і функціями моделі FCAPS

В результаті, прикладні функції відповідно до концепції TMN і моделі FCAPS розділяються на п'ять категорій (рис. 2.4) [О:1, 2, 3, Д:2, 3, 4].

Рисунок 2.4 – Класифікація функцій мережевого управління

Управління конфігурацією забезпечує інвентаризацію мережних елементів (їх типи, місцезнаходження, ідентифікатори і тому подібне), включення елементів в роботу, їх конфігурування і відключення, організацію і зміну фізичних з’єднань між елементами.

Управління якістю роботи відповідає за контроль і підтримку на необхідному рівні основних характеристик мережі. Воно включає збір, обробку, реєстрацію, зберігання і відображення статистичних даних про роботу мережі і її елементів, виявлення тенденцій в їх поведінці і попередження про можливі порушення в роботі.

Управління усуненням несправностей забезпечує можливості виявлення, визначення місцеположення несправностей в мережі, їх реєстрацію, доведення відповідної інформації до обслуговуючого персоналу, видачу рекомендацій по усуненню несправностей.

Управління розрахунками здійснює контроль за ступенем використання мережних ресурсів і підтримує функції по нарахуванню оплати за це використання.

Управління безпекою необхідне для захисту мережі від несанкціонованого доступу. Воно може включати обмеження доступу за допомогою паролів, видачу сигналів тривоги при спробах несанкціонованого доступу, відключення небажаних користувачів, або, навіть криптографічний захист інформації.

2.4. Функціональна архітектура

Функціональна архітектура TMN описується за допомогою функціональних блоків (ФБ) [О:1, 2, 3]. Основними з них є наступні ФБ: ФБ мережного елементу (NEF – Network Element Function), ФБ операційної системи (OSF – Operations System Function), ФБ робочої станції (WSF – Work Station Function), ФБ медіатора (проміжного пристрою сполучення) (MF – Mediation Function) і ФБ Q-адаптера (QAF – Q-Adapter Function). NEF є моделлю довільного елементу мережі, який підлягає управлінню. OSF забезпечує виконання функцій TMN по обробці, зберіганню і пошуку керуючої інформації. Ці ФБ формують ядро TMN. WSF організовує людино-машинний інтерфейс між системою управління і людиною-оператором. MF обробляє інформацію, що проходить між NEF і OSF, і може здійснювати проміжну обробку і зберігання даних, перетворення протоколів і тому подібне. QAF призначені для взаємодії з мережними елементами або операційними системами, що мають непередбачені в TMN інтерфейси.

Відповідно до ієрархічної структури системи управління визначаються ФБ OSF чотирьох рівнів: управління елементами (NE-OSF), управління мережею (N-OSF), управління обслуговуванням (S-OSF) і адміністративного управління (B-OSF) (рис. 2.5) [О:1, 2, 3].

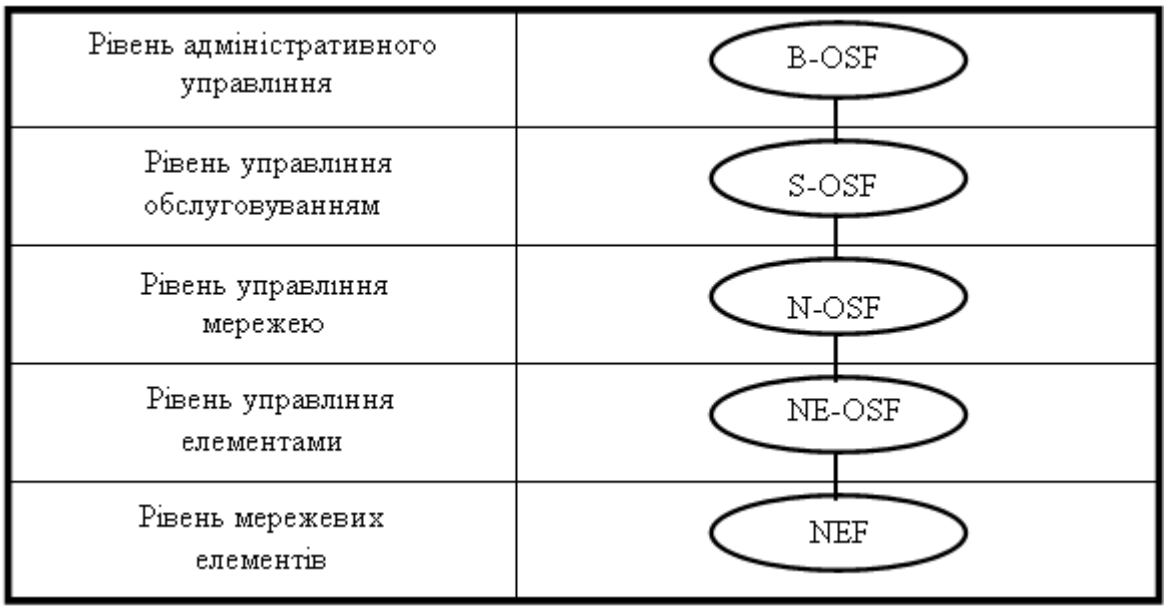

Рисунок 2.5 – Функціональна ієрархія операційних систем

2.5. Інформаційна архітектура

Інформаційна архітектура TMN вводить характерні для моделі взаємозв'язку відкритих систем (OSI) принципи управління, що базуються на об'єктно-орієнтованому підході. Інформаційний обмін описується в термінах керованих об'єктів, що розглядаються як деякі ресурси, над якими здійснюється управління, або які служать для підтримки певних функцій по управлінню [О:1, 2, 3]. Керований об'єкт може представляти також відношення між ресурсами або комбінацію ресурсів (наприклад, мережа). Кожен керований об'єкт належить деякому класу об'єктів, який може бути підкласом іншого класу. Підклас успадковує всі властивості класу, з якого він виділений, і уточнює визначення класу додаванням нових властивостей до тих, які покладені в основу опису вищестоящого класу. Різні класи можуть бути представлені у вигляді дерева, що показує ієрархію успадкованих властивостей [О:1, 2, 3].

Керований об'єкт характеризується:

- атрибутами,
- операціями управління, які можуть бути до нього застосовані,
- повідомленнями, які їм генеруються,

- поведінкою, що є реакцією на команди управління або на інші дії.

Управління телекомунікаційною мережею являє собою прикладний розподілений інформаційний процес [О:1, 2, 3]. Це передбачає необхідність організації обміну інформацією між процедурами управління для цілей моніторингу і контролю різних фізичних і логічних мережних ресурсів. Для управління об’єктами використовується структура «менеджер-агент» (рис. 2.6) [О:1]. Менеджер є частиною розподіленого процесу, яка направляє команди на виконання операцій управління й отримує повідомлення. Агент – це частина процесу, яка безпосередньо управляє відповідними керованими об’єктами. Він відповідає за виконання команд, що направляються йому менеджером, і за інформування менеджера шляхом посилання повідомлень про поведінку підвідомчих об’єктів.

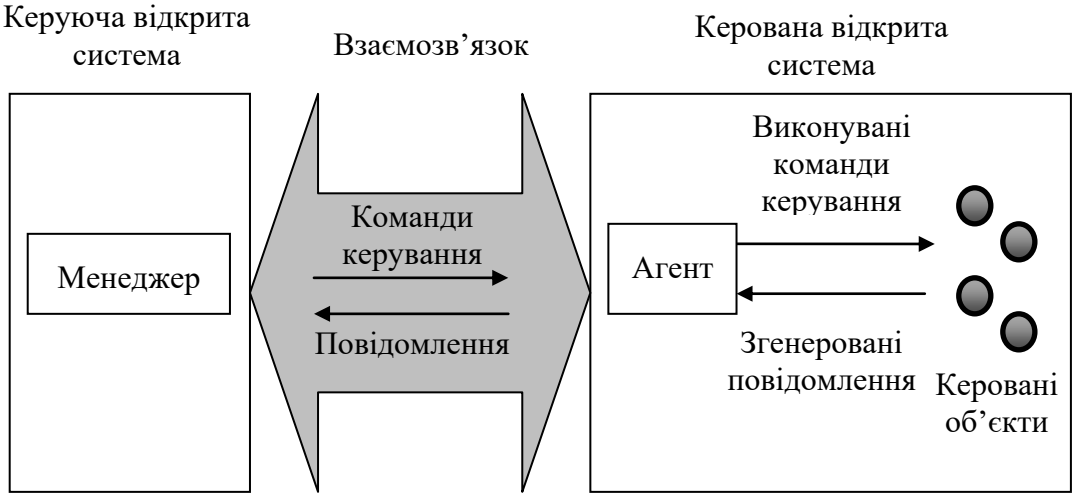

Рисунок 2.6 – Взаємодія між менеджером, агентом і керованими об’єктами

Між менеджерами і агентами може здійснюватися взаємодія за принципом «багато з багатьма» в тому сенсі, що один менеджер може брати участь у обміні інформації з декількома агентами, і один агент – з декількома менеджерами. Весь інформаційний обмін по управлінню між менеджером і агентом виражається у вигляді узгодженого набору команд управління і повідомлень. Спосіб же взаємодії агентів з ресурсами на місцях не є предметом стандартизації.

У рекомендаціях ITU-T у якості протоколу обміну інформацією пропонується використовувати протокол CMIP (протокол загальної інформації – рекомендація X.711), але оскільки він не набув широкого поширення в комп'ютерних мережах, то, замість нього, частіше використовують простий протокол управління мережею – SNMP.

2.6. Особливості схеми «менеджер-агент»

У основі будь-якої системи управління мережею лежить елементарна схема взаємодії агента з менеджером. На основі цієї схеми можуть бути побудовані системи практично будь-якої складності з великою кількістю агентів і менеджерів різного типу [О:1, 2, 3].

Схема «менеджер – агент» представлена на рис. 2.7.

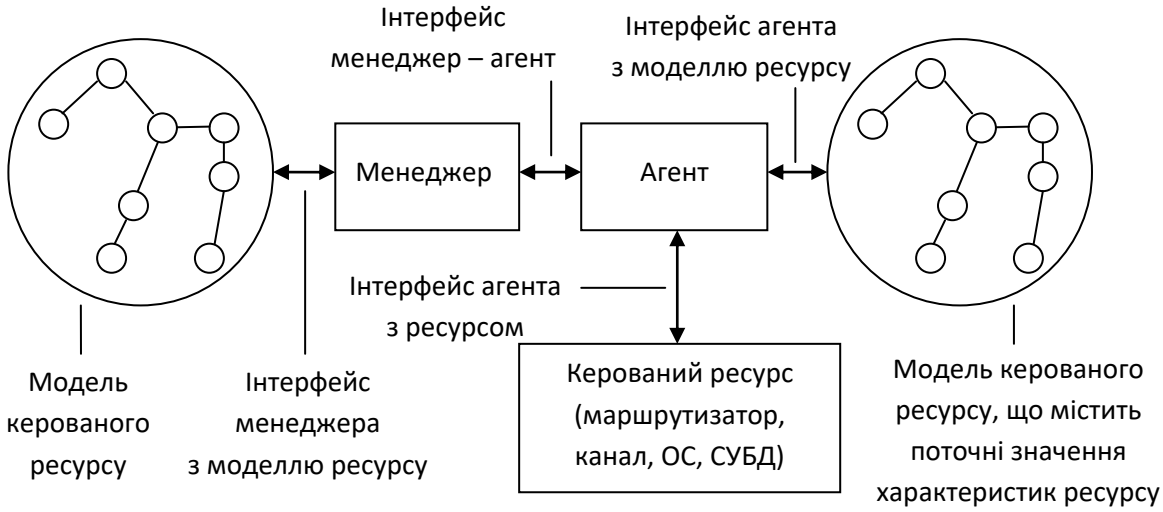

Рисунок 2.7 – Схема взаємодії агента, менеджера і керованого ресурсу

Агент є посередником між керованим ресурсом і основною програмою-менеджером, що управляє. Щоб один і той же менеджер міг управляти різними реальними ресурсами, створюється деяка модель керованого ресурсу, яка відображає тільки ті характеристики ресурсу, які потрібні для його контролю і управління. Наприклад, модель маршрутизатора зазвичай включає такі характеристики, як кількість портів, їх тип, таблицю маршрутизації,

кількість кадрів і пакетів протоколів канального, мережного і транспортного рівнів, що пройшли через порти маршрутизатора.

Менеджер отримує від агента тільки ті дані, які описуються моделлю ресурсу. Агент звільняє менеджера від непотрібної інформації про деталі реалізації ресурсу, надаючи йому оброблену і представлену в нормалізованому вигляді інформацію. На основі цієї інформації менеджер ухвалює рішення по управлінню, а також виконує подальше узагальнення даних про стан керованого ресурсу, наприклад, будує залежність завантаження порту від часу.

Для отримання необхідних даних від об’єкта, а також для видачі на нього дій, що управляють, агент взаємодіє з реальним ресурсом деяким нестандартним способом. Коли агенти вбудовуються в комунікаційне устаткування, то розробник устаткування передбачає точки і способи взаємодії внутрішніх вузлів пристрою з агентом. При розробці агента для операційної системи розробник агента користується тими інтерфейсами, які існують в цій ОС, наприклад інтерфейсами ядра, драйверів і додатків. Агент може забезпечуватися спеціальними датчиками для отримання інформації, наприклад, датчиками температури.

Менеджер і агент повинні мати в своєму розпорядженні одну і ту ж модель керованого ресурсу, інакше вони не зможуть зрозуміти один одного. Проте, у використанні цієї моделі агентом і менеджером є істотна відмінність. Агент наповнює модель керованого ресурсу поточними значеннями характеристик даного ресурсу (модель агента називають базою даних інформації, що управляє, – Management Information Base, MIB). Менеджер використовує модель, щоб знати про те, чим характеризується ресурс, які характеристики він може запитати у агента і якими параметрами можна управляти.

Менеджер взаємодіє з агентами по стандартному протоколу. Цей протокол повинен дозволяти менеджерові запрошувати значення параметрів, що зберігаються в базі MIB, а також передавати агентові інформацію, що управляє, на основі якої той повинен управляти пристроєм. Розрізняють управління

inband, тобто по тому ж каналу, по якому передаються призначені для користувача дані, і управління **out-of-band**, тобто поза каналом, по якому передаються призначені для користувача дані [О:1]. Наприклад, якщо менеджер взаємодіє з агентом, вбудованим в маршрутизатор, по протоколу SNMP, що передає дані керування тим же каналом, яким передаються і дані користувачів, то це буде управління inband. Якщо ж менеджер контролює маршрутизатор телекомунікаційної мережі за допомогою окремої мережі X.25, до якої підключений агент, то це буде управління out-of-band. Управління по тому ж каналу, по якому працює мережа, економніше, оскільки не вимагає створення окремої інфраструктури передачі даних управління. Проте, спосіб out-of-band надійніший, оскільки він надає можливість управляти устаткуванням мережі і тоді, коли якісь елементи мережі вийшли з ладу і по основних каналах устаткування недоступне. Стандарт багаторівневої системи управління TMN має в своїй назві слово Network, що підкреслює, що в загальному випадку для управління телекомунікаційною мережею створюється окрема мережа, що управляє, яка забезпечує режим out-of-band.

Зазвичай менеджер працює з декількома агентами, обробляючи отримувані від них дані і видаючи на них дії, що управляють. Агенти можуть вбудовуватися в кероване устаткування, а можуть і працювати на окремому комп’ютері, пов’язаному з керованим устаткуванням по якому-небудь інтерфейсу. Менеджер зазвичай працює на окремому комп’ютері, який виконує також роль консолі управління для оператора або адміністратора системи.

Агенти можуть відрізнятися різним рівнем інтелекту – вони можуть володіти як наймінімальнішим інтелектом, необхідним для підрахунку кадрів і пакетів, що проходять через устаткування, так і вельми високим, достатнім для виконання самостійно послідовності управляючих дій в аварійних ситуаціях, побудови часових залежностей, фільтрації аварійних повідомлень і тому подібне.

2.7. Структури розподілених систем управління

У крупній корпоративній мережі повністю централізована система управління, побудована на базі єдиного менеджера, буде малоефективною з кількох причин. По-перше, такий варіант не забезпечує необхідної масштабованості по продуктивності, оскільки єдиний менеджер вимушений буде обробляти весь потік повідомлень від всіх агентів, що, при декількох тисячах керованих об'єктів, зажадає дуже високопродуктивної платформи для роботи менеджера і перенавантажуватиме службовою управляючою інформацією канали передачі даних в тій частині мережі, де буде розташований менеджер. По-друге, таке рішення не забезпечить необхідного рівня надійності, оскільки при відмові єдиного менеджера буде втрачено управління мережею [О:1, 2, 3].

Схема «менеджер – агент» дозволяє будувати достатньо складні в структурному відношенні розподілені системи управління.

Зазвичай розподілена система управління включає велику кількість зв'язок менеджер – агент, які доповнюються робочими станціями операторів мережі, за допомогою яких вони дістають доступ до менеджерів (рис. 2.8) [О:1, 2, 3].

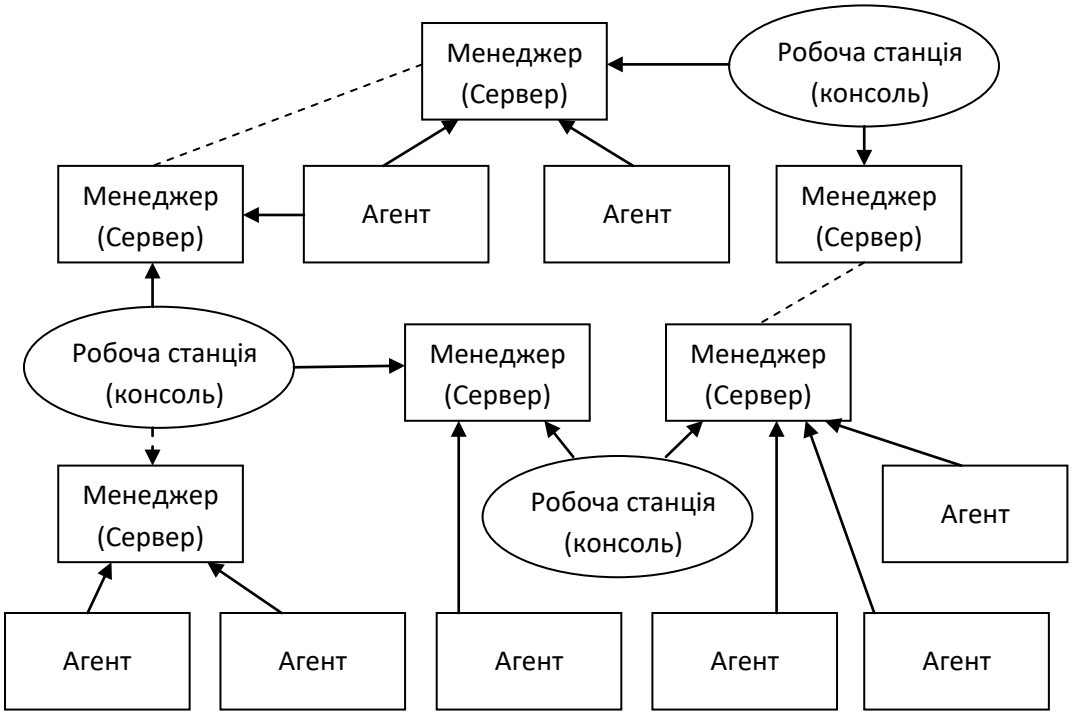

Рисунок 2.8 – Розподілена система управління на основі декількох менеджерів і робочих станцій

Кожен агент збирає дані і керує певним елементом мережі. Менеджери, іноді також звані серверами системи управління, збирають дані від своїх агентів, узагальнюють їх і зберігають в базі даних. Оператори, що працюють за робочими станціями, можуть з'єднатися з будь-яким з менеджерів і, за допомогою графічного інтерфейсу, проглянути дані про керовану мережу, а також видати менеджерові деякі директиви по управлінню мережею або її елементами.

Наявність декількох менеджерів дозволяє розподілити між ними навантаження по обробці даних управління, забезпечуючи масштабованість системи.

Як правило, зв'язки між агентами і менеджерами носять більш впорядкований характер, ніж той, який показаний на рис. 2.8. Найчастіше використовуються два підходи до їх з'єднання – **одноранговий** (рис. 2.9) і **ієрархічний** (рис. 2.10) [О:1, 2, 3].

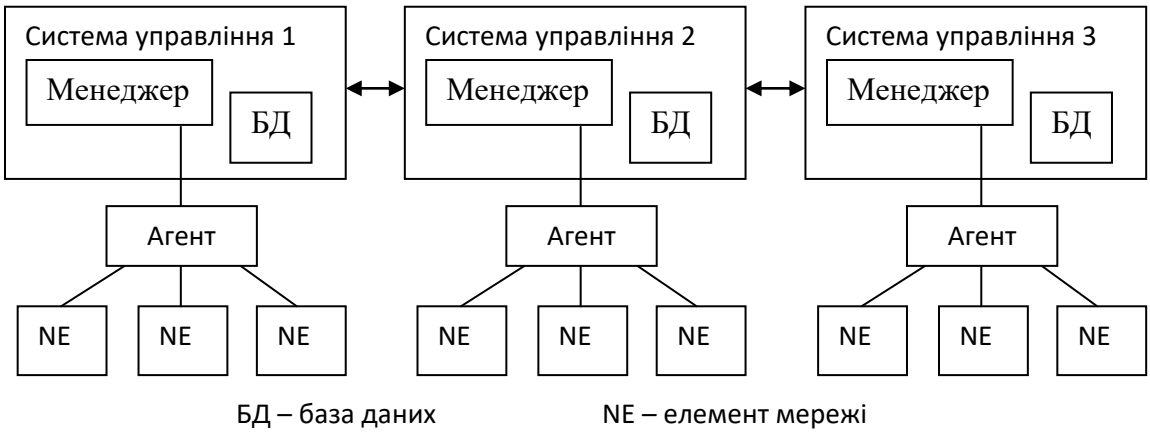

Рисунок 2.9 – Однорангові зв'язки між менеджерами

У разі однорангових зв'язків кожен менеджер управляє своєю частиною мережі на основі інформації, що отримується від агентів, що розташовані нижче. Центральний менеджер відсутній. Координація роботи менеджерів досягається за рахунок обміну інформацією між базами даних кожного менеджера.

Однорангова побудова системи управління сьогодні вважається неефективною і застарілою. Це обумовлено тим, що елементарні системи управління будувалися як монолітні системи, які не були орієнтовані

на модульну системи (наприклад, багато систем управління, розроблені виробниками устаткування, не підтримують стандартні інтерфейси для взаємодії з іншими системами управління). Потім ці менеджери нижнього рівня почали об’єднуватися для створення інтегрованої системи управління мережею, але зв’язки між ними виявилося можливим створювати тільки на рівні обміну між базами даних, що є достатньо повільним. Крім того, в базах даних таких менеджерів накопичується дуже детальна інформація про керовані елементи мережі (оскільки спочатку ці менеджери розроблялися як менеджери нижнього рівня), унаслідок чого, така інформація малопридатна для координації роботи всієї мережі в цілому. Такий підхід до побудови системи управління називається підходом «знизу вверх».

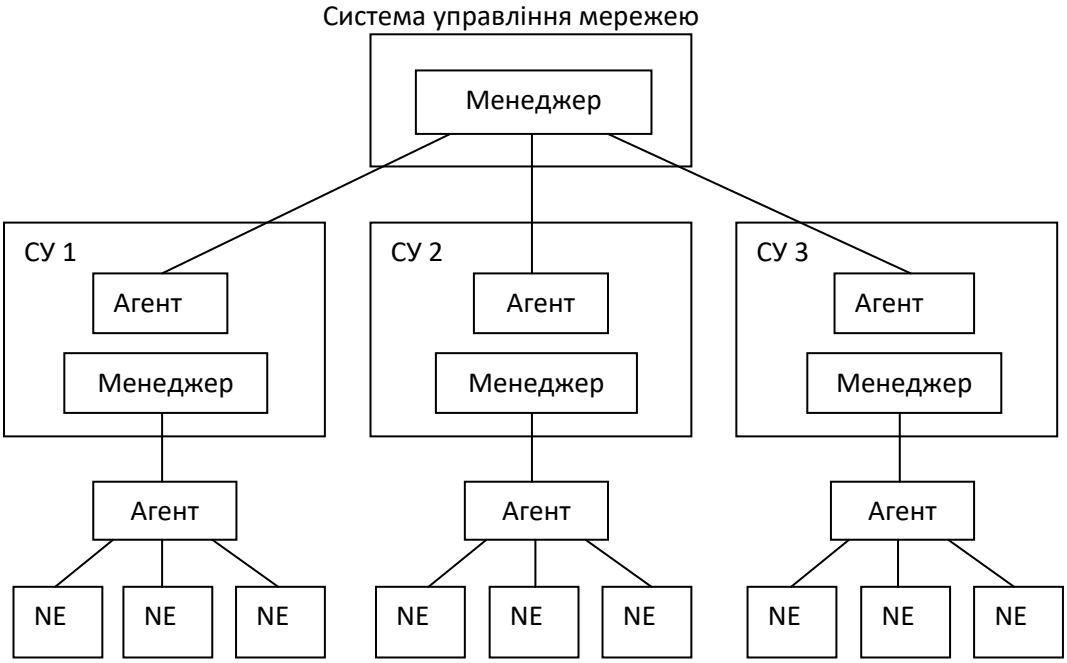

Рисунок 2.10 – Ієрархічні зв’язки між менеджерами

Набагато гнучкішою є ієрархічна побудова зв’язків між менеджерами. Кожен менеджер нижнього рівня виконує також функції агента для менеджера верхнього рівня. Такий агент працює вже з набагато більш укрупненою моделлю MIB своєї частини мережі, в якій збирається саме та інформація, яка потрібна менеджерові верхнього рівня для управління мережею в цілому. Зазвичай при розробці багаторівневої системи управляння мережею проектування починають з верхнього рівня, на якому визначається склад

інформації, потрібної від менеджерів-агентів нижчого рівня, тому такий підхід названий підходом «зверху вниз». Він скорочує об'єми інформації, циркулюючої між рівнями системи управління, і призводить до набагато ефективнішої системи управління.

Модель TMN найбільшою мірою відповідає ієрархічній архітектурі зв'язків між менеджерами, хоча відомі реалізації принципів TMN і в одноранговій архітектурі.

2.8. Платформний підхід

При побудові систем управління крупними локальними і корпоративними мережами зазвичай використовується платформний підхід, коли індивідуальні програми управління розробляються не «з нуля», а використовують служби і примітиви, що надаються спеціально розробленим для цих цілей програмним продуктом – платформою. Прикладами платформ для систем управління є такі відомі продукти, як HP OpenView, IMB/Tivoli TMN10 та інші [O:1, 2, 3].

Ці платформи створюють загальне операційне середовище для додатків системи управління точно так, як і універсальні операційні системи, такі як Unix або Windows, створюють операційне середовище для додатків будь-якого типу, таких як MS Word, Oracle і тому подібне. Платформа зазвичай включає підтримку протоколів взаємодії менеджера з агентами – SNMP і рідше CMIP, набір базових засобів для побудови менеджерів і агентів, а також засоби графічного інтерфейсу для створення консолі управління. У набір базових засобів зазвичай входять функції, необхідні для побудови карти мережі, засоби фільтрації повідомлень від агентів, засоби ведення бази даних. Набір інтерфейсних функцій платформи утворює інтерфейс прикладного програмування (API) системи управління. Користуючись цим API, розробники з третіх фірм створюють закінчені системи управління, які можуть управляти специфічним устаткуванням відповідно до п'яти основних функціональних груп.

Зазвичай платформа управління поставляється з яким-небудь універсальним менеджером, який може виконувати деякі базові функції

управління без додаткового програмування. Найчастіше до цих функцій відносяться функції побудови карти мережі (група Configuration Management), а також функції відображення стану керованих пристроїв і функції фільтрації повідомлень про помилки (група Fault Management). Наприклад, одна з найбільш популярних платформ HP OpenView поставляється з менеджером Network Node Manager, який виконує перераховані функції.

Великі компанії, які виготовляють комунікаційне устаткування, зазвичай, розробляють додаткові менеджери для найбільш популярних платформ, які виконують функції управління устаткуванням даного виробника більш повно.

Запитання для самоперевірки та контролю засвоєння знань

1. Який основний принцип покладено в основу концепції мережі управління телекомунікаціями (Telecommunication Management Network – TMN)?
2. Які основні елементи виділяють в TMN?
3. Яку функцію виконує «операційна система»?
4. Яку функцію виконує «робоча станція»?
5. Яку функцію виконує «мережа передачі даних»?
6. Чи застосовується в TMN у якості мережі передачі даних телекомунікаційна мережа, що керується?
7. Які рівні виділяють в архітектурі TMN?
8. Які об’єкти розташовані на рівні мережних елементів?
9. Які функції реалізовані на рівні управління мережними елементами?
10. Які функції реалізовані на рівні управління мережею?
11. Які функції реалізовані на рівні управління послугами?
12. Які функції реалізовані на рівні адміністративного управління (управління бізнесом)?
13. На які частини розділяють функції мережного управління?
14. Яка модель описує функціональну архітектуру системи управління мережею?
15. Що забезпечують загальні функції управління?
16. На які групи розділяють прикладні функції управління?

17. Які завдання вирішують функції управління відмовами?

18. Які завдання вирішують функції управління конфігурацією?

19. Які завдання вирішують функції обліку?

20. Які завдання вирішують функції управління продуктивністю?

21. Які завдання вирішують функції управління безпекою?

22. Як модель FCAPS пов'язана з архітектурою TMN?

23. Які основні функціональні блоки виділяють в архітектурі TMN?

24. Що являє собою блок мережного елементу?

25. Які функції виконує блок операційної системи (OSF):

26. Які функції виконує блок робочої станції (WSF)?

27. Які функції виконує блок медіатора (MF)?

28. Які функції виконує блок Q-адаптера ?

29. Який з функціональних блоків буде задіяний на усіх рівнях архітектури системи керування?

30. Що описує інформаційна архітектура TMN?

31. Як описується інформаційний обмін в інформаційній архітектурі TMN?

32. Що розуміють під керованим об'єктом?

33. Що розуміють під класом керованих об'єктів?

34. У якому вигляді представляють зв'язки між класами?

35. Якими параметрами характеризується керований об'єкт?

36. Яка схема взаємодію покладена в основу систем керування?

37. Які функції виконує менеджер системи керування?

38. Які функції виконує агент системи керування?

39. Зі скількома менеджерами/агентами можуть взаємодіяти інші агенти/менеджери системи керування?

40. Який протокол найбільш широко застосовують для організації обміну керуючою інформацією?

41. За допомогою чого менеджер може керувати різними реальними ресурсами?

42. У чому полягає відмінність використання моделі керованого ресурсу агентом і менеджером?

43. Що розуміють під inband керуванням?

44. Що розуміють під out-of-band керуванням?

45. Яка схема керування використовується в TMN?

46. Де можуть бути розташовані агенти системи керування?

47. Які основні недоліки системи керування на базі єдиного менеджера?

48. Які виділяють підходи для побудови розподілених систем керування?

49. Що характерно для однорангових систем?

50. Які основні недоліки однорангових систем?

51. Що характерно для ієрархічних систем?

52. Що розуміють під платформним підходом до побудови систем керування?

53. Які основні компоненти містять платформи для побудови систем керування?

3. ПРОТОКОЛ УПРАВЛІННЯ SNMP

3.1. Концепція SNMP-управління

SNMP – це протокол прикладного рівня, розроблений для стека TCP/IP, хоча є його реалізації і для інших стеків, наприклад IPX/SPX. Протокол SNMP використовується для отримання від мережних пристроїв інформації про їх стан, продуктивність і інші характеристики, які зберігаються в базі даних управляючої інформації (MIB – Management Information Base)) [О:1, 4, Д: 5-11].

Протокол SNMP і тісно пов'язана з ним концепція SNMP MIB були розроблені для управління маршрутизаторами Internet як тимчасове рішення. Але простота і ефективність рішення забезпечили успіх цього протоколу, і сьогодні він використовується при управлінні практично будь-якими видами устаткування і програмного забезпечення обчислювальних мереж. І хоча в області управління телекомунікаційними мережами спостерігається стійка тенденція застосування стандартів ITU-T, в які входить протокол CMIP, і тут є достатньо багато прикладів успішного використання SNMP-управління. Агенти SNMP вбудовуються в аналогові модеми, модеми ADSL, комутатори ATM та ін.

Враховуючи, що протокол SNMP спочатку розроблявся для вирішення задач управління в локальних мережах, то, у якості транспортного протоколу було обрано протокол UDP. В останній час широке розповсюдження отримала ідеологія розподіленого протокольного інтерфейсу DPI (Distributed Protocol Interface). Згідно цієї ідеології, для транспортування SNMP-запитів може використовуватись не тільки UDP, а і TCP протокол. Це дає можливість використовувати SNMP протокол не тільки в локальних мережах. Формати SNMP-DPI – запитів описані в документі RFC1592.

SNMP-протокол є прикладом системи управління, де для досягнення потрібного результату видається не команда, а здійснюється обмін інформацією [О:1, 4, Д: 5-11].

У системах управління, побудованих на основі протоколу SNMP, стандартизуються наступні елементи [О:1, 4, Д: 5-11]:

- протокол взаємодії агента і менеджера;
- мова опису моделей MIB і повідомлень SNMP – мова абстрактної синтаксичної нотації ASN.1 (стандарт ISO 8824:1987, рекомендації ITU-T X.208);
- декілька конкретних моделей MIB (MIB-I, MIB-II, RMON, RMON 2), імена об’єктів яких реєструються в дереві стандартів ISO.

Все інше віддається на відкуп розробникові системи управління.

Існують стандарти, що визначають структуру MIB, зокрема набір типів її об’єктів, їх імена і допустимі операції над цими об’єктами.

Деревоподібна структура MIB містить обов’язкові (стандартні) піддерева, а також в ній можуть знаходитися приватні (private) піддерева, що дозволяють виробнику інтелектуальних пристроїв керувати якими-небудь специфічними функціями пристрою на основі специфічних об’єктів MIB.

Агент в протоколі SNMP – це обробляючий елемент, який забезпечує менеджерам, розміщеним на станціях керування мережею, доступ до значень змінних MIB і тим самим дає їм можливість реалізовувати функції з управління і спостереження за керованими ресурсами. Основні операції з управління винесені до менеджера, а агент SNMP виконує найчастіше пасивну роль, передаючи в менеджер по його запиту значення накопичених статистичних змінних. При цьому керований пристрій працює з мінімальними витратами на підтримку керуючого протоколу. Він використовує майже всю свою обчислювальну потужність для виконання своїх основних функцій, а агент займається збором статистики і значень змінних стану пристрою і передачею їх менеджерові системи управління.

Агентами в SNMP є програмні модулі, які працюють в керованих пристроях. Агенти збирають інформацію про керовані пристрої, в яких вони працюють, і роблять цю інформацію доступною для систем управління мережами (network management systems – NMS) за допомогою протоколу SNMP (рис. 3.1).

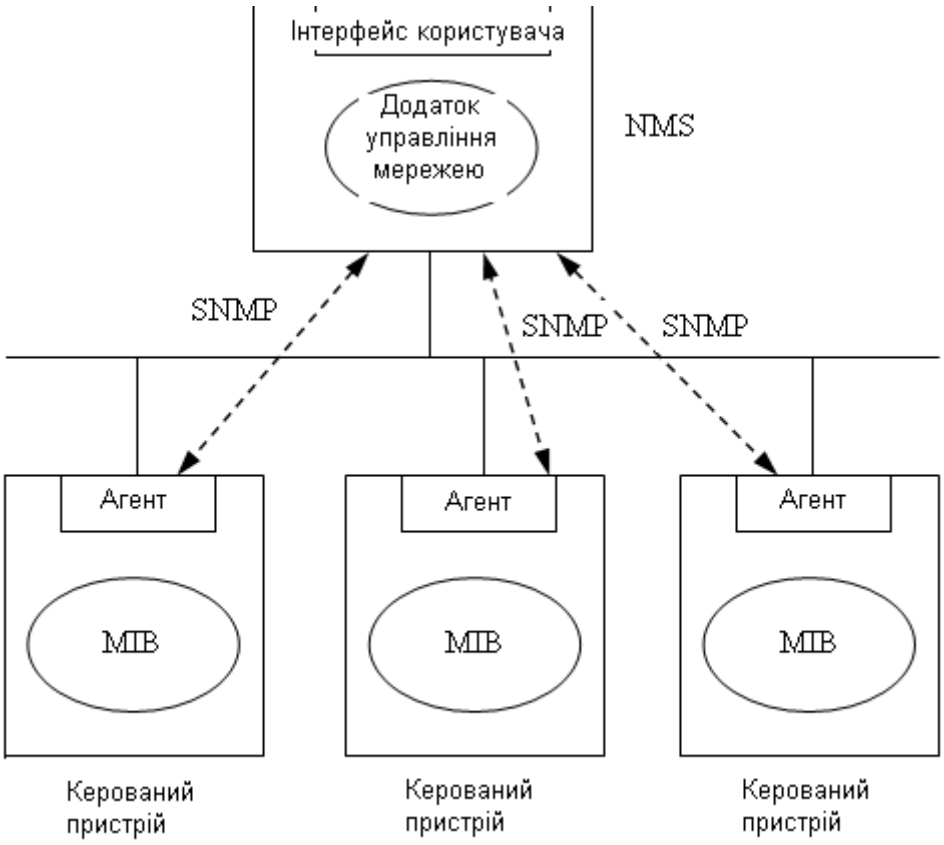

Рисунок 3.1 – Графічне представлення моделі управління

Менеджер використовує модель керованого ресурсу, щоб знати, які характеристики він може запитати у агента і якими параметрами можна управляти. Менеджер взаємодіє з агентами по стандартному протоколу (рис. 3.2).

У загальному вигляді послідовність роботи можна сформулювати наступним чином:

– суб’єкт управління формує і відправляє об’єкту управління стандартне повідомлення-запит для отримання інформації про об’єкт;
– об’єкт управління формує відповідь на запит і посилає цю відповідь суб’єкту;
– суб’єкт на підставі отриманої інформації про стан ОУ формує і відправляє йому стандартне повідомлення про зміну параметрів;
– об’єкт управління, отримавши повідомлення на зміну своїх параметрів, надсилає повідомлення-переривання і проводить відповідну власну реконфігурацію.

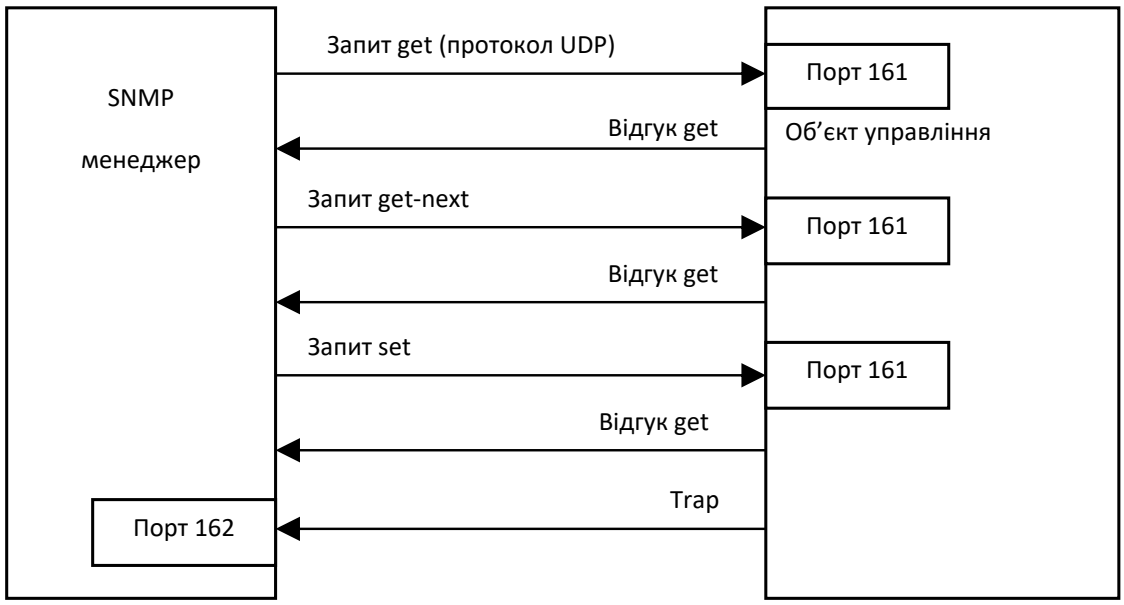

Рисунок 3.2 – Приклад схеми запитів/відгуків SNMP-протоколу

Керований пристрій може бути вузлом будь-якого типу, що знаходиться в мережі. Програми управління мають бути побудовані так, щоб мінімізувати свій вплив на продуктивність керованого пристрою.

Практично усі функції управляння реалізуються в NMS. У будь-якій керованій мережі може бути одна або більше NMS. NMS виконують прикладні програми мережного управління, які представляють інформацію управління користувачам. Інтерфейс користувача зазвичай базується на стандартизованому графічному інтерфейсі користувача (graphical user interface – GUI).

Повідомлення між керованими пристроями і NMS регулюються протоколом мережного управління. При цьому, керовані пристрої підтримують значення ряду змінних і повідомляють їх на вимогу в NMS. Наприклад, керований пристрій може відстежувати наступні параметри:

– число і стан своїх віртуальних мереж;
– число певних видів отриманих повідомлень про несправність;
– число байтів і пакетів, що входять і виходять з даного пристрою;
– максимальна довжина черги на виході (для маршрутизаторів та інших пристроїв об’єднання мереж);
– відправлені і прийняті широкомовні повідомлення;
– мережні інтерфейси, що відмовили і знов запрацювали.

Якщо NMS хоче проконтролювати який-небудь з керованих пристроїв, вона робить це шляхом відправки йому повідомлення з вказівкою про зміну значення однієї з його змінних. У загальному випадку, керовані пристрої відповідають на чотири типи команд (або ініціюють їх).

Reads – для контролю керованих пристроїв NMS вичитують змінні, підтримувані цими пристроями.

Writes – для контролю керованих пристроїв NMS записують змінні, накопичені в керованих пристроях.

Traversal operations (операції простежування). NMS використовують операції простежування, щоб визначити, які змінні підтримує керований пристрій, а потім зібрати інформацію в таблиці змінних (наприклад, в таблиці маршрутизації).

Traps – керовані пристрої використовують пастки для асинхронних повідомлень в NMS про деякі події.

Керовані пристрої, агенти і менеджери систем керування різних виробників мають відмінності в техніці представлення даних. Для узгодження синтаксису і форматів даних використовується підмножина абстрактного синтаксису, створеного для OSI, – Abstract Syntax Notation One (ASN.1) (система позначень для опису абстрактного синтаксису). ASN.1 визначає як формати пакетів, так і керовані об’єкти.

3.2. Формат SNMP-повідомлень

Повідомлення SNMP складаються з 2 частин [О: 1, 4, Д: 5-11]: імені співтовариства (*community name*) і даних (*data*). Ім’я співтовариства визначає середовище, де функціонує набір NMS, які використовують це ім’я. NMS, що належать одному співтовариству, знаходяться під одним і тим же адміністративним керівництвом.

Інформаційна частина повідомлення містить специфічну операцію SNMP (get, set і так далі) і пов’язані з нею операнди. Операнди визначають значення характеристик об’єкта, які включені в дану операцію SNMP.

Повідомлення SNMP називаються **протокольними одиницями даних** (protocol data units – PDU). Нижче наведена табл. 3.1 команд і типів PDU SNMP [О:1, 4, Д: 5-11].

Таблиця 3.1 – Команди SNMP

Команда SNMP	**Тип PDU**	**Призначення**
GET-request	0	Набути значення вказаної змінної або інформацію про стан мережного елементу
GET_next_request	1	Набути значення змінної, не знаючи точного її імені (наступний логічний ідентифікатор на дереві MIB)
SET-request	2	Привласнити змінній відповідне значення. Використовується для опису дії, яка має бути виконана
GET response	3	Відгук на GET-request, GET_next_request і SET-request. Містить також інформацію про стан (коди помилок та інші дані)
TRAP	4	Відгук мережного об’єкта на подію або на зміну стану
GetBulkRequest	5	Запит пересилки великих об’ємів даних, наприклад, таблиць
InformRequest	6	Менеджер звертає увагу партнера на певну інформацію в MIB
SNMPv3-Trap	7	Відгук на подію (тільки SNMPv3)
Report	8	Звіт

На рис. 3.3 зображений формат пакету SNMP який вкладається в UDP-датаграми (PDU операцій get/set і PDU операції trap).

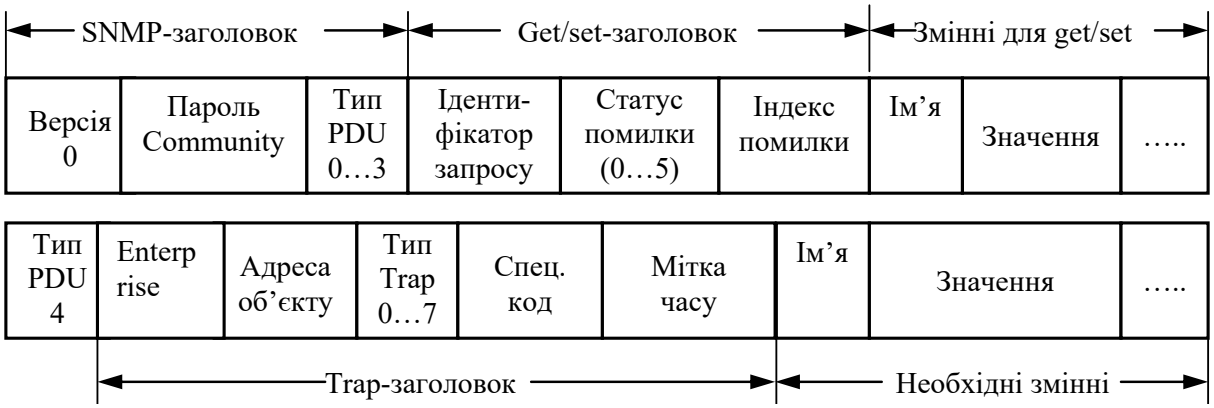

Рисунок 3.3 – Формат SNMP-повідомлень, що вкладаються в UDP-датаграми

PDU операцій get і set SNMP складаються з наступних частин.

Поле «*версія»* містить значення, яке дорівнює номеру версії SNMP мінус один.

Поле «*пароль»* (community – визначає групу доступу) містить послідовність символів, яка є перепусткою при взаємодії менеджера і об’єкта

управління. Зазвичай це поле містить 6-ти байтову строку public, що означає загальнодоступність.

REQUEST-ID (ідентифікатор запиту) – встановлює зв'язок між командами і відповідями. Для запитів GET, GET-Next та SET значення ідентифікатора запиту встановлюється менеджером і повертається об'єктом управління у відгуку GET, що дозволяє пов'язувати в пари запити та відгуки.

Error-status (статус помилки) – указує помилку і її тип. Коди помилок наведені в табл. 3.2.

Error-index (індекс помилки) – встановлює зв'язок між помилкою і конкретною реалізацією об'єкта. Якщо сталася помилка, поле «*індекс помилки»* вказує, до якої із змінних це відноситься. *Error index* є покажчиком змінної та встановлюється відмінним від нуля об'єктом управління для помилок badvalue, Readonly і nosuchname.

Variable bindings (змінні прив'язки) – складаються з даних SNMP PDU. Змінні прив'язки встановлюють зв'язок між конкретними змінними і їх поточними значеннями.

PDU пастки відрізняються від PDU інших операцій. Вони складаються з таких частин.

Enterprise (наочна область) – ідентифікує тип об'єкта, що генерує дану пастку.

Agent address (адреса агента) – містить адресу об'єкта, що генерує дану пастку.

Generic trap type (загальний тип пастки) – вказує на загальний тип пастки (табл. 3.3).

Specific trap code (специфічний код пастки) – вказує на специфічний код пастки.

Time stamp (часова мітка) – містить значення, яке дорівнює числу сотих часток секунди (тіків) з моменту останньої ініціалізації об'єкта керування до генерації даної пастки.

Variable bindings (змінні прив'язки) – перелік змінних, що містять інформацію про пастку.

Таблиця 3.2 – Коди помилок

Статус помилки	**Ім'я помилки**	**Опис**
0	Noerror	Все в порядку
1	Toobig	Об'єкт не може укласти відгук в одне повідомлення
2	Nosuchname	В операції вказана невідома змінна
3	badvalue	У команді set використана неприпустима величина або невірний синтаксис
4	Readonly	Менеджер спробував змінити константу
5	Generr	Інші помилки

Таблиця типів команди Trap представлена нижче (табл. 3.3).

Таблиця 3.3 – Коди Trap (Generic trap type)

Тип TRAP	**Ім'я TRAP**	**Опис**
0	Coldstart	Встановлення початкового стану об'єкта
1	Warmstart	Відновлення початкового стану об'єкта
2	Linkdown	Інтерфейс виключився. Перша змінна в повідомленні ідентифікує інтерфейс
3	Linkup	Інтерфейс включився. Перша змінна в повідомленні ідентифікує інтерфейс
4	Authenticationfailure	Від менеджера отримано snmp-повідомлення з невірним паролем (community)
5	EGPneighborloss	EGP-партнер відключився. Перша змінна в повідомленні визначає IP-адресу партнера
6	Entrprisespecific	Інформація про Trap міститься в полі «спеціальний код»

Приклад заголовка snmp-запиту (зображені поля утворюють єдиний масив) наведено на рис. 3.4.

Поле «*Прапор»* = 0x30 є ознакою ASN.1-заголовка. Коди LN – представляють собою довжини полів, що починаються з байта, який слідує за кодом довжини, аж до кінця повідомлення-запиту (n – номер поля довжини), якщо не обумовлено інше. Так L1 – довжина пакета-запиту, починаючи з T1 і до кінця пакету, а L3 – довжина поля пароля. Субполя TN – поля типу наступного за ними субполя запиту. Так T1 = 2 означає, що поле характеризується цілим

числом, а T2 = 4 вказує на те, що далі слідує пароль (поле «community», у наведеному прикладі = public). Цифри під рисунками означають типові значення субполів. Код 0xA – є ознакою GET-запиту, за ним слідує поле коду PDU. Блок субполів «*ідентифікатор запиту*» служить для тих самих цілей, що й інші ідентифікатори – для визначення пари запит-відгук. Власне ідентифікатор запиту може займати один або два байти, що визначається значенням Liз (із – ідентифікатор запиту).

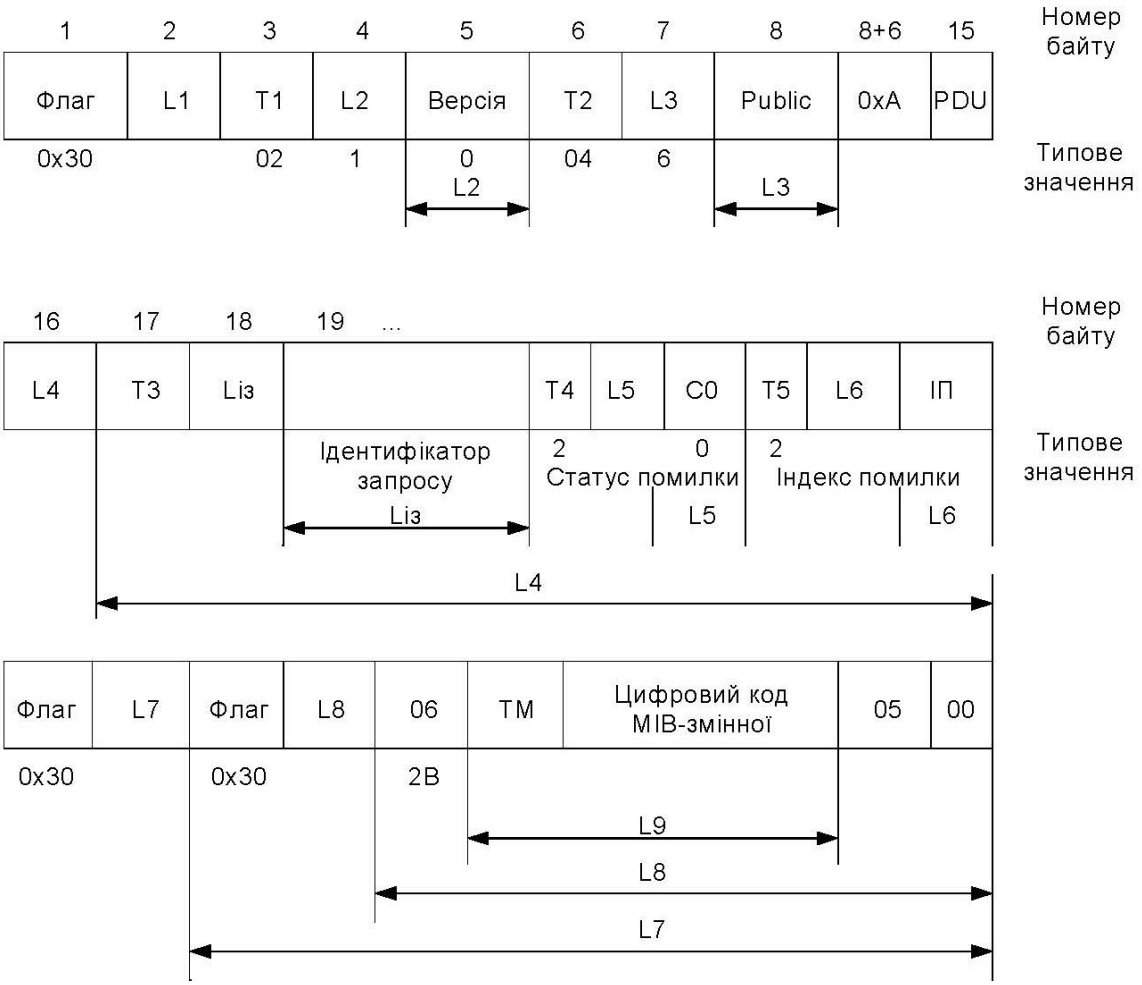

Рисунок 3.4 – Формат заголовка SNMP-запиту

CO – статус помилки (CO = 0 – помилки немає); TM – тип MIB-змінної (у наведеному прикладі = 0x2B); ІП – індекс помилки. Цифровий код MIB-змінної відображається послідовністю цифрових субполів, що характеризують змінну. Наприклад: змінна 1.3.6.1.2.1.5 (в символьному вираженні iso.org.dod.internet.mgmt.mib.icmp) характеризується послідовністю кодів 0x2B 0x06 0x01 0x02 0x01 0x05 0x00.

3.3. Структура керуючої інформації (SMI)

Структура керуючої інформації (Structure of Management Information, SMI) визначає допустимі в базі керуючої інформації типи даних і способи їх подання [О: 1, 4, Д: 5-11]. Крім того, вона визначає ієрархічну структуру імен, щоб керовані об’єкти мали унікальні однозначні імена. SMI представляє свого роду підмножину ASN.1 – вона використовує частину типів даних цього стандарту та вводить кілька своїх типів даних.

SMI визначає наступні типи інформації.

Network addresses (Мережні адреси) – являють собою яку-небудь адресу з конкретного сімейства протоколів. Прикладом мережних адрес є 32-бітові адреси IPv4.

Counters (Лічильники) – цілі числа, які монотонно збільшуються до тих пір, поки не досягнуть максимального значення, після чого вони скидаються до нуля. Прикладом лічильника є загальне число байтів, прийнятих інтерфейсом.

Gauges (Вимірювач, розмір) – цілі числа, які можуть збільшуватися або зменшуватися. При досягненні максимального значення вимірювач більше не змінюється. Прикладом вимірювача є довжина черги, що складається з вихідних пакетів.

Ticks (Тіки) – соті частки секунди, що минули після якої-небудь події. Прикладом є проміжок часу, що минув після входження інтерфейсу в свій поточний стан.

Opaque (Непрозорий) – довільне кодування. Використовується для передачі довільних інформаційних послідовностей, які не визначені в SMI.

Для того, щоб агенти та менеджери могли керувати об’єктами в мережі з безліччю пристроїв і протоколів різних постачальників, змінні, які характеризують об’єкти, повинні бути описані, а їх значення стандартним чином закодовані для передачі по мережі.

Об’єкти бази керуючої інформації зазвичай мають шість атрибутів. Як правило, це:

– ***ім’я***, наприклад, ifInErrors або tcpAttemptFails;

– ***ідентифікатор об'єкта*** в точково-десятковій нотації виду 1.3.6.1.2.1.2.2.1.1.4;
– ***поле синтаксису*** для вибору одного з декількох можливих типів даних (Integer, Ipaddress або Counter);
– ***поле методу доступу*** – «недоступний», «тільки читання», «читання-запис» і «тільки запис»;
– ***поле статусу*** – «обов'язковий», «необов'язковий» або «вийшов із вживання»;
– ***текстовий опис об'єкта***.

SMI задає правила опису об'єктів. Всі ці абстрактні правила і зарезервовані слова дозволяють отримати специфікації, зрозумілі людині. Об'єкти MIB мають статичну природу. Вони компілюються з текстів файлів з описом у двійкову форму, яку агенти та менеджери і завантажують. Використовуючи SMI, виробник може дати власне визначення об'єкта управління (наприклад, PacketsContainingWordSpam), пропустити текст через стандартний компілятор MIB і створити, таким чином, код, який виконується. Цей код можна встановити на агенти з належним апаратним і програмним забезпеченням для підрахунку кількості пакетів, які містять слово «spam».

3.4. Примітиви протоколу SNMP

SNMP – це протокол типу «запит-відповідь», тобто на кожен запит, що надійшов від менеджера, агент повинен надіслати відповідь. Особливістю протоколу є його надзвичайна простота – він включає в себе лише кілька команд (табл. 3.1) [О: 1, 4, Д: 5-11].

- Команда **Get-request** використовується менеджером для отримання від агента значення якого-небудь об'єкта за його ім'ям.
- Команда **GetNext-Request** використовується менеджером для отримання значення наступного об'єкта (без зазначення його імені) при послідовному перегляді таблиці об'єктів.

- За допомогою команди **Get – Response** агент SNMP передає менеджеру відповідь на команди **Get-Request** або **GetNext-Request**.
- Команда **Set** використовується менеджером для зміни значення якогось об’єкта. За допомогою команди **Set** відбувається власне керування пристроєм. Агент повинен розуміти сенс значень об’єкта, що використовується для керування пристроєм, і на підставі цих значень виконувати реальний управляючий вплив – відключити порт, приписати порт певної VLAN тощо. Команда **Set** використовується також для встановлення умови, при виконанні якої агент SNMP повинен надіслати менеджеру відповідне повідомлення. Може бути визначена реакція на такі події, як ініціалізація агента, рестарт агента, обрив зв’язку, відновлення зв’язку, невірна автентифікація і втрата найближчого маршрутизатора. Якщо відбувається будь-яке з цих подій, то агент ініціалізує переривання.
- Команда **Trap** використовується агентом для повідомлення менеджеру про виникнення особливої ситуації.

Версія SNMP v.2 додає до цього набору команду **GetBulk**, яка дозволяє менеджеру отримати кілька значень змінних за один запит.

Повідомлення, що містять такі блоки даних протоколу як, GetRequest-PDU, GetNextRequest-PDU, SetRequest-PDU, відправляються від менеджера до агента, а повідомлення, що містять GetResponse-PDU і Trap-PDU, відправляються від агента до менеджера. Менеджер відправляє свої три запити на UDP порт 161. Агент відправляє «пастки» (trap) на UDP порт 162. Оскільки використовуються два різних порти, одна система може виступати в ролі менеджера та агента одночасно.

3.5. Команди SNMP Trap

Як було заначено вище, команда Trap використовується агентом для повідомлення менеджеру про виникнення особливої ситуації. Існують основні (general) і промислові (enterprise) переривання. Основні повідомлення обумовлені в RFC протоколів SNMPv1, SNMPv2. Згідно RFC1215 «Угода

по визначенню trap-повідомлень для використання з SNMP» основними перериваннями є (табл. 3.3) [О:1, 4, Д: 11]:

coldStart (тип TRAP – 0) – повідомляє, що посилаючий протокольний об’єкт переініціалізував сам себе таким чином, що конфігурація агента або реалізація протокольного об’єкта могли бути змінені;

warmStart (тип TRAP-1) – повідомляє, що посилаючий протокольний об’єкт переініціалізував себе таким чином, що ні конфігурація агента, ні реалізація протокольного об’єкта не змінені;

linkDown (тип TRAP-2) – повідомляє, що посилаючий протокольний об’єкт розпізнав несправність в одному з комунікаційних зв’язків, представлених в конфігурації агента;

linkUp (тип TRAP-3) – повідомляє, що посилаючий протокольний об’єкт розпізнав відновлення в одному з комунікаційних зв’язків, представлених в конфігурації агента;

authenticationFailure (тип TRAP-4) – повідомляє, що посилаючий протокольний об’єкт є одержувачем протокольного повідомлення з некоректною автентифікацією;

EgpNeighborLoss (тип TRAP-5) – повідомляє, що EGP (Exterior Gataway Protocol) сусід-маршрутизатор позначений як вимкнутий і отримання повідомлень від нього припинено.

Промислові переривання мають структуру повідомлення ідентичну основним. При цьому виробник сам визначає назви і типи змінних в повідомленні.

Найбільший інтерес для адміністратора мережі представляють переривання агентів linkDown, linkUp і authenticationFailure. Перші два дозволяють швидко реагувати на фізичні обриви комунікаційних ліній, третє – на спробу неавторизованого доступу (або некоректних установок пакетних фільтрів, програм по збору SNMP інформації), що також є одним із складових елементів моніторингу безпеки мережних пристроїв.

Великі виробники активного мережного обладнання створюють досить великий набір промислових SNMP переривань, які можуть надати

адміністратору додаткові зручності при підтримці не тільки працездатності, але й безпеки мережі. Зазвичай всі переривання описані в документації на апаратуру або на web-сайтах виробника.

3.6. Структура SNMP MIB

На сьогодні існує кілька стандартів на бази даних керуючої інформації для протоколу SNMP. Основними є стандарти MIB-I та MIB-II, а також версія бази даних для віддаленого керування RMON MIB [О:1, 4, Д: 5-12]. Крім цього існують стандарти для спеціальних пристроїв MIB конкретного типу (наприклад, MIB для комутаторів або MIB для маршрутизаторів), а також приватні MIB конкретних фірм-виробників обладнання.

Згідно з нормативами MIB визначено 185 об'єктів, керуюча інформація про які поділяється на 10 категорій (табл. 3.4) (MIB-I – 114 об'єктів і 8 категорій).

Всі керовані об'єкти містяться в інформаційній базі управління (Management Information Base – MIB), яка фактично є базою даних об'єктів. MIB можна зобразити у вигляді абстрактного дерева, листями якого є окремі інформаційні елементи. Ідентифікатори об'єктів унікальним чином ідентифікують об'єкти MIB цього дерева. Ідентифікатори об'єктів вищого рівня MIB призначаються Міжнародною Електротехнічною Комісією ISO (ISO IEC). Ідентифікатори об'єктів нижчого рівня призначаються організаціями, до яких ці об'єкти відносяться. Зображення кореневої і декількох найбільш великих гілок дерева MIB наведено на рис. 3.5.

Таблиця 3.4 – MIB категорії керуючої інформації

MIB-категорія	**Опис**	**Код**
system	Операційна система ЕОП або маршрутизатора	1
Interfaces	Мережний інтерфейс	2
addr.trans	Перетворення адреси (наприклад, за допомогою ARP)	3
IP	Дані, які відносяться до IP- протоколу	4
ICMP	Дані, які відносяться до ICMP-протоколу	5
TCP	Дані, які відносяться до TCP-протоколу	6
UDP	Дані, які відносяться до UDP-протоколу	7
EGP	Дані, які відносяться до EGP-протоколу	8
SNMP	Дані, які відносяться до SNMP-протоколу	11

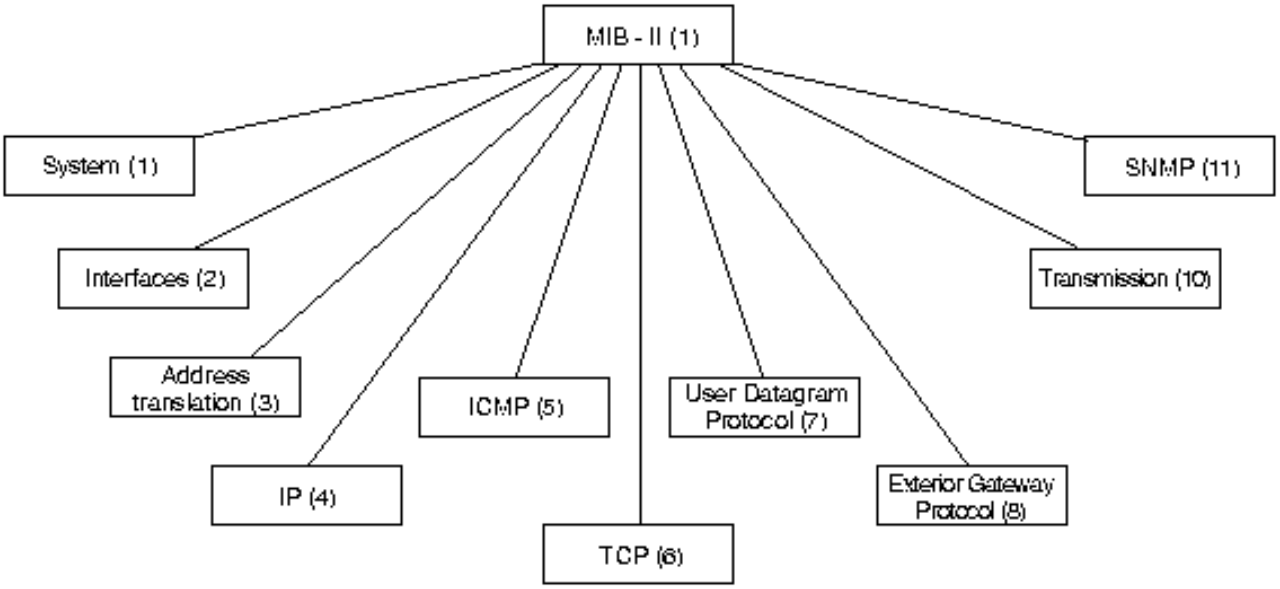

Рисунок 3.5 – Коренева і декілька найбільш великих гілок дерева MIB

Первісна специфікація MIB-I визначала тільки операції читання значень змінних. Операції зміни або встановлення значень об’єкта є частиною специфікацій MIB-II. Дерево MIB розширюється завдяки експериментальним і приватним гілкам. Наприклад, виробники можуть визначати свої власні гілки для включення реалізацій своїх виробів. В даний час вся робота по стандартизації ведеться на експериментальній гілці.

Версія MIB-I (RFC 1156) визначає 114 об’єктів, які поділяються на 8 груп.

System – загальні дані про пристрій (наприклад, ідентифікатор розробника, час останньої ініціалізації системи).

Interfaces – параметри мережних інтерфейсів пристрою (наприклад, їх кількість, типи, швидкості обміну, максимальний розмір пакету).

Address Translation Table – опис відповідності між мережними і фізичними адресами (наприклад, по протоколу ARP).

Internet Protocol – дані, що відносяться до протоколу IP (IP-адреси шлюзів, хостів, статистика про IP-пакети).

ICMP – дані, що відносяться до протоколу обміну керуючими повідомленнями ICMP.

TCP – дані, що відносяться до протоколу TCP.

UDP – дані, що відносяться до протоколу UDP (число переданих, прийнятих і помилкових UPD-датаграм).

EGP – дані, що відносяться до протоколу обміну маршрутної інформації EGP (число прийнятих з помилками і без помилок повідомлень).

З цього переліку груп змінних видно, що стандарт MIB-I розроблявся з орієнтацією на управління маршрутизаторами, що підтримують протоколи стека TCP/IP.

У версії MIB-II (RFC 1213), прийнятої в 1992 році, було істотно (до 185) розширено набір стандартних об'єктів, а кількість груп збільшилася до 10. База MIB-II є універсальною. Вона містить дані про сам пристрій та всі його мережні інтерфейси. Ці дані розділені на групи, які включають інформацію про різні аспекти мережного інтерфейсу.

Transmission – об'єднує підбази MIB, в яких зберігається інформація, що відноситься до конкретних мережних технологій, наприклад, FastEthernt або Token Ring.

SNMP – містить інформацію протоколу SNMP, яка використовується для управління даним пристроєм.

На рис. 3.6 наведений приклад деревоподібної структури бази об'єктів MIB-II. На ньому показані дві з 10 можливих груп об'єктів – **System** (імена об'єктів починаються з префікса Sys) та **Interfaces** (префікс If).

Об'єкт *SysUpTime* містить значення тривалості часу роботи системи з моменту останнього перезавантаження, об'єкт SysObjectID – ідентифікатор пристрою (наприклад, маршрутизатора). Об'єкт *ifNumber* визначає кількість мережних інтерфейсів пристрою, а об'єкт *ifEntry* є вершиною піддерева і описує один з конкретних інтерфейсів пристрою. Об'єкти *ifType* та *ifAdminStatus*, що входять в це піддерево визначають відповідно тип і стан одного з інтерфейсів, у даному випадку інтерфейсу *Ethernet*.

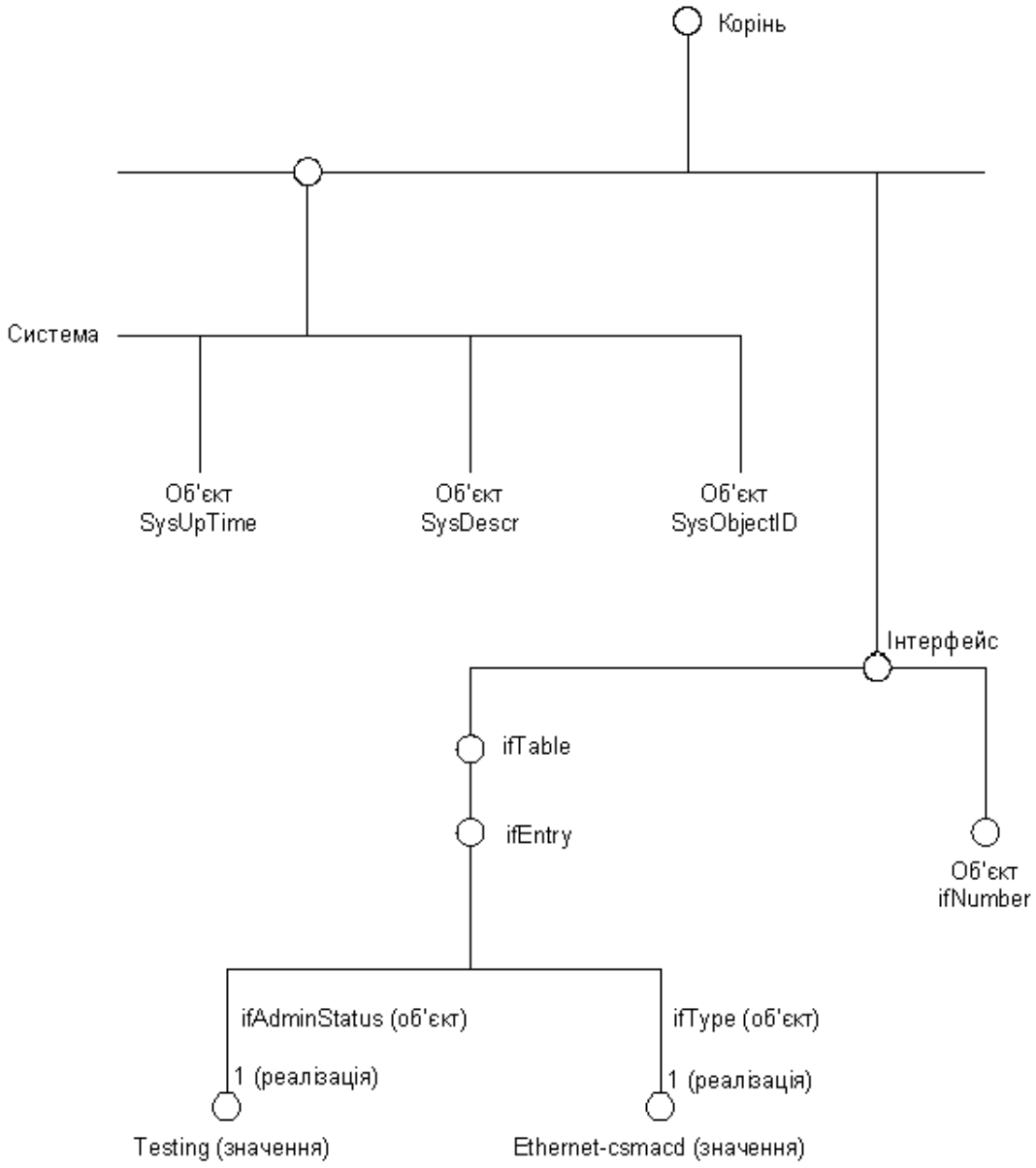

Рисунок 3.6 – Стандартне дерево MIB-II (фрагмент)

До числа об’єктів, які описують кожен конкретний інтерфейс пристрою, включені наступні:

ifType – тип протоколу, який підтримує інтерфейс. Цей об’єкт приймає значення всіх стандартних протоколів канального рівня, наприклад rfc877-x25, Ethernet-csmacd, iso88023-csmacd, iso88024-tokenBus, iso88025-TokenRing тощо;

ifMtu – максимальний розмір пакета мережного рівня, який можна переслати через цей інтерфейс;

ifSpeed – пропускна здатність інтерфейсу в бітах в секунду (100 для Fast Ethernet);

ifPhysAddress – фізична адреса інтерфейсу, для Fast Ethernet цією адресою буде MAC-адреса;

ifAdminStatus – бажаний статус інтерфейсу:

up – готовий передавати пакети;

down – не готовий передавати пакети;

testing – знаходиться в тестовому режимі;

ifOperStatus – фактичний поточний статус інтерфейсу. Має ті ж значення, що й ifAdminStatus;

ifInOctets – загальна кількість байт, прийнятих даним інтерфейсом, включаючи службові, з моменту останньої ініціалізації SNMP-агента;

ifInUcastPkts – кількість пакетів з індивідуальною адресою інтерфейсу, переданих протоколу верхнього рівня;

ifInNUcastPkts – кількість пакетів з широкомовною або груповою адресою інтерфейсу, переданих протоколу верхнього рівня;

ifInDiscards – кількість пакетів, які були прийняті інтерфейсом, виявилися коректними, але не були передані протоколу верхнього рівня, швидше за все через переповнення буфера пакетів або ж з іншої причини;

ifInErrors – кількість пакетів, які були прийняті інтерфейсом, але не були передані протоколу верхнього рівня через виявлені в них помилки.

Крім об’єктів, що описують статистику за вхідними пакетами, є аналогічні об’єкти, але які відносяться до вихідних пакетів.

Недоліком бази MIB-II є те, що вона не дає детальної статистики за характерними помилками кадрів Ethernet. Крім цього, вона не відображає зміну характеристик у часі, що часто цікавить мережного адміністратора. Ці обмеження були згодом зняті новим стандартом на MIB – **RMON MIB**, який спеціально орієнтований на збір детальної статистики по протоколу Ethernet, до того ж з підтримкою такої важливої функції, як побудова агентом статистичних залежностей характеристик від часу.

3.7. Система RMON

Система RMON (Remote Network Monitoring – віддалений моніторинг мережі) розроблена Internet Engineering Task Force (IETF), як стандарт керування для спостереження за мережею в цілому, а не за окремим мережним інтерфейсом, що характерно для MIB-II. Ця система визначена у документі RFC1757, випущеному у лютому 1995 року, і є доповненням до RFC1271.

У RMON передбачено 9 об'єктних груп (табл. 3.5) [О: 1, 4, Д: 12].

Таблиця 3.5 – Функціональні групи RMON

Група	**Призначення**
statistics	Таблиця, яка відстежує близько 20 статистичних параметрів мережного трафіку, включаючи загальне число кадрів і кількість помилок
history	Дозволяє задати частоту і інтервали для вимірювання трафіку
alarm	Дозволяє встановити поріг і критерії, за яких агенти видають сигнал тривоги
host	Таблиця, що містить усі вузли мережі, дані з яких наводяться у мережній статистиці
hostTopN	Дозволяє створити впорядковані списки, які базуються на пікових значеннях трафіку групи ЕОП
matrix	Дві таблиці статистики трафіку між парами вузлів. Одна таблиця базується на адресах вузлів-відправників, інша – на адресах вузлів-одержувачів
filter	Дозволяє визначити конкретні характеристики кадрів в каналі. Наприлад, можна виділити TCP-трафік
packet capture	Працює спільно з групою Filter. Дозволяє визначити обсяг ресурсу пам'яті, що виділяється для запам'ятовування кадрів, які відповідають критеріям Filter
event	Дозволяє визначити набір параметрів або умов, які повинен контролювати агент. Коли умови виконуються, інформація про подію записується у спеціальний журнал

RMON MIB призначена для збору статистики про будь-який сегмент мережі в цілому, в протилежність MIB-II, яка збирає статистичні дані про конкретний мережний інтерфейс.

Дані RMON MIB збираються пристроєм, який називається зондом RMON. Зонд може бути встановлений у будь-який пристрій, що має доступ до всіх кадрів сегменту FastEthernet або Ethernet, наприклад, у комутатор. Зонд RMON періодично збирає статистичну інформацію та записує її в пам'ять. Все це відбувається незалежно від NMS, яка може сконфігурувати зонд RMON

для запису даних і більше не спілкуватися з ним до тих пір, поки не потрібно буде знову зібрати такого роду відомості. Цей прийом дозволяє суттєво зменшити об’єм трафіку, необхідного для збору статистичної інформації.

3.8. Інформаційна безпека протоколу SNMP

Розглянемо, як враховуються в SNMP основні елементи інформаційної безпеки [О:1, 4]:

– конфіденційність,

– цілісність,

– доступність,

– контроль.

У ході розвитку стандартів SNMP інфраструктура безпеки піддавалася найбільшим змінам. Для того, щоб простежити ці зміни, спочатку розглянемо вимоги безпеки і загрози, потім структуру моделей безпеки, після чого розглянемо різні версії SNMP моделі.

Вимоги та загрози безпеці.

Основними загрозами безпеці, проти яких повинна надавати захист будь-яка модель безпеки SNMP, є:

– модифікація інформації,

– маскарад,

– модифікація потоку повідомлень,

– розголошення.

Загроза модифікації інформації – це можливість того, що деяка неавторизована сутність може змінити повідомлення SNMP, згенеровані за запитом авторизованого об’єкта, на шляху їх слідування таким чином, що це призведе до неавторизованих операцій управління, включаючи фальсифікацію значення об’єкта (змінної).

Загроза маскараду – це можливість того, що операції управління, не авторизовані для деякого об’єкта, можуть бути здійснені при підстановці ідентифікатора іншого об’єкта, для якого подібні дії авторизовані.

Загроза модифікації потоку повідомлень полягає в наступному. Протокол SNMP зазвичай заснований на транспортній службі без організації з'єднання (протокол UDP). Перестановка, затримка або повторне надсилання повідомлень може відбуватися і відбувається в ході звичайної експлуатації мереж із застосуванням таких протоколів. Загроза модифікацій потоку повідомлень – це можливість того, що повідомлення можуть бути переставлені, затримані або повторно послані зі злим умислом із затримками, які перевищують затримку при нормальній експлуатації, з метою виконання неавторизованих операцій управління.

Загроза розголошення – це можливість прослуховування обмінів повідомленнями між сутностями SNMP. Захист від цієї загрози може вимагатися локальною політикою безпеки. Модель безпеки може не протидіяти такими загрозами, як, наприклад, відмова в обслуговуванні, оскільки стан мережних збоїв і помилок (нормальне для SNMP) часто не відрізнити від ситуацій відмови в обслуговуванні.

Структура моделей безпеки.

Структура моделей безпеки була описана у документі по архітектурі SNMP. Сутність SNMP містить дві підсистеми:

– підсистема безпеки,

– підсистема керування доступом.

Підсистема безпеки функціонує на рівні повідомлень SNMP. Вона відповідає за автентифікацію, шифрування і своєчасність. Документ з безпеки описує модель безпеки, загрози, проти яких вона захищає, цілі моделі безпеки, протоколи, які використовуються для досягнення цих цілей, модуль MIB, що описує відповідні структури даних. Такий модуль потрібний, в тому числі і для того, щоб віддалено конфігурувати параметри безпеки рівня повідомлень SNMP.

Підсистема керування доступом функціонує на рівні SNMP PDU. Вона відповідає за управління доступом до керованих об'єктів (змінних). Документи, що описують її, також визначають модуль MIB для віддаленого налаштування параметрів.

Моделі безпеки.

Існують такі основні моделі безпеки і відповідні їм інфраструктури [О: 1, 4, Д: 13-16].

1. SNMPv1
   - SNMPv1 – Community-based Security Model
2. SNMPv2
   - SNMPv2p – Party-based Security Model
   - SNMPv2c – Community-based Security Model
   - SNMPv2u – User-based Security Model
3. SNMPv3
   - SNMPv3 USM User-based Security Model

Модель безпеки на основі спільнот (Community-based Security Model) була першою, найпростішою і самою вразливою. Вона передбачає лише автентифікацію на основі «рядка спільноти», фактично – пароля, переданого по мережі в тілі повідомлення SNMP у відкритому тексті. Ця модель безпеки невзмозі боротися з жодною із загроз інформаційної безпеки в SNMP. Тим не менш, вона часто використовується до сих пір в зв'язку зі своєю простотою, а також завдяки наявності зовнішніх, не пов'язаних з SNMP систем безпеки, наприклад, міжмережних екранів.

Модель безпеки на основі сторін (Party-based Security Model) передбачає введення поняття сторони. **Сторона** – це віртуальне оточення виконання, в якому набір допустимих операцій адміністративно обмежений. Сутність SNMP при обробці повідомлення діє як сторона і обмежена операціями, визначеними для цієї сторони. Сторона визначається наступними параметрами:

- унікальний ідентифікатор сторони,
- логічна мережна адреса,
- протокол автентифікації і параметри, необхідні для автентифікації всіх повідомлень сторони,
- протокол шифрування і параметри, необхідні для шифрування всіх повідомлень сторони.

Можуть використовуватися різні алгоритми для протоколів автентифікації та шифрування. Зазвичай в якості алгоритму для протоколу автентифікації використовують хеш-функцію Message Digest 5 (MD5), а для протоколу шифрування – алгоритм Data Encryption Standard (DES) в режимі Cipher Block Chaining (CBC). При використанні відповідних протоколів автентифікації і шифрування, модель успішно протидіє описаним вище загрозам безпеки SNMP. Дана модель безпеки не була широко прийнятою, оскільки вона багатьом здалася занадто складною і заплутаною.

Модель безпеки на основі користувачів (User-based Security Model) вводить поняття користувача, від імені якого діє сутність SNMP. Цей користувач характеризується ім'ям користувача, яке використовується протоколами автентифікації та шифрування, а також закритим ключем автентифікації та шифрування. Автентифікація та шифрування є необов'язковими. Їх наявність описується одним з параметрів моделі. Модель безпеки багато в чому схожа на модель на основі сторін, але вона спрощує ідентифікацію користувачів, розподіл ключів і протокольні операції.

3.9. Версії протоколу SNMP

Стандарти SNMP описують аспекти, відображені нижче [О: 1, 4, Д: 5-16].

1. Загальна інформація.
2. Обробка повідомлень (SNMP Messages):
   – прив'язки до транспорту,
   – розбір та диспетчеризація повідомлень,
   – безпека (автентифікація, шифрування, своєчасність).
3. Обробка протокольних блоків даних (SNMP PDUs):
   – операції протоколу,
   – додатки SNMP,
   – управління доступом.
4. Інформаційна модель:
   – структура керуючої інформації (SMI),

– текстові конвенції,

– вирази відповідності.

5. Модулі баз керуючої інформації (MIB).

Коротко розглянемо відмінності між версіями SNMP та документи, що визначають ці версії. Зауважимо, що тут наведено перші версії RFC кожної з версій SNMP, тобто майже всі з них на сьогоднішній день заміщені більш новими RFC.

SNMPv1

Перші RFC, що описують стандарти SNMP, з'явилися в 1988 році. Версія 1 піддалася критиці за її посередню модель безпеки на основі спільнот. У той час безпека в Інтернеті не входила в коло першочергових завдань робочих груп IETF.

SNMPv2

Версія 2 відома також як Party-based SNMPv2 або SNMPv2p, не отримала широкого розповсюдження через серйозні розбіжності з приводу інфраструктури безпеки в стандарті.

SNMPv2 поліпшує версію 1 в області швидкодії, безпеки, конфіденційності і взаємодій «менеджер- менеджер». Він представив новий тип PDU *Get-Bulk-Request*, альтернативу *Get-Next-Request* для отримання великих об'ємів інформації за допомогою одного запиту. Тим не менш, нова система безпеки на основі сторін виглядала для багатьох як занадто складна і не була широко визнана.

SNMPv2c

Community-based SNMPv2, або SNMPv2c, представив SNMPv2 без нової моделі безпеки версії 2. Замість неї пропонувалося використовувати стару модель безпеки версії 1 на основі спільнот. Відповідну пропозицію RFC було прийнято тільки як чернетку стандарту, однак це стало *де факто* стандартом SNMPv2. Безпека SNMP знову опинилася невирішеним питанням.

SNMPv2u

User-based SNMPv2, або SNMPv2u, є компромісом між незахищеністю SNMPv1 та надмірною складністю SNMPv2p. Запропонована модель безпеки на основі користувачів була покладена в основу SNMPv3.

Друга версія SNMP додає до набору команд SNMPv1 команду get-bulk, яка дозволяє менеджеру отримати кілька значень змінних за один запит.

Фундаментальним нововведенням в SNMPv2 є те, що елемент адміністрування мережі може працювати в якості менеджера, агента чи менеджера та агента одночасно. Дана концепція дає можливість користувачам застосовувати SNMP в ієрархічній структурі, в якій локальні менеджери звітують перед менеджерами середньої ланки, які, в свою чергу, контролюються менеджером вищого рівня. Чимало місця приділяється проблемам захищеності SNMP, мабуть, самої вразливої точки протоколу.

SNMPv3

SNMPv3 нарешті вирішив проблеми з безпекою способом, який багато хто вважають прийнятним. Версія 3 SNMP прийнята IETF як стандарт Інтернету (IETF STD 62). Майже всі попередні RFC визнані застарілими.

На особливу увагу заслуговують такі особливості специфікацій SNMPv3.

Модульність. Мається на увазі модульність архітектури, як технічних рішень, так і специфікацій SNMPv3. Модульність дозволяє поєднувати в рамках одного рішення компоненти від різних постачальників, робити поступові модернізації та розвивати процес стандартизації, що гарантує живучість SNMP.

Масштабованість. Підтримуються конфігурації по суті довільного масштабу, причому вартість системи управління виявляється пропорційною цьому розміру.

Інформаційна безпека. Передбачається захист керуючих повідомлень і розмежування доступу до керуючої інформації. У той час як широко використовувана SNMPv1 і менш поширена SNMPv2 реалізації протоколу SNMP, по суті, не приділяли особливої уваги питанням інформаційної безпеки управління, специфікації SNMPv3 включають модель безпеки, яка передбачає заходи захисту проти наступних можливих загроз:

– модифікація інформації в процесі її передачі;

– «маскарад» як засіб неавторизованого виконання керуючих дій;

– модифікація потоку повідомлень (до рівня, що перевищує звичайні відхилення, можливі при використанні датаграмних транспортних протоколів);
– несанкціоноване ознайомлення з повідомленнями.

SNMPv3 не передбачає спеціальних засобів захисту проти атак на доступність та аналізу трафіку, оскільки вважається, що в багатьох випадках прояви атак на доступність не відрізняються від відмов в мережі, з якими будь-який керуючий протокол повинен вміти справлятися. Що стосується аналізу керуючого трафіку, то в ньому немає великого сенсу, оскільки характер подібного трафіку є свідомо передбачуваним. Підсистема безпеки SNMPv3 займається захистом керуючих повідомлень і забезпечує:

– контроль цілісності повідомлень;
– автентифікацію джерела даних;
– шифрування / дешифрування, якщо це передбачено політикою безпеки.

Крім того, за рахунок кешування повідомлень і встановлення часових меж досягається частковий захист від дублювання, перевпорядкування і крадіжки повідомлень.

Для контролю цілісності та автентифікації джерела передбачаються хеш-функції, які обчислюються на підставі алгоритмів MD5 і SHA. Стандартним засобом шифрування є CBC-DES.

Модель безпеки включає і поняття розмежування доступу до керуючої інформації. Ця ідея означає видачу кожному суб’єкту свого представлення (View) даних, а також підмножини керуючої інформації, що задається специфікаціями MIB. Таким чином, суб’єкти бачать тільки те, до чого вони мають доступ. Більш того, при виконанні операцій читання, запису і видачі повідомлень використовуються, взагалі кажучи, різні уявлення, так що розмежування доступу може бути досить тонким. Відповідність між суб’єктами і видимими уявленнями встановлює політика безпеки.

Відзначимо, що специфікації SNMPv3, згідно з концепцією модульності, розраховані на одночасну підтримку декількох моделей безпеки

та розмежування доступу. Це, звичайно ж, набагато більш потужний засіб налаштування, ніж підтримка різних криптографічних алгоритмів.

У SNMPv3 передбачено п’ять стандартних керуючих сервісів:

– генерація команд;
– прийом повідомлень;
– обробка команд;
– породження повідомлень;
– довірене перенаправлення.

В залежності від підтримуваних сервісів можна виділити кілька базових типів програмної конфігурації у вузлах мережі, керованої по протоколу SNMPv3:

– мінімальний агент (підтримуються обробка команд і породження повідомлень);
– довірений агент (підтримується довірене перенаправлення);
– мінімальний менеджер (генерація команд і прийом повідомлень з можливістю управління з командного рядка);
– менеджер середнього рівня або дуальний вузол, що підтримує додатки мінімального агента і мінімального менеджера;
– керуюча станція (менеджер верхнього рівня), що підтримує додатки мінімального менеджера і, можливо, інші засоби, призначені для керування великою кількістю вузлів.

Наведене чітке розмежування функціональності – ще один елемент модульної архітектури SNMPv3, засіб забезпечення простоти цієї архітектури.

3.10. Недоліки протоколу SNMP

Головний недолік протоколу SNMP – це відсутність засобів взаємної автентифікації агентів і менеджерів [О:1, 4]. Єдиний засіб ідентифікації – «рядок спільноти» – «community string». Рядок передається у відкритій формі (до 3-ої версії протоколу) в повідомленні SNMP і служить основою для поділу агентів і менеджерів на «спільноти», так що агент взаємодіє лише з тими менеджерами, які вказують у полі community string той самий символьний

рядок, який зберігається в пам'яті агента. Це не є способом автентифікації, а являє собою спосіб структурування агентів і менеджерів.

Робота через ненадійний протокол UDP (переважна більшість реалізації агентів SNMP) призводить до втрат аварійних повідомлень (повідомлень trap) від агентів до менеджерам, що може призвести до неякісного керування.

Розробники платформ керування намагаються подолати ці недоліки. Наприклад, у платформі HP OpenView Telecom DM TMN, яка є платформою для розробки багаторівневих систем управління у відповідності зі стандартами TMN та ISO, працює нова реалізація SNMP, в якій передбачено надійний обмін повідомленнями між агентами і менеджерами за рахунок самостійної організації повторних передач повідомлень SNMP при їх втратах.

3.11. Особливості протоколу CMIP

Common Management Information Protocol – CMIP (протокол загальної керуючої інформації) – стандарт OSI протоколу керування мережею [O: 1, 4]. Він визначає деякі функції, відсутні в SNMP. В силу своєї складності CMIP має набагато менше розповсюдження і привертає набагато менший інтерес, ніж SNMP, але іноді його використання обов'язкове. У розглянутій вище моделі мережного керування OSI (OSI Management FrameWork) доступ до керуючої інформації, що зберігається в керованих об'єктах, забезпечується за допомогою елемента системи управління, який називають службою CMISE (Common Management Information Service Element). Служба CMISE побудована в архітектурі розподіленого додатка, де частину функцій виконує менеджер, а частину – агент. Взаємодія між менеджером і агентом здійснюється по протоколу CMIP. Послуги, що надаються службою CMISE, називаються послугами CMIS (Common Management Information Services).

Протокол CMIP та послуги CMIS визначені в стандартах X.710 і X.711 ITU-T.

Послуги CMIS поділяються на дві групи – послуги, ініційовані менеджером (запити), та послуги, ініційовані агентом (повідомлення).

Послуги, ініційовані менеджером, включають наступні операції:

M-CREATE – інструктує агента про необхідність створити новий екземпляр об’єкта певного класу або новий атрибут всередині екземпляра об’єкта;

M-DELETE – інструктує агента про необхідність видалення деякого екземпляра об’єкта певного класу або атрибуту всередині екземпляра об’єкта;

M-GET – інструктує агента про повернення значення деякого атрибуту певного екземпляра об’єкта;

M-SET – інструктує агента про зміну значення деякого атрибуту певного екземпляра об’єкта;

M-ACTION – інструктує агента про необхідність виконання певної дії над одним чи кількома екземпляра об’єктів.

Агент ініціює тільки одну операцію:

M-EVENT_REPORT – відправка повідомлення менеджеру.

Відмінність послуг CMIS від аналогічних послуг SNMP полягає в більшій гнучкості. Якщо запити GET і SET протоколу SNMP застосовані тільки до одного атрибуту даного об’єкта, то запити M-GET, M-SET, M-ACTION та M-DELETE можуть застосовуватися більш ніж до одного об’єкта.

3.12. Порівняння протоколів SNMP та CMIP

SNMP дозволяє будувати прості і складні системи керування, а CMIP визначає високий початковий рівень складності системи керування, так як для його роботи необхідно реалізувати ряд допоміжних служб, об’єктів і баз даних об’єктів [О: 1, 4].

CMIP розрахований на інтелектуальних агентів, які можуть по одній простій команді від менеджера виконати складну послідовність дій.

Агенти CMIP виконують більш складні функції, ніж агенти SNMP. Через це операції, які менеджеру можна виконати над агентом SNMP, носять точковий характер, що призводить до численних обмінів між менеджером і агентом.

Повідомлення агента SNMP посилаються менеджеру без підтвердження, в той час як повідомлення агента CMIP завжди передаються за допомогою надійного транспортного протоколу.

Частину проблем SNMP можна вирішити за рахунок застосування більш інтелектуальних MIB (до яких відноситься RMON MIB), але для багатьох пристроїв таких MIB немає.

CMIP краще масштабується, так як може впливати відразу на кілька об'єктів, а відповіді від агентів проходять через фільтри, що обмежують передачу керуючої інформації тільки певним агентам і менеджерам.

Запитання для самоперевірки та контролю засвоєння знань

1. На якому рівні моделі OSI працює протокол SNMP?
2. Для чого використовується протокол SNMP?
3. Який транспортний протокол використовується для передачі SNMP повідомлень?
4. Які елементи стандартизуються у системах управління на основі SNMP?
5. Що розуміють під SNMP-агентом?
6. Які функції виконує SNMP-агент?
7. Як менеджер використовує модель керованого об'єкта?
8. Які порти UDP протоколу використовують агенти і менеджери для обміну інформацією по протоколу SNMP?
9. Які основні операції виділяють у послідовності обміну інформацією по протоколу SNMP?
10. Які функції виконує система мережного управління (NMS)?
11. Наведіть приклади параметрів, які можуть підтримувати керовані пристрої.
12. Які типи команд використовуються в системах управління?
13. Для чого використовується команда READ?
14. Для чого використовується команда WRITE?
15. Для чого використовується команда Traversal operations ?
16. Для чого використовується команда Traps?

17. З яких загальних частин складається SNMP повідомлення?

18. Що містить інформаційна частина повідомлення?

19. Як називають SNMP повідомлення?

20. Яка команда SNMP використовується для отримання значення якоїсь змінної?

21. Яка команда SNMP використовується для присвоєння значення якійсь змінній?

22. Яка команда SNMP використовується для повідомлення про зміну стану керованого об’єкта?

23. Що визначає структура керуючої інформації (SMI)?

24. Які типи інформації визначені в SMI?

25. Що містить інформація про мережні адреси (Network addresses)?

26. Що розуміють під лічильниками (Counters)?

27. Що розуміють під вимірювачем (Gauges)?

28. Що розуміють під вимірювачем тіками(Ticks)?

29. Які атрибути можуть мати об’єкти в базі керуючої інформації?

30. Який принцип взаємодії агента та менеджера покладено в основу роботи протоколу SNMP?

31. Яку функцію виконує команда Get-request ?

32. Яку функцію виконує команда GetNext-Request?

33. Яку функцію виконує команда Get – Response?

34. Яку функцію виконує команда Set?

35. Що ще можна встановити за допомогою команди Set?

36. Коли агент ініціює переривання (ловушку)?

37. Яку функцію виконує команда Trap?

38. Які команди передаються від менеджера до агента?

39. Які команди передаються від агента до менеджера?

40. Який транспортний протокол і які порти застосовуються для обміну по протоколу SNMP?

41. Про що свідчить повідомлення Trap coldStart?

42. Про що свідчить повідомлення Trap warmStart?

43. Про що свідчить повідомлення Trap linkDown?

44. Про що свідчить повідомлення Trap linkUp?

45. Про що свідчить повідомлення Trap authenticationFailure?

46. Яку структуру має MIB-I?

47. Наведіть основні категорії об’єктів MIB-I?

48. Які дані містить категорія «System»?

49. Які дані містить категорія «Interfaces»?

50. Які дані містить категорія «Address Translation Table»?

51. Які дані містить категорія «Transmission»?

52. Наведіть приклади об’єктів, які описують якійсь мережний пристрій.

53. Наведіть приклади об’єктів, які описують якійсь інтерфейс мережного пристрою.

54. Які недоліки MIB-II усуває RMON MIB?

55. Які функції виконує RMON зонд?

56. У чому полягає основна проблема інформаційної безпеки протоколу SNMP?

57. Яка структура моделі інформаційної безпеки описана для протоколу SNMP?

58. Які функції покладаються на підсистему безпеки?

59. Які функції покладаються на підсистему керування доступом?

60. Наведіть основні відмінності протоколів CMIP SNMP.

4. ЗАСОБИ МОНІТОРИНГУ ТА АНАЛІЗУ

Нижче буде розглянуто найбільш поширені агенти, які підтримують функції однієї зі стандартних MIB і використовують для обміну інформацією протокол SNMP або CMIP [О: 1, Д: 17, 18].

Як було заначено у попередніх розділах найбільш поширеним протоколом у системах моніторингу, навіть побудованих за концепцією TMN, є протокол SNMP. Головним недоліком SNMP – агентів є те, що при появі збоїв показання такого агента не можна вважати достовірними. Це особливо актуально, коли збої відбуваються в самому пристрої з встановленим SNMP-агентом. SNMP-агент пристрою спостерігає за сегментом мережі завжди тільки з однієї точки та, що особливо важливо для діагностики, не має можливості робити генерацію тестового трафіку. В результаті, якщо не все обладнання має вбудовані агенти, то помилок канального рівня в сегменті мережі може не фіксуватися. Крім того, ємність буфера для збирання пакетів у SNMP-агентів з підтримкою групи Packet Capture завжди обмежена і її часто недостатньо для локалізації складних дефектів у мережі.

4.1. Вбудовані системи діагностики та керування

Системи виконуються у вигляді програмно-апаратних модулів, які встановлюються в комунікаційне обладнання, а також у вигляді програмних модулів, вбудованих в операційні системи. Вони виконують функції діагностики та керування тільки одним пристроєм, і в цьому їхня основна відмінність від централізованих систем управління [О: 1, Д: 17, 18]. Прикладом засобів цього класу може служити модуль управління комутатором Ethernet, який реалізує функції автосегментації портів при виявленні несправностей, приписування портів сегментам VLAN і деякі інші. Як правило, вбудовані модулі управління «за сумісництвом» виконують роль SNMP-агентів, що постачають дані про стан пристрою для систем керування.

4.2. Сканери безпеки

Мережа складається з каналів зв'язку, вузлів, серверів, робочих станцій, прикладного і системного програмного забезпечення, баз даних та ін. Всі ці компоненти мають потребу в оцінці ефективності їх захисту. Засоби аналізу захищеності досліджують мережу і шукають «слабкі» місця в ній, аналізують отримані результати і на їх основі створюють різного роду звіти [О:1, Д: 17, 18]. Функціонувати такі засоби можуть на мережному рівні (Network-Based), рівні операційної системи (Host-Based) і рівні додатка (Application-Based). Сканери мережного рівня виявляють вразливості шляхом моделювання методів, які використовуються для атаки на віддалені системи. Практично будь-який сканер проводить аналіз захищеності в кілька етапів:

– збір інформації про мережу;
– виявлення потенційних вразливостей;
– підтвердження обраних вразливостей;
– генерація звітів.

Крім виявлення вразливостей, за допомогою засобів аналізу захищеності можна швидко визначити всі вузли корпоративної мережі, які доступні у момент проведення тестування, виявити всі використовувані в ній сервіси та протоколи, їх налаштування та можливості для несанкціонованого впливу (як зсередини корпоративної мережі, так і зовні). Також ці засоби виробляють рекомендації та покрокові заходи, що дозволяють усунути виявлені недоліки.

4.3. Аналізатори протоколів

Являють собою програмні або апаратно-програмні системи, які обмежуються на відміну від систем управління лише функціями моніторингу та аналізу трафіку в мережах [О: 1, Д: 17, 18]. Хороший аналізатор протоколів може захоплювати і декодувати пакети великої кількості протоколів, що застосовуються в мережах, – зазвичай кілька десятків. Аналізатори протоколів дозволяють сформулювати деякі логічні умови

для захоплення окремих пакетів і виконують повне декодування захоплених пакетів, тобто показують в зручній для фахівця формі вкладеність пакетів протоколів різних рівнів одне в одного з розшифруванням змісту окремих полів кожного пакета.

Аналізатори протоколів можуть бути локальними, тобто призначеними для діагностики лише одного сегменту мережі, або розподіленими. Останні дозволяють одночасно проводити аналіз великого числа (як мінімум двох) сегментів мережі.

Прикладом програмного локального аналізатора може служити LANanalyzer for Windows компанії Novell. Прикладом програмного розподіленого аналізатора протоколів – Distributed Observer компанії Network Instruments.

4.4. Порівняння найбільш поширених інструментів моніторингу

Всі системи, які базуються на відкритому коді (open sources) є безкоштовними для використання та розповсюдження. Програмні засоби з відкритим кодом, які призначені для моніторингу комп'ютерних систем та мереж, дозволяють стежити за вказаними вузлами та службами, оповіщають адміністратора у тому випадку, якщо деякі служби припиняють (або відновлюють) свою роботу і т. і. [О: 1, Д: 17, 18].

Порівняльні характеристики найбільш поширених інструментів моніторингу наведені в табл. 4.1. Кожен з розглянутих там інструментів має можливість графічного представлення інформації.

Для розгортання всіх розглянутих нижче інструментів потрібно, як мінімум, встановлений веб-сервер (зазвичай apache) та база даних (MySQL).

NETMON – модульна програма, де модулі працюють автономно і паралельно і використовують загальні модулі (планувальник, корелятор, модулі оповіщення та протоколювання подій). Всі дані можуть динамічно виводитися на web-інтерфейс.

Таблиця 4.1 – Характеристики основних інструментів моніторингу

Продукт	Характеристики
Nagios	Програма моніторингу працездатності вузлів і сервісів мережі (SMTP, POP3, HTTP, NNTP, PING). Моніторинг реалізується через набір зовнішніх програмних модулів (плагінів). Є зручний web-інтерфейс. Звіт про виявлені проблеми направляється на електронну пошту
Zabbix	Система моніторингу роботи додатків і вузлів у мережі. Стан мережі і сервісів відображається на графіку у вигляді динаміки зміни. Дані з вузлів передаються або шляхом запиту з сервера моніторингу, або через надсилання SNMP trap. Присутня гнучка система оповіщення
NETMON	Система моніторингу IP-мереж. NETMON дозволяє контролювати у реальному часі працездатність мережі, що складається з безлічі об'єктів – маршрутизатори, інтерфейси, вузли, сервіси. За допомогою NETMON можна відстежувати стан таких об`єктів, збирати значення всіляких лічильників, визначати події і реакцію на них, зберігати історію роботи мережі
Cacti	Аналізатор завантаження каналів з побудовою графіків з розширеним функціоналом, який підтримує моніторинг серверів через SMNP
MRTG	Аналізатор завантаження каналів зв'язку з графічним відображенням завантаження каналів
PRTG	Засіб моніторингу, який підтримує мережі стандартів LAN, WAN, WLAN і VPN. Також є можливість моніторингу фізичних і віртуальних веб-серверів, поштових серверів, файлових серверів, Linux систем, Windows клієнтів, маршрутизаторів

Різні модулі оповіщення реалізовані у вигляді набору зовнішніх програм, які викликаються при певних умовах. Способи збереження даних мають простий і гнучкий формат визначення. Кожен об'єкт моніторингу здатен мати власну підмножину модулів оповіщення і протоколювання даних.

Таким чином, NETMON дозволяє вирішувати такі основні завдання:

– моніторити стан маршрутизаторів, інтерфейсів і BGP-сесій;

– збирати і зберігати значення лічильників з інтерфейсів;

– моніторити стан вузлів та їх сервісів;

– протоколювати роботу мережі;

– динамічно визначати мережеву топологію;

– достовірно визначати джерело проблем і повідомляти про них;

– відображати це все на web-інтерфейс.

Плюси інструменту:

Досить простий в установці не вимагає додаткового встановлення php, достатньо підтримки cgi.

Мінуси інструменту:

Нестандартний синтаксис написання скриптів і конфігураційних файлів, що досить сильно ускладнює налаштування та конфігурування своїх задач.

MRTG – дозволяє будувати графіки завантаження та використання каналів зв'язку, збір даних відбувається по SNMP. Також, якщо немає можливості працювати з БД для підрахунку та обліку трафіку, то можна, так само, скористатися даним пакетом.

MRTG генерує HTML-сторінки із зображеннями у форматі PNG (Portable Network Graphics) для графічного представлення статистики трафіку. Можна сконфігурувати його для моніторингу будь-яких змінних SNMP, наприклад навантаження систем або мережних інтерфейсів.

Плюси інструменту:

Досить простий в установці, налаштуванні та конфігуруванні. Не завантажує систему додатковими БД, відповідно вище швидкість обробки.

Мінуси інструменту:

Частину конфігураційних файлів потрібно самому правити, так як програма не вміє їх створювати. При великій кількості каналів зв'язку, які потрібно моніторити по SNMP (більше 20) починає сильно завантажувати процесор на маршрутизаторі (при 25 каналах завантаження становило 50-60% в середньому).

Cacti – це дуже гнучка система моніторингу з великою кількістю додаткових плагінів та з web – інтерфейсом. Cacti будує інформативні графіки за допомогою утиліти RRDTool. Дозволяє моніторити доступність віддаленого обладнання засобами SNMP (v1/2c/3) і методом простого пінгу (ping). За допомогою SNMP можна дивитися завантаження каналів зв'язку, процесора на кінцевому обладнані, використану пам'ять RAM і ще багато чого.

Cacti надає користувачу зручний веб-інтерфейс до утиліти RRDTool, призначеної для роботи з циклічними базами даних (Round Robin Database), які використовуються для зберігання інформації про зміну однієї або декількох величин за певний проміжок часу. Інтерфейс відображення статистики, зібраної з мережних пристроїв, представлений у вигляді дерева, структура якого

задається самим користувачем. Як правило, графіки групуються по певним критеріям, причому один і той же графік може бути присутнім в різних гілках дерева (наприклад, трафік через мережний інтерфейс сервера – у тій, яка присвячена загальній картині інтернет-трафіку компанії, і гілці з параметрами даного пристрою). Кожен з графіків можна розглянути окремо, при цьому він буде представлений за останні день, тиждень, місяць і рік. Можливо самому вибрати часовий проміжок, за який буде згенерований графік, причому зробити це можна, як вказавши календарні параметри, так і просто виділивши мишкою певну ділянку на ньому.

Плюси інструменту:

Будує красиві і зрозумілі графіки. Все налаштування засноване на веб-інтерфейсі, що спрощує конфігурування. Є додаткова авторизація в цілях підвищення безпеки. Точно знімає показники завантаженості мережних інтерфейсів.

Мінуси інструменту:

Виводить не точні дані для OS FreeBSD по кількості запущених процесів, завантаженості процесора і пам’яті, що використовується дисковим простором. Мало стандартних шаблонів.

Zabbix – відкрита система моніторингу мережі в масштабі реального часу. Багато різноманітних корисних опцій, включаючи багаторівневі карти, гнучкі графіки, шаблони, підтримка зберігання даних в Oracle, MySQL, PostgreSQL, агрегування хостів та багато іншого.

Можливості:

- автоматизований процес установки (Installation Wizard);
- нові засоби для створення розподіленої мережі моніторингу;
- можливість розподілу навантаження, через запуск різних процесів (discove-rer, poller, HTTP poller, trapper тощо) на окремих серверах;
- реалізована підтримка SQLite;
- покращена зручність та швидкість роботи web-інтерфейсу;
- підтримка Jabber;

– підтримка UNIX/Windows клієнтів;
– засоби для імпорту та експорту даних (конфігурація, стан перевірок тощо) в XML форматі;
– нова схема установки привілеїв користувачів;
– формування слайдшоу з комбінацій екранів моніторингу.

Плюси інструменту:

Можливість повного налаштування через веб-інтерфейс, досить зрозумілий і зручний інтерфейс, зручна і зрозуміла інструкція зі встановлення і налаштування сервісів. Готові шаблони для моніторингу більш ніж 400-т параметрів. Простий та інтуїтивно зрозумілий синтаксис команд для написання власних шаблонів, повна підтримка зовнішніх модулів. Підтримка мультипроцесорних систем. Інформація може зніматися як через стандартні команди ping і traceroute про доступність хоста, так і через SNMP, є також власний агент (UNIX/Windows).

Агенти дозволяють зробити тонке налаштування для точної передачі даних з додатковим шифруванням і передавати дані будуть тільки тому хосту, що у них записаний. Конфігурування, яку інформацію знімати з хоста, проводиться лише на сервері, що значно полегшує налаштування, тому що на кожен хост, який потрібно моніторити, потрібно заходити тільки один раз, для встановлення агента. Агент і сервер використовують окремий порт TCP 10050 та 10051 для запиту і відповіді, що підвищує надійність системи. Для того, щоб переконатися в правильності даних, виставляється поріг спрацьовування, наприклад, у випадку негативної відповіді на запит можна задати перевірку протягом 15 секунд ще 3 рази, і якщо кожен з них поверне негативну відповідь, система приймає рішення про недоступність сервісу і посилає повідомлення адміністратору. Всі дані зберігаються в окремій базі даних, до якої ніхто сторонній не зможе отримати доступ. Зберігається вся історія зміни даних. Підтримка досить великої кількості обладнання різноманітного типу.

Мінуси інструменту:

Немає готових шаблонів для обладнання Cisco.

Nagios – програма моніторингу комп'ютерних систем і мереж з відкритим кодом. Призначена для спостереження, контролю стану обчислювальних вузлів і служб, оповіщає адміністратора в тому випадку, якщо якісь із служб припиняють (або відновлюють) свою роботу.

Спочатку Nagios була розроблена для роботи під Linux, але вона також добре працює і під іншими ОС, такими як Sun Solaris, FreeBSD, AIX і HP-UX.
Основні можливості:

- моніторинг мережевих служб (SMTP, POP3, HTTP, NNTP, ICMP, SNMP);
- моніторинг стану хостів (завантаження процесора, використання диска, системні логи) у більшості мережних операційних систем;
- підтримка віддаленого моніторингу через шифровані тунелі SSH або SSL;
- проста архітектура модулів розширень (плагінів) дозволяє, використовуючи будь-яку мову програмування за вибором (Shell, C ++, Perl, Python, PHP, C# та інші), легко розробляти свої власні способи перевірки служб;
- паралельна перевірка служб;
- можливість визначати ієрархії вузлів мережі за допомогою «батьківських» вузлів, що дозволяє виявляти і розрізняти вузли, які вийшли з ладу, а також ті, які недоступні;
- відправка повідомлень у разі виникнення проблем зі службою або вузлом (за допомогою пошти, смс, або будь-яким іншим способом, визначеним користувачем через модуль системи);
- можливість визначати обробники подій, що відбулися зі службами або вузлами;
- автоматична ротація лог-файлів;
- можливість організації спільної роботи декількох систем моніторингу з метою підвищення надійності і створення розподіленої системи моніторингу;
- включає утиліту nagiostats, яка виводить загальне зведення по всім вузлам, за якими ведеться моніторинг.

PRTG є потужним засобом моніторингу для персональних комп'ютерів, заснованих на операційних системах сімейства Windows. Цей додаток може застосовуватися в мережах будь-якого розміру і підтримує моніторинг мереж стандартів LAN, WAN, WLAN і VPN. Також є можливість моніторингу фізичного і віртуального веб-сервера, поштових і файлових серверів, Linux систем, Windows клієнтів, маршрутизаторів і багато іншого. PRTG контролює як доступ і пропускну здатність, так і різні інші мережні параметри, наприклад, якість обслуговування, завантаження пам'яті і CPU. Така система надає мережним адміністраторам поточну інформацію та статистичні дані про поведінку мережі, що дозволяє оптимізувати ефективність, розташування і конфігурацію каналів передачі даних, маршрутизаторів, брандмауерів, серверів та інших мережних компонентів.

PRTG поширюється з різними схемами ліцензування, включаючи безкоштовну, некомерційну версію. У безкоштовної версії підтримуються всі надані сенсори, проте обмеженням є їх одночасне використання в кількості 10 сенсорів.

Для чого може бути використаний PRTG:

- моніторинг і повідомлення готовності/простою або затримок серверів;
- моніторинг стану системи різних апаратних пристроїв;
- моніторинг і підрахунок даних про пропускну здатність та використання мережних ресурсів (пристроїв);
- моніторинг додатків;
- моніторинг SLA (Service Level Agreement);
- моніторинг використання системи (завантаження CPU, вільної пам'яті RAM, жорсткого диску і т.і.);
- моніторинг продуктивності баз даних,
- моніторинг прикладних серверних систем (веб, поштових, файлових та ін.),
- класифікація/розподіл мережного трафіку за джерелом/ призначенням і змістом,

– моніторинг мережного оточення,
– виявлення незвичайних, підозрілих, або шкідливих дій користувачів і пристроїв,
– вимірювання QoS і VoIP параметрів і управління сервісних рівнем угоди (SLA),
– виявлення та оцінювання мережних пристроїв,
– знаходження незапланованих зв'язків між мережними компонентами для виявлення потенційних проблем у захисті та оцінки використання мережі та апаратних ресурсів,
– моніторинг відмовостійкості при використанні сконфігурованого кластера обробки відмов.

Візуалізація результатів можлива через власний графічний інтерфейс програми або в інтернет-браузері. Крім того, в програму інтегрований веб-сервер, який дозволяє отримувати дані за допомогою віддаленого підключення, а вбудована системи автентифікації дозволяє стежити за результатами в багатокористувацькому режимі.

PRTG підтримує різноманітні способи сповіщення:

– 9 технологій сповіщення, таких як: пошта, SMS, syslog і SNMP trap (пастки), HTTP запити, журнал подій, Amazon SNS, виконання скриптів та інше,
– різноманітні тригери, наприклад, оповіщення стану, порогові попередження, умовні оповіщення, що розширюють оповіщення,
– різні підтвердження попереджень для їх подальшого виключення, а також планування цих сповіщень,

PRTG підтримує різні технології для моніторингу мереж (SNMP, WMI (Windows Management Instrumentation), SSH, Packet Sniffling та NetFlows для оцінки пропускної здатності каналів), дозволяє моніторити параметри QoS, підтримує резервування, віртуальні оточення, бази даних.

Запитання для самоперевірки та контролю засвоєння знань

1. Який основний недолік агентів систем керування на базі протоколів SNMP або CMIP?
2. Що розуміють під вбудованими системами діагностики та керування?
3. У чому полягає головна відмінність вбудованими системами діагностики та керування від централізованих систем?
4. Які функції виконують сканери безпеки?
5. На яких архітектурних рівнях можуть працювати сканери безпеки?
6. Що являють собою аналізатори протоколів?
7. Яку основну функцію виконують аналізатори протоколів?
8. Що характерно для локальних аналізаторів протоколів?
9. Що характерно для розподілених аналізаторів протоколів?
10. Наведіть приклади найбільш поширених систем моніторингу, які базуються на відкритому коді.

5. ПОНЯТТЯ ЯКОСТІ ОБСЛУГОВУВАННЯ

У конвергованій мережі завжди присутні різні типи трафіку, що пред'являють свої вимоги до параметрів середовища передачі, таких як [О: 5, 6, 7]:

– ***затримка,***

– ***смуга пропускання,***

– ***рівень втрати пакетів.***

При цьому вони можуть по-різному сприймати зміни цих параметрів. Наприклад, протокол TCP має вбудовані засоби управління смугою пропускання, що використовуються для кожного окремо взятого з'єднання, і здатний виявляти втрати пакетів і ініціювати повторну передачу. Проте, прикладна програма, що працює з використанням цього протоколу, може мати нижній критичний рівень смуги пропускання. В цьому випадку необхідно забезпечити трафіку такої прикладної програми необхідну смугу пропускання і мінімізувати втрату пакетів. В той же час, протокол UDP не має таких засобів виявлення втрати пакетів, а відповідальність за це покладається на прикладну програму. Таким чином, UDP-трафік виявляється чутливішим до середовища передачі, ніж TCP-трафік. Проте, існує низка прикладних програм, характерних для мультисервісних мереж, для яких доцільна робота саме по цьому протоколу. Такими є, наприклад, відеоконференції і передача голосового трафіку. Ці застосування пред'являють досить жорсткі вимоги до транспортної мережі, оскільки погіршення будь-якого з параметрів негативно позначається на якості сервісу (втрата кадрів при відтворенні відео і дуже відчутні спотворення звуку і паузи при передачі мови). А так як в мережі завжди присутні і інші види трафіку, які можуть використовувати велику смугу пропускання, але, бути менш чутливими до параметрів транспортної мережі, виникає необхідність диференційованої обробки різних видів трафіку. Іншим підходом для забезпечення необхідних параметрів передачі є попереднє резервування ресурсів мережі для конкретних видів трафіку перед початком передачі.

І диференційована обробка і попереднє резервування направлені на оптимальне використання ресурсів мережі при одночасному забезпеченні ресурсами критично важливих прикладних програм [О: 5, 6, 7]. При цьому у разі потреби може відбуватися витіснення трафіку менш критичних додатків.

Терміну «якість обслуговування» можна дати наступне визначення.

Якість обслуговування (QoS) – це забезпечення мережею необхідних характеристик передачі для різних типів мережного трафіку.

В загальному випадку якість обслуговування базується на сукупності механізмів, що дозволяють оптимально використовувати ресурси мережі і забезпечують необхідні характеристики передачі даних.

Якість обслуговування (QoS) можна охарактеризувати наступними параметрами [О: 5, 6, 7]:

- *швидкість передачі або пропускна спроможність;*
- *затримка* (час розповсюдження пакету);
- *число дейтаграм, що стоять у черзі для передачі*;
- *завантаженість каналу*;
- *вимоги безпеки*;
- *тип трафіку;*
- *число проміжних вузлів на шляху передачі даних;*
- *можливості проміжних зв'язків (наприклад, наявність декількох маршрутів передачі даних).*

Функції якості обслуговування в мережах IP (IP QoS) полягають в забезпеченні гарантованого і диференційованого обслуговування мережного трафіку шляхом передачі контролю над використанням ресурсів і завантаженістю мережі її оператору. QoS є набором вимог, що висуваються до ресурсів мережі при транспортуванні потоку даних. Крім того QoS забезпечує заснований на системі правил контроль за засобами підвищення продуктивності IP-мережі, такими як: механізми розподілу ресурсів, комутація, маршрутизація, механізми обслуговування черг і механізми відкидання пакетів [О: 5, 6, 7].

Нижче перераховані деякі з основних переваг якості обслуговування в IP мережах:

– забезпечення підтримки існуючих і тих що з'являються мультимедійних служб і прикладних програм. Деякі нові застосування, такі, як передача голосу по мережах IP (VoIP), пред'являють певні вимоги до якості обслуговування;
– передача контролю над ресурсами мережі і їх використанням оператору мережі;
– забезпечення гарантії обслуговування і диференціювання мережного трафіку. Ця умова є необхідною для об'єднання аудіо, відео-трафіку і трафіку прикладних програм в межах однієї IP-мережі;
– дозволяє постачальникам послуг пропонувати клієнтам додаткові послуги поряд із стандартною послугою негарантованої доставки даних (іншими словами, надавати послуги відповідно до так званого класу обслуговування – Class of Service (CoS)). Постачальник послуг може визначити декілька класів додаткових послуг і налагодити мережеві правила, що дозволяють обробляти трафік кожного класу відповідно до заданих параметрів;
– дає можливість організувати обслуговування мережного трафіку в залежності від прикладної програми, що згенерувала цей трафік, інформація про яку міститься в заголовку IP-пакету;
– грає значну роль в розвитку нових мережних технологій, таких, як віртуальні приватні мережі (Virtual Private Networks – VPN).

5.1. Рівні якості обслуговування

Мережний трафік складається з безлічі потоків, що були згенеровані прикладними програмами кінцевих вузлів. Ці прикладні програми відрізняються одна від одної різними вимогами до обслуговування і до робочих характеристик мережі. По суті, вимога до обслуговування кожного потоку цілком і повністю визначається вимогами прикладної програми, що згенерувала

цей потік. Отже, для того, щоб з'ясувати вплив структури запитів, що існують в мережі, на якість обслуговування, необхідно визначити типи мережних прикладних програм.

Здатність мережі забезпечувати різні рівні обслуговування, що запрошуються тими або іншими мережними прикладними програмами, поряд з проведенням контролю над характеристиками продуктивності (смугою пропускання, затримкою/тремтінням і втратою пакетів) може бути класифікована на три категорії [О: 5, 6, 7].

А. Негарантована доставка даних (best-effort service). Забезпечення зв'язності вузлів мережі без гарантії часу і самого факту доставки пакету в точку призначення. Слід зазначити, що відкидання пакету може статися тільки в разі переповнювання буфера вхідної або вихідної черги маршрутизатора.

Б. Диференційоване обслуговування (differentiated service). Передбачає розділення трафіку на класи на основі вимог до якості обслуговування. Кожен клас трафіку диференціюється і обробляється мережею відповідно до заданих для цього класу механізмів QoS. Подібна схема забезпечення якості обслуговування досить часто називається схемою CoS. Диференційоване обслуговування само по собі не передбачає забезпечення гарантій послуг, що надаються. Відповідно до даної схеми трафік розподіляється по класах, кожен з яких має свій власний пріоритет. З цієї причини диференційоване обслуговування доволі часто також називають «м'якими» QoS (soft QoS).

Диференційоване обслуговування зручно застосовувати в мережах з інтенсивним трафіком прикладних програм. В цьому випадку важливо забезпечити відділення адміністративного трафіку мережі від решти всього трафіку і призначити йому пріоритет, що дозволяє у будь-який момент часу бути упевненим у зв'язності вузлів мережі.

В. Гарантоване обслуговування (guaranteed service). Передбачає резервування мережних ресурсів з метою задоволення специфічних вимог до обслуговування з боку потоків трафіку. Відповідно до гарантованого

обслуговування виконується попереднє резервування мережних ресурсів по всій траєкторії руху трафіку. Гарантоване обслуговування також називають «жорсткими» QoS (hard QoS) у зв'язку з пред'явленням строгих вимог до ресурсів мережі. Прикладні програми, що вимагають гарантованого обслуговування, включають в себе мультимедійні програми, що проводять передачу голосової інформації і відеопотоків.

5.2. Характеристики продуктивності мережного з'єднання

Впровадження механізмів QoS передбачає забезпечення з боку мережі з'єднання між кінцевими вузлами з певними обмеженнями по продуктивності.

Основними характеристиками продуктивності мережного з'єднання є смуга пропускання, затримка, тремтіння і рівень втрати пакетів [О: 5, 6, 7].

Смуга пропускання – номінальна пропускна спроможність середовища передачі інформації, протоколу або з'єднання. Цей термін досить ефективно визначає «ширину каналу», потрібну прикладній програмі для взаємодії у мережі. Як правило, кожне з'єднання, що потребує гарантованої якості обслуговування, вимагає від мережі резервування мінімальної смуги пропускання.

Затримка при передачі пакету (packet delay), або латентність (latency), на кожному переході складається із затримки серіалізації, затримки поширення і затримки комутації. Нижче наведені визначення кожного з перерахованих типів затримки.

Затримка серіалізації (serialization delay) – час, який потрібний пристрою на передачу пакету при заданій ширині смуги пропускання. Затримка серіалізації залежить як від ширини смуги пропускання каналу передачі інформації, так і від розміру пакету, що передається. Наприклад, передача пакету розміром 64 байт при заданій смузі пропускання 3 Мб/с займає всього лише 171 нс. Затримка серіалізації дуже сильно залежить від смуги пропускання: передача того ж самого пакету розміром 64 байт при заданій

смузі пропускання 19.2 Кб/с займає вже 26 мс. Досить часто затримку серіалізації називають ще затримкою передачі (transmission delay).

Затримка поширення (propagation delay) – час, який потрібний біту інформації, що передається, для досягнення приймаючого пристрою на іншому кінці каналу. Затримка поширення залежить від відстані і середовища передачі інформації, що використовується, а не від смуги пропускання. Для ліній зв'язку глобальних мереж затримка поширення вимірюється в мілісекундах.

Затримка комутації (switching delay) – час, який потрібен пристрою, що отримав пакет, для початку його передачі наступному пристрою. Як правило, це значення менше 10 нс.

Зазвичай, кожен з пакетів, що належить одному і тому ж потоку трафіку, передається з різним значенням затримки. Затримка при передачі пакетів змінюється залежно від стану проміжних мереж. За умови перевантаження мережі, затримки при організації черг в маршрутизаторах починають впливати на спільну затримку при передачі пакетів і приводять до виникнення різниці в затримці при передачі різних пакетів одного і того ж потоку.

Коливання затримки при передачі пакетів отримали назву ***тремтіння при передачі пакетів*** (packet jitter). Тремтіння пакетів має велике значення, оскільки саме воно визначає максимальну затримку при прийомі пакетів в кінцевому пункті призначення. Приймаюча сторона, залежно від типа прикладної програми, що використовується, може спробувати компенсувати тремтіння пакетів за рахунок організації приймального буфера для зберігання прийнятих пакетів на якийсь час, менший або рівний верхній межі тремтіння. До цієї категорії відносяться прикладні програми, орієнтовані на передачу/прийом безперервних потоків даних, наприклад IP-телефонія або прикладні програми, що забезпечують проведення відеоконференцій.

Рівень втрати пакетів (packet loss) визначає кількість пакетів, що відкидаються мережею під час передачі. Основними причинами втрати пакетів є перевантаження мережі і пошкодження пакетів під час передачі по фізичним каналам зв'язку. Найчастіше відкидання пакетів відбувається в місцях

перевантаження, де число пакетів, що надходять, набагато перевищує верхню межу розміру вихідної черги. Крім того, відкидання пакетів може бути викликано недостатнім розміром вхідного буфера. Як правило, рівень втрати пакетів виражається як частка відкинутих пакетів за певний інтервал часу.

Деякі прикладні програми не здатні нормально функціонувати або ж функціонують украй неефективно в разі втрати пакетів. Подібні програми вимагають від мережі гарантії надійної доставки всіх пакетів. Як правило, добре спроектовані мережі характеризуються дуже низьким значенням втрати пакетів. Втрата пакетів також невластива прикладним програмам, для яких були заздалегідь зарезервовані потрібні цим програмам ресурси. Відкидання пакетів є неминучим явищем при негарантованій доставці трафіку. Відкинуті пакети вказують на неефективне використання ресурсів мережі, частина яких була витрачена на доставку пакетів в точку, де вони були відкинуті.

5.3. Механізми якості обслуговування

Як було зазначено вище, QoS забезпечує наскрізну гарантію передачі даних і заснована на системі правил контроль за засобами підвищення продуктивності IP-мережі, такими, як механізми розподілу ресурсів, комутація, маршрутизація, механізми обслуговування черг і механізми відкидання пакетів.

Для цього необхідно реалізувати ряд функцій [О: 5, 6, 7].

1. Функція класифікації. Маршрутизатори, розташовані на границі мережі, використовують функцію класифікації для розпізнавання пакетів, що належать різним класам трафіку, залежно від значення одного або декількох полів в заголовку IP-пакету.

2. Функція маркування пакетів використовується для розмітки класифікованого трафіку шляхом встановлення значення поля пріоритету або поля коду диференційованого обслуговування (Differentiated Services Code Point – DSCP).

3. Обмежуюча функція. Постачальники послуг використовують обмежуючу функцію для приведення параметрів клієнтського трафіку, що поступає в мережу, у відповідність з його профілем.

4. Вирівнююча функція. Користувачі послуг використовують вирівнюючу функцію для дозування трафіку, що поступає в мережу постачальника послуг, і вирівнювання його інтенсивності відповідно до заданого профілю.

Механізми обслуговування черг

Мінімальна вимога, що пред'являється до алгоритмів обслуговування черг з підтримкою функції QoS, – здатність диференціювати і визначати вимоги до обробки різних пакетів. Відповідно до цих параметрів алгоритм обслуговування повинен планувати порядок передачі поставлених в чергу пакетів [О: 5, 6, Д: 19]. Частота обслуговування пакетів одного і того ж потоку трафіку визначає виділену цьому потоку смугу пропускання.

Найбільш поширеним механізмом обслуговування черг в маршрутизаторах і комутаторах є механізм, що став вже традиційним – «першим прийшов, першим вийшов» (first-in, first-out – FIFO). Незважаючи на простоту реалізації, для механізму FIFO характерні декілька фундаментальних проблем, що ускладнюють виконання функцій якості обслуговування [О: 5, 6, Д: 19]. Так, механізм FIFO не передбачає пріоритетної обробки чутливого до затримки трафіку шляхом його переміщення на початок черги. Весь трафік обробляється однаково, без врахування приналежності потоків до різних класів з різними вимогами до обслуговування.

Традиційний механізм обслуговування черг FIFO передбачає відкидання всіх вхідних пакетів після досягнення максимального значення довжини черги. Подібний спосіб управління чергою отримав назву «відкидання хвоста» (tail drop) і характеризується тим, що сигнал про перевантаження поступає лише у момент фактичного переповнювання черги. Механізм FIFO не передбачає проведення яких-небудь активних дій із запобігання перевантаженню або зменшення розміру черги з метою зниження часу затримки.

Для усунення наведених недоліків механізму FIFO використовують активні алгоритми управління чергами, які дозволяють маршрутизатору передбачати перевантаження ще до переповнювання черги [О: 5, 6, Д: 19]. Такими алгоритмами є: зважений алгоритм довільного раннього виявлення (Weighted Random Early Detection – WRED), зважений механізм рівномірного обслуговування черг (Weighted Fair Queuing – WFQ), зважений механізм кругового обслуговування (Weighted Round Robin – WRR) і механізм кругового обслуговування з дефіцитом (Deficit Round Robin – DRR). Більш детально особливості деяких з цих алгоритмів буде розглянуто в наступному розділі.

Комутація

Головна функція маршрутизатора полягає в швидкій і ефективній комутації вхідного трафіку на відповідні вихідні інтерфейси згідно інформації, що зберігається в таблиці маршрутизації. При цьому можуть бути використані такі механізми [О: 5, 6].

1. Кешування, або буферизація пакетів перед передачею.

Традиційний механізм розповсюдження пакетів через кеш, незважаючи на його ефективність, має низьку масштабованість; крім того, його продуктивність залишає бажати кращого в періоди нестабільного функціонування мережі, що виражається в різкому збільшенні витрат на обслуговування кеша і зниженні ефективності комутації пакетів.

2. Метод просування пакетів, що враховує топологію мережі, володіє безперечними перевагами перед методом, що базується на кешуванні пакетів, що обумовлено збігом таблиці просування пакетів, яка використовується у даному методі, з таблицею маршрутизації. Механізм просування пакетів, що враховує топологію мережі, називається також методом швидкісної комутації пакетів Cisco (Cisco Express Forwarding – CEF).

Маршрутизація

Традиційна маршрутизація здійснюється на підставі адреси призначення пакету і передбачає вибір найкоротшого маршруту, що зберігається в таблиці маршрутизації. Проте подібний механізм є недостатньо гнучким для деяких

мережних сценаріїв. Для забезпечення QoS більш ефективною є маршрутизація на основі політики. На відміну від традиційних механізмів маршрутизації, маршрутизація на основі політики при виборі оптимального маршруту передачі даних дозволяє враховувати ряд додаткових параметрів, які визначають QoS [О: 5, 6, 7]. Це, перш за все, завантаженість каналів передачі та проміжних маршрутизаторів, з яких складається той чи інший маршрут.

Сучасні протоколи маршрутизації, що працюють по методу вибору найкоротшого шляху, дозволяють враховувати такі значення метрики, як адміністративну відстань, вагу або число переходів. Маршрутизація пакетів здійснюється на підставі інформації, що зберігається в таблиці маршрутизації, без врахування вимог потоку трафіку до якості обслуговування або доступності мережних ресурсів вздовж всього маршруту. QoS-маршрутизація є механізмом маршрутизації пакетів, що враховує вимоги потоків трафіку до якості обслуговування і здійснює вибір маршруту залежно від наявності мережних ресурсів.

5.4. Особливості використання технологій канального і мережного рівнів для забезпечення наскрізної якості обслуговування

Технології QoS канального рівня еталонної моделі OSI здатні стати повноцінним рішенням лише в мережах обмеженого розміру і не можуть бути застосовані для забезпечення наскрізної якості обслуговування в територіально-розподілених та глобальних IP-мережах внаслідок того, що ці мережі складаються з безлічі сегментів, побудованих на базі різних технологій канального рівня. Наскрізна взаємодія для забезпечення QoS в таких мережах починається з мережного рівня еталонної моделі OSI. Тільки протокол мережного рівня (яким є протокол IP) може забезпечити надання наскрізних послуг QoS.

З метою реалізації наскрізної якості обслуговування протокол IP, крім забезпечення наскрізної взаємодії між вузлами, повинен встановити

відповідність між функціями QoS мережного рівня і механізмами QoS канального рівня (ATM, Frame Relay, IEEE 802.1p і IEEE 802.1Q) (особливо в комутованих мережах) [О: 5-9].

Ще одним можливим підходом є використання технології мультипротокольної комутації міток (MPLS), яка допомагає забезпечити якість обслуговування в мережах IP і надає розширені можливості по управлінню трафіком, які сприяють організації заснованих на технології MPLS віртуальних приватних мереж (VPN) [О: 10, 11].

5.5. Методи управління потоком передачі

Існує дві стратегії управління якістю обслуговування [О: 5, 6, Д: 20, 21, 22, 23]:

– ***надання інтегрованих послуг (intserv);***

– ***надання диференційованих послуг (diffserv).***

При реалізації першої стратегії, для інформування мережі про потреби різних потоків трафіку використовують наступні сигнальні протоколи якості обслуговування.

1. Протокол резервування ресурсів (Resource Reservation Protocol – RSVP). RSVP – це сигнальний протокол QoS, який дозволяє кінцевим прикладним програмам, що вимагають певні гарантовані послуги, проводити наскрізну сигналізацію своїх QoS – вимог.

Робоча група IETF визначила RSVP як сигнальний протокол для архітектури інтегрованих послуг [Д: 24, 25]. Цей протокол дозволяє прикладним програмам посилати сигнали в мережу про свої QoS – вимоги для кожного потоку. Щоб визначити кількісні характеристики цих вимог з метою управління доступом, використовуються службові параметри.

2. Диспетчер смуги пропускання підмережі (Subnet Bandwidth Manager – SBM) – це версія протоколу RSVP і його розширень для канального рівня мереж стандарту IEEE 802 (перш за все, мереж Ethernet) [Д: 26].

Друга стратегія управління якістю обслуговування – диференціація послуг (diffserv). Диференціація послуг передбачає наявність декількох рівнів якості обслуговування, кожному з яких відповідає власна архітектурна модель забезпечення функцій QoS.

Головним завданням підходу диференційованих послуг є визначення стандартизованого байту диференційованої послуги (DS) – байту типу обслуговування (Type of Service – ToS) в заголовку пакету IPv4, або байту класу трафіку (Traffic Class) пакету IPv6 – і його відповідне маркування [О: 5, 6, Д: 22]. Від даної маркеровки залежить ухвалення специфічного рішення про просування пакету даних на кожному переході (per-hop behavior – PHB), тобто в кожному проміжному вузлі.

Архітектура диференційованих послуг забезпечує базову основу, для надання різних пропозицій залежно від вимог, що пред'являються до якості обслуговування.

Необхідний рівень послуг вибирається шляхом встановлення відповідного значення поля коду диференційованої послуги (Differentiate Services Code Point – DSCP) для кожного окремого IP-пакету. Код диференційованої послуги буде визначати послідовність рішень про просування пакету в кожному проміжному вузлі мережі постачальника послуг (PHB-політика). PHB-політика може бути визначена в термінах пріоритету в наданні ресурсів по відношенню до інших PHB-політик або ж за допомогою таких вимірюваних характеристик трафіку, як затримка пакетів, рівень втрати пакетів або тремтіння трафіку [Д: 22, 23, 25, 26, 27].

Запитання для самоперевірки та контролю засвоєння знань

1. Чим викликана необхідність забезпечення якості обслуговування у пакетних мережах?
2. У чому полягає диференційована обробка різних видів трафіку?
3. Що передбачає попереднє резервування ресурсів мережі?
4. Що розуміють під якістю обслуговування?
5. Якими параметрами характеризується якість обслуговування?
6. У чому полягають функції якості обслуговування в IP-мережах?
7. Які механізми можуть бути використані для вирішення задач забезпечення якості обслуговування?
8. Які виділяють переваги реалізації якості обслуговування в мережах?
9. На які категорії поділяють мережі за критерієм забезпечення різних рівнів якості обслуговування?
10. Що розуміють під негарантованою доставкою даних?
11. Що розуміють під диференційованих обслуговуванням?
12. Чи надає диференційоване обслуговування гарантії якості обслуговування?
13. Що розуміють під гарантованим обслуговуванням?
14. Наведіть основні характеристики продуктивності мережного з'єднання, які визначають якість обслуговування?
15. Що розуміють під смугою пропускання?
16. З яких складових складається затримка передачі пакету?
17. Що визначає затримка серіалізації?
18. Що визначає затримка поширення?
19. Що визначає затримка комутації?
20. Що розуміють під коливанням затримки, або «джитером»?
21. Яким чином можна «згладити» «джитер»?
22. Що визначає рівень втрати пакетів?
23. Наведіть основні причини втрати пакетів?
24. Що розуміють під функцією класифікації?

25. Що розуміють під функцією маркування пакетів?

26. Що розуміють під обмежуючою функцією?

27. Що розуміють під вирівнюючою функцією?

28. Які вимоги висувають до механізмів обробки черг для забезпечення якості обслуговування?

29. У чому полягає сутність механізму FIFO?

30. Чи можливо при використанні механізму FIFO реалізувати пріоритетну обробку пакетів у черзі?

31. Чи можливо при використанні механізму FIFO відстежувати перевантаження до досягнення максимального розміру черги?

32. Яким чином реалізують механізми якості обслуговування у протоколах маршрутизації?

33. Коли для забезпечення наскрізної якості обслуговування можна використовувати тільки механізми та протоколи 2-го рівня?

34. Коли для забезпечення наскрізної якості обслуговування необхідно задіяти механізми та протоколи 3-го рівня?

35. Які існують стратегії управління якістю обслуговування?

6. БАЗОВІ АЛГОРИТМИ ОБСЛУГОВУВАННЯ ЧЕРГ

6.1. Загальна характеристика розподілу канальних ресурсів

Задача розподілу канальних ресурсів між потоками розв’язується шляхом організації черг і встановленням певного порядку обслуговування поставлених у чергу пакетів [О: 5, 6, Д: 19]. Порядок обробки поставлених у чергу пакетів визначає частоту їхнього обслуговування, а отже, і кількість канальних ресурсів, які надаються даному потоку.

Черги й алгоритм їхньої обробки є інструментами управління перевантаженнями, коли мережний пристрій не встигає передавати пакети на вихідний інтерфейс в тому темпі, в якому вони надходять. Якщо причиною перевантаження є процесорний блок мережного пристрою, то для тимчасового збереження неопрацьованих пакетів використовується вхідна черга, тобто черга, зв’язана з вхідним інтерфейсом. У випадку, коли причина перевантаження полягає в обмеженій швидкості вихідного інтерфейсу (а вона завжди обмежена швидкістю канального протоколу), то пакети тимчасово зберігаються у вихідній черзі [О: 5, 6, Д: 19].

За своєю суттю, чергами є області пам’яті комутатора або маршрутизатора, де групуються пакети з однаковими пріоритетами передачі. Алгоритм обслуговування черги визначає порядок, у якому відбувається передача пакетів з цієї черги. Задача застосування всіх алгоритмів зводиться до того, щоб забезпечити найкраще обслуговування трафіка з більш високим пріоритетом за умови, що і пакетові з низьким пріоритетом гарантується передача з черги [О: 5, 6, Д: 19].

Алгоритм обслуговування черг як механізм забезпечення QoS має задовольняти таким вимогам [О: 5, 6, Д: 19].

1. Алгоритм обслуговування черг із підтримкою QoS повинен мати засіб диференціювання пакетів і засіб визначення рівня обслуговування кожного пакета.

2. Алгоритм має гарантувати якість обслуговування пакетів шляхом розподілення ресурсів для кожного окремого потоку трафіка або за допомогою визначення відносного пріоритету потоків.

3. Алгоритм обслуговування черг із підтримкою QoS має забезпечувати захист і рівномірну обробку всіх потоків трафіка з однаковим пріоритетом (наприклад, трафіка, що доставляється без гарантій).

4. Додатковими вимогами до алгоритму обслуговування черг є легкість реалізації і підтримка управління доступом для потоків, що потребують гарантованого надання ресурсів.

Механізми обслуговування черг можна класифікувати за такими ознаками [О:5, 6, Д:19] (рис. 6.1): принцип розподілу ресурсів (без розподілу ресурсів, пріоритетний розподіл шляхом застосування однойменного обслуговування, пропорційний розподіл шляхом кругового обслуговування черг і рівномірний розподіл шляхом реалізації максимінної схеми); надання гарантій за будь-якими параметрами мережного з'єднання (у термінах смуги пропускання або гарантованої затримки); принцип формування черг (формування черг за потоками або за класами); режим виконання (розподілений на процесорах інтерфейсних плат або нерозподілений на центральному процесорі маршрутизатора).

Найчастіше в маршрутизаторах і комутаторах застосовуються такі алгоритми обробки черг [О:5, 6, Д:19]:

– алгоритм FIFO;
– пріоритетне обслуговування (Priority Queuing, PQ);
– справедливе обслуговування (Fair Queuing, FQ);
– обслуговування на основі класу (Class Based Queuing, CBQ);
– зважене справедливе обслуговування (Weighted Fair Queuing, WFQ);
– зважене справедливе обслуговування на основі класу (Class Based WFQ, CBWFQ);
– обслуговування з малою затримкою (Low Latency Queuing, LLQ);

– зважене кругове обслуговування (Weighted Round-Robin, WRR) і його модифікації;
– кругове обслуговування з дефіцитом (Deficit Round-Robin, DRR) і його модифікації.

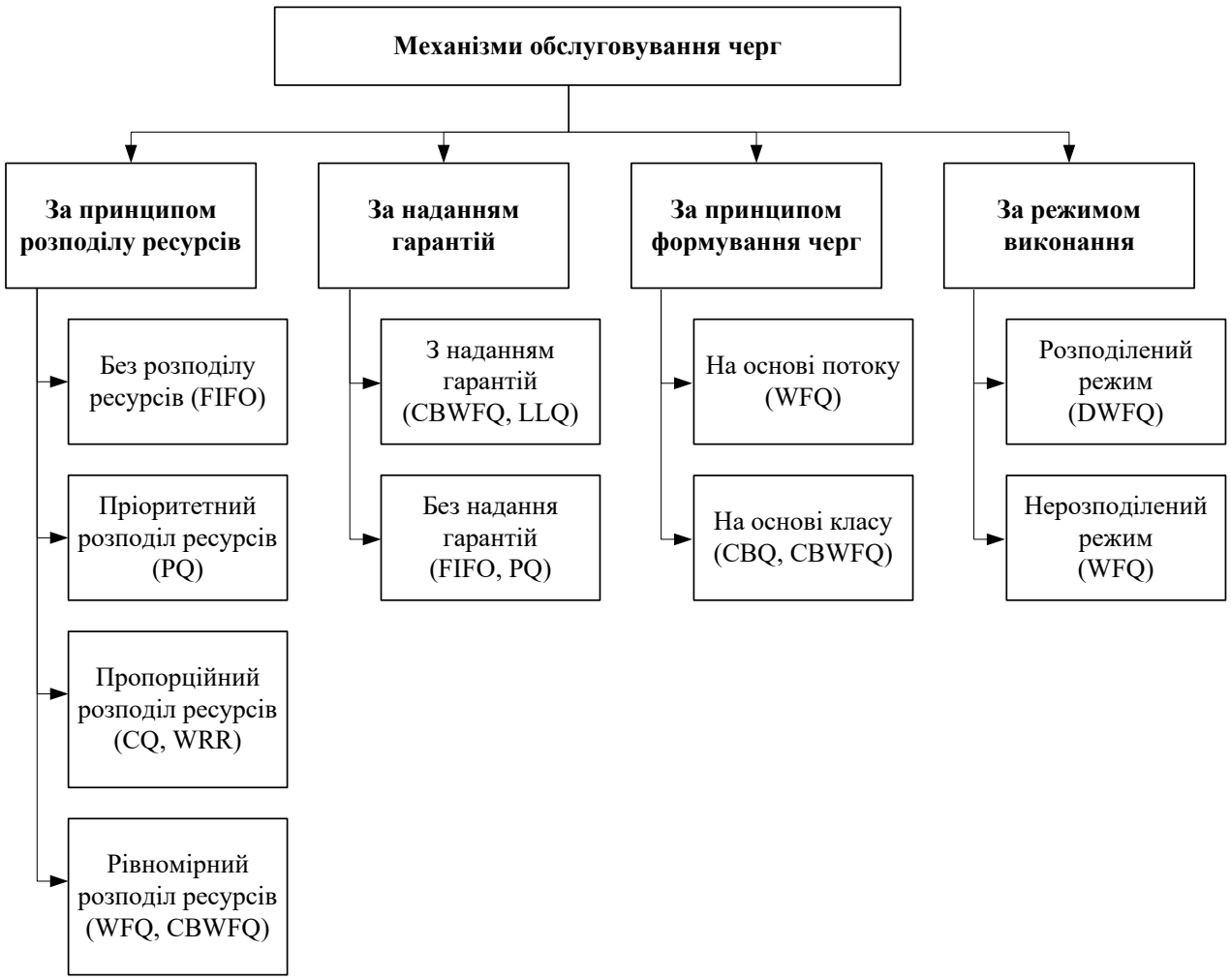

Рисунок 6.1 – Класифікація механізмів обслуговування черг

6.2. Механізм обслуговування черг FIFO

FIFO (First In – First Out, «першим прийшов – першим пішов») є найпростішим механізмом обслуговування черг, відповідно до якого пакети передаються на вихід у тому порядку, в якому вони надійшли на вхід [О: 5, 6, Д: 19].

Цей механізм є алгоритмом обробки черг за замовчуванням у всіх пристроях з комутацією пакетів. Незаперечною його перевагою є простота

реалізації і відсутність потреби в конфігуруванні. Головний недолік – неможливість диференційованої обробки пакетів різних потоків. Усі пакети знаходяться у загальній черзі на рівних правах – і пакети чутливого до затримок голосового трафіку, і пакети нечутливого до затримок, але дуже інтенсивного трафіку резервного копіювання, тривалі пульсації якого можуть надовго затримати голосовий пакет. Іншими словами механізм FIFO не здатний забезпечити рівномірне обслуговування потоків трафіка з однаковим пріоритетом і захистити їх від потоків з нерівномірною інтенсивністю, він допускає перевагу потоків високої інтенсивності над всіма іншими потоками трафіка. Отже, черги FIFO необхідні для нормальної роботи мережних пристроїв, але їх недостатньо для підтримки диференційованої якості обслуговування.

6.3. Механізм пріоритетного обслуговування черг PQ

Механізм пріоритетного обслуговування черг припускає наявність чотирьох вихідних черг – черги з високим, середнім, звичайним і низьким пріоритетом обслуговування [О: 5, 6, Д: 19]. Належність потоку трафіка до кожної з чотирьох черг визначається мережним адміністратором. Пакети, що знаходяться у черзі з високим пріоритетом обслуговування, передаються першими. Коли ця черга виявляється порожньою, починається передача пакетів наступної за пріоритетом обслуговування черги і т.і. Відзначимо, що передача пакетів черги із середнім пріоритетом обслуговування не почнеться доти, доки не будуть обслуговані всі пакети черги з вищим пріоритетом. Механізм пріоритетного обслуговування черг PQ підтримує класифікацію пакетів залежно від вхідного інтерфейсу, простого або розширеного списку доступу на підставі IP-адреси, розміру пакету і додатку, що згенерував цей пакет (за номер порту транспортного протоколу) [О: 5, 6, Д: 19]. Слід зазначити, що некласифікований трафік за замовчуванням належить до черги зі звичайним пріоритетом обслуговування. У межах черги пакети обробляються відповідно до механізму FIFO. Обмежений розмір буферної пам’яті мережного пристрою вимагає, щоб черги пакетів, що очікують обслуговування, мали деяку граничну

довжину. Звичайно за замовчуванням усім пріоритетним чергам надаються однакові буфери, але багато пристроїв дозволяють адміністраторові призначати кожній черзі буфер індивідуального розміру. Пріоритетне обслуговування черг забезпечує високу якість обслуговування для пакетів із черги з найвищим пріоритетом. Якщо середня інтенсивність їхнього надходження в пристрій не перевершує пропускної здатності вихідного інтерфейсу (і продуктивності внутрішніх блоків самого пристрою), то пакети вищого пріоритету завжди отримують ту пропускну здатність, що їм потрібна. Рівень затримок пакетів з найвищим пріоритетом також мінімальний. Що ж стосується інших класів пріоритетів, то якість їхнього обслуговування буде нижчою, ніж у пакетів найвищого пріоритету, причому рівень зниження заздалегідь передбачити досить важко. Це залежить від інтенсивності трафіка з найвищим пріоритетом. Пріоритетне обслуговування пакетів потрібно в мережах, де передача трафіка, необхідного для розв'язання критично важливих задач, має бути здійснена навіть за умови повного домінування трафіка з найвищим пріоритетом в моменти перевантаження мережі. Тому пріоритетне обслуговування зазвичай застосовується в тому випадку, коли в мережі є один клас трафіка, наприклад, голосовий трафік, чуттєвий до затримок, але його інтенсивність невелика, так що його обслуговування не занадто знижує якість передачі пакетів іншого трафіка [О: 5, 6, Д: 19]. Отже, головною перевагою пріоритетного обслуговування є відносна простота в реалізації і висока якість обслуговування трафіка з найвищим пріоритетом. Серед основних недоліків можна відзначити такі [О: 5, 6, Д: 19]:

- необхідність ретельного контролю трафіка на етапі доступу в мережу з метою належного надання пріоритету;
- відсутність обмеження щодо обслуговування черги для кожного з рівнів пріоритету;
- високий ризик зменшення обслуговування низькопріоритних потоків потоками з найвищим пріоритетом.

6.4. Зважені черги, що настроюються

Механізм CQ (Custom Queuing, CQ, «черга, що настроюється» або «звичайна черга») надає адміністратору можливість вручну задати розподіл ресурсів, тобто розподілити смугу пропускання [О: 5, 6, Д: 19]. Іноді CQ називають також механізмом замовленого резервування ресурсів. Механізм CQ обробляє кожну непорожню чергу в режимі кругового обслуговування (Round Robin), усякий раз передаючи з цієї черги певну кількість пакетів, яка додатково настроюється. Замовлене обслуговування черг дозволяє гарантувати надання певної частини смуги пропускання трафіка, необхідному для виконання критично важливих задач, у той же час не забуваючи і про інший трафік мережі [О: 5, 6, Д: 19]. Механізм замовленого обслуговування черг CQ підтримує класифікацію пакетів у залежності від вхідного інтерфейсу, простого або розширеного списку доступу на підставі IP-адреси, розміру пакета та додатку, що згенерував цей пакет. Відповідно до механізму CQ трафік можна розподілити за 16 чергами [О: 5, 6, Д: 19] (рис. 6.2). Крім цього, існує додаткова нульова системна черга, яка призначена виключно для обробки високопріоритетних пакетів, таких, як пакети підтримки з'єднання і управляючих пакетів. Ця черга є пріоритетною і її пакети передаються першими. Трафік користувачів не може оброблятися в системній черзі.

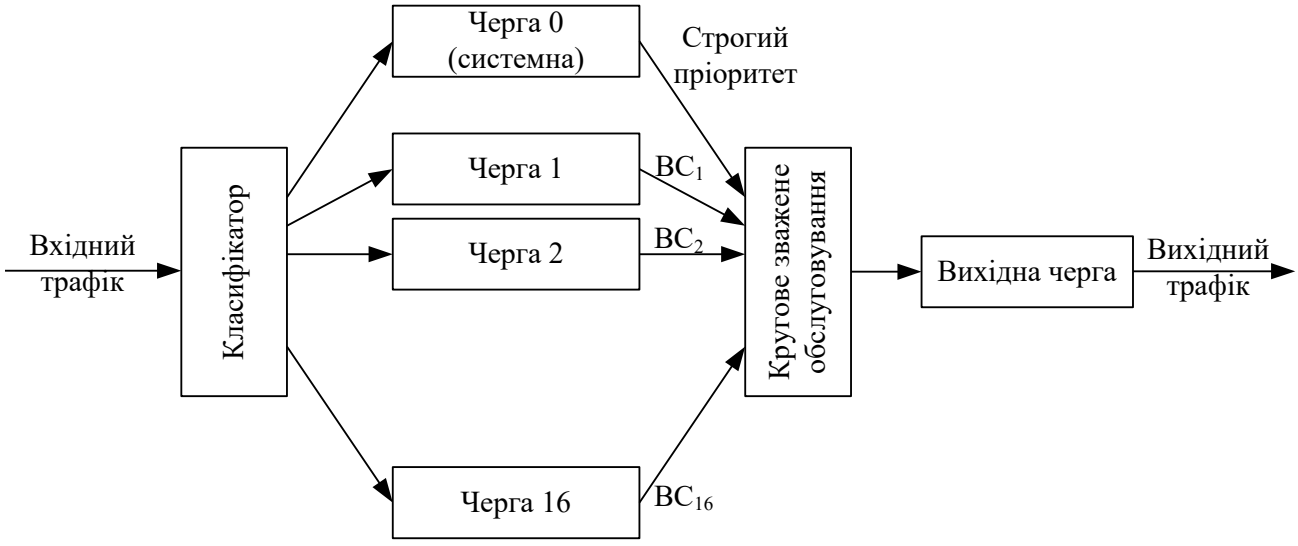

Рисунок 6.2 – Механізм замовленого обслуговування черг CQ

У рамках CQ черги 1-16 обслуговуються за круговим принципом, починаючи з першої. З кожною чергою пов’язаний свій лічильник байт (Byte Count, BC), який визначає кількість байтів, що має бути передана з цієї черги, перш ніж перейти до наступної. Пакети з черги передаються доти, поки не буде перевищено значення лічильника байт або черга не виявиться порожньою. Після цього система переходить до наступної непорожньої черги. Відзначимо, що якщо значення лічильника байт було перевищено, пакет, що передається в поточний момент часу, передається повністю. Наприклад, якщо значення лічильника байтів дорівнюватиме 100, а розмір пакета –1024 байта, то, щораз при обслуговуванні даної черги, маршрутизатор передаватиме 1024, а не 100 байт [О: 5, 6, Д: 19].

Припустимо, що розмір пакетів трьох різних протоколів дорівнює 500, 300 і 100 байт відповідно. Для того, щоб рівномірно розподілити смугу пропускання між трьома протоколами, можна спробувати встановити значення лічильника байтів для кожної черги рівним 200. Слід зазначити, що подібне рішення, проте, не приведе до розподілу смуги пропускання в пропорції 33/33/33. Коли маршрутизатор обслуговує першу чергу, він передає щораз 500-байтовий пакет інформації; другу чергу – 300-байтовий пакет; третю чергу – два 100-байтових пакета. Отже, дійсний розподіл смуги пропускання здійснюється в пропорції 50/30/20. До такого непередбачуваного результату призвело занадто мале значення лічильника байтів [О: 5, 6, Д: 19].

Насправді досить велике значення лічильників байтів також може призвести до досить небажаного розподілу смуги пропускання. Якщо всім трьом чергам надати значення лічильника байтів, яке дорівнює 10 Кбайт, то, незважаючи на те, що пакети кожного з протоколів обслуговуватимуться швидко, з моменту обробки черги, відповідної конкретному протоколові, може пройти досить багато часу, перш ніж ця черга обслуговуватиметься знову. У розглянутому нами випадку одним з найбільш оптимальних рішень є встановлення значень лічильників байтів черг таким, що дорівнюють

500, 600 і 500 байт відповідно, що приведе до розподілу смуги пропускання у пропорції 31:38:31 [О: 5, 6, Д: 19].

Отже, задача розподілу смуги пропускання в заданих пропорціях між декількома потоками зводиться до визначення коректного значення лічильника байтів. Для розрахунку такого значення лічильника байтів BC може бути використаний такий алгоритм [О: 5, 6, Д: 19].

Нехай потрібно розподілити смугу пропускання між трьома чергами в пропорції $x : y : z$ з огляду на те, що розміри пакетів, що надходять у кожну з цих черг, складають q_x, q_y, q_z відповідно.

Крок 1. Розраховуються допоміжні змінні

$$a_x = \frac{x}{q_x},\ a_y = \frac{y}{q_y},\ a_z = \frac{z}{q_z}.$$

Крок 2. Отримані значення a_x, a_y, a_z нормалізуються й округляються у більший бік до найближчого цілого числа

$$a'_x = round\left(\frac{a_x}{\min(a_x, a_y, a_z)}\right),$$

$$a'_y = round\left(\frac{a_y}{\min(a_x, a_y, a_z)}\right),\ a'_z = round\left(\frac{a_z}{\min(a_x, a_y, a_z)}\right).$$

Крок 3. Розраховуються значення лічильників байтів

$$BC_x = a'_x \cdot q_x,\ BC_y = a'_y \cdot q_y,\ BC_z = a'_z \cdot q_z.$$

Крок 4. Розраховується розподіл смуги між чергами, який забезпечується отриманими значеннями лічильників байтів BC_x, BC_y, BC_z:

$$\frac{BC_x}{\sum_{i=1}^{n} BC_i} : \frac{BC_y}{\sum_{i=1}^{n} BC_i} : \frac{BC_z}{\sum_{i=1}^{n} BC_i},$$

де n – загальна кількість черг. У даному прикладі $n = 3$.

Крок 5. Якщо отриманий розподіл значно відхиляється від заданого ($x : y : z$), то необхідно повернутися до кроку 2 і помножити коефіцієнти пропорції кількості пакетів з кожної черги, що мають передаватися за один раз,

$\frac{a_x}{\min(a_x,a_y,a_z)}, \frac{a_y}{\min(a_x,a_y,a_z)}, \frac{a_z}{\min(a_x,a_y,a_z)}$ (до операції округлення) на деяке число. Це число підбирається, виходячи з міркувань максимального наближення коефіцієнтів пропорцій до цілих чисел, хоча воно не обов'язково має бути цілим.

6.5. Алгоритм організації черг на основі класу CBQ

Алгоритм CBQ (Class-Based Queuing) забезпечує ієрархічний розподіл ресурсів і цікавий, у першу чергу, ієрархічним підходом до організації черг [О: 5, 6, Д: 19].

У рамках CBQ смуга пропускання каналу поділяється між декількома класами трафіка в заздалегідь заданих співвідношеннях, причому кожному з класів у період перевантаження гарантується певна частка загальної смуги. За відсутності перевантаження смугу пропускання, яку виділено для одного класу трафіка і яка не використовується ним цілком, можна розділити між іншими класами, інтенсивність надходження яких перевищує виділену їм смугу. У рамках CBQ використовується два планувальника: загальний планувальник (general scheduler) і планувальник розподілу ресурсів (link-sharing scheduler), між якими проводиться строге розмежування [О: 5, 6, Д: 19] (рис. 6.3).

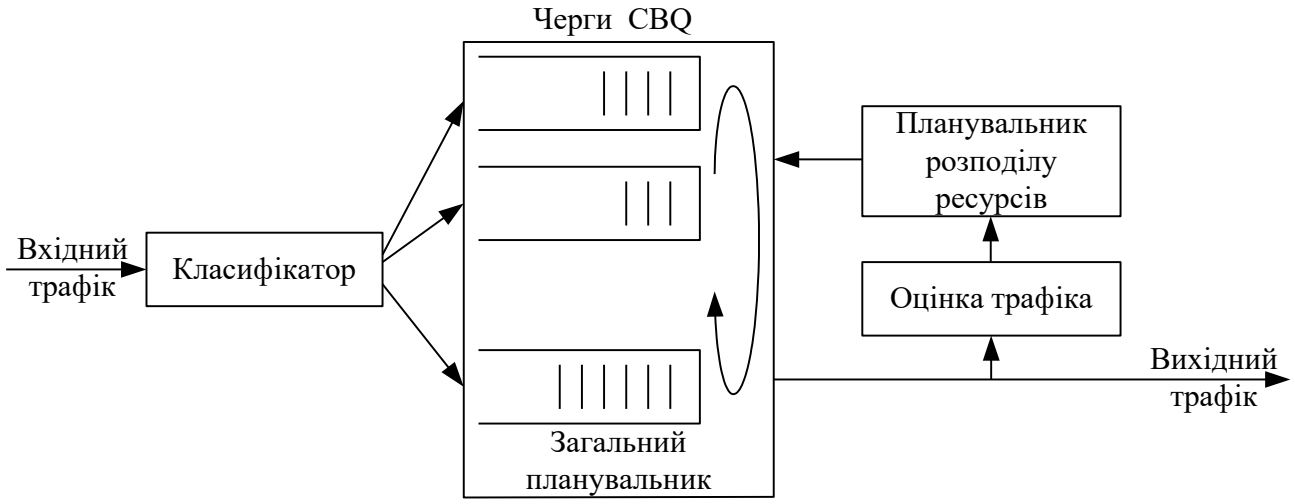

Рисунок 6.3 – Механізм CBQ

Вхідний трафік класифікується і розділяється на кілька черг відповідно до заздалегідь заданих правил фільтрації. Загальний планувальник (як правило, це WRR) визначає, з якої черги обслуговуватиметься наступний пакет. Черга вибирається виходячи з того, щоб гарантувати кожному класу трафіка щонайменше задану частку від сумарної пропускної здатності каналу (гарантована смуга). У рамках окремих черг використовується механізм FIFO [О: 5, 6, Д: 19].

Вихідний трафік контролюється шляхом оцінки часу між сусідніми пакетами одного класу трафіка. На підставі цієї інформації планувальник розподілу ресурсів класифікує трафік на такий, що перевищує ліміт (over-limit), на такий, що укладається в ліміт (under-limit) або на такий, що залишився (at-limit). Трафік вважається як таким, що перевищує ліміт, якщо смуга, яку він використовує більше, ніж йому гарантується, відповідно трафік є таким, що укладається в ліміт, якщо він використовує смугу менше йому гарантованої [О: 5, 6, Д: 19].

Якщо черга якого-небудь класу трафіка виявиться порожньою, планувальник розподілу ресурсів розділить гарантовану даному класові смугу між іншими класами. У випадку виникнення перевантаження, планувальник розподілу ресурсів переводить клас, що перевищив ліміт, у розряд пасивного і загальний планувальник тимчасово не обслуговує пакети з цієї черги [О: 5, 6, Д: 19].

Як уже відзначалося вище, у CBQ застосовується ієрархічний підхід до розподілу ресурсів (hierarchical link-sharing). Для окремого класу трафіка в CBQ створюється своя черга, причому існують різні варіанти побудови критеріїв класифікації, наприклад, класифікація може базуватися на додатку, який згенерував пакет, або на адресах відправника й одержувача. Приклади відповідних схем поділу ресурсів наведені на рис. 6.4 [О: 5, 6, Д: 19].

На рис. 6.4 для кожного класу трафіка зазначена частка сумарної смуги пропускання, гарантована даному класові в умовах перевантаження. Виділення

якому-небудь класові 0% означає, що в період перевантаження даному класові не гарантується передача.

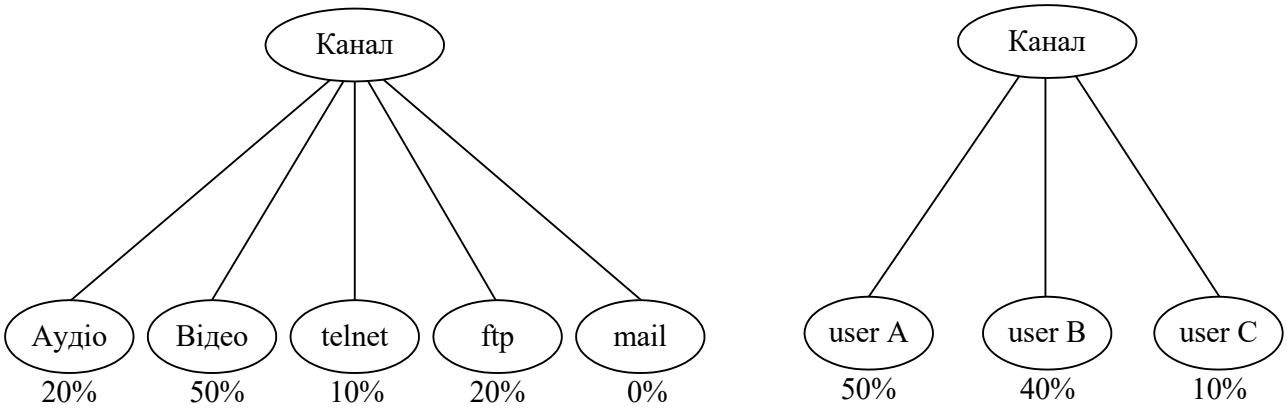

Рисунок 6.4 – Приклади розподілу ресурсів з використанням різних критеріїв класифікації трафіка

Приклад ієрархічного розподілу ресурсів наведений на рис. 6.5.

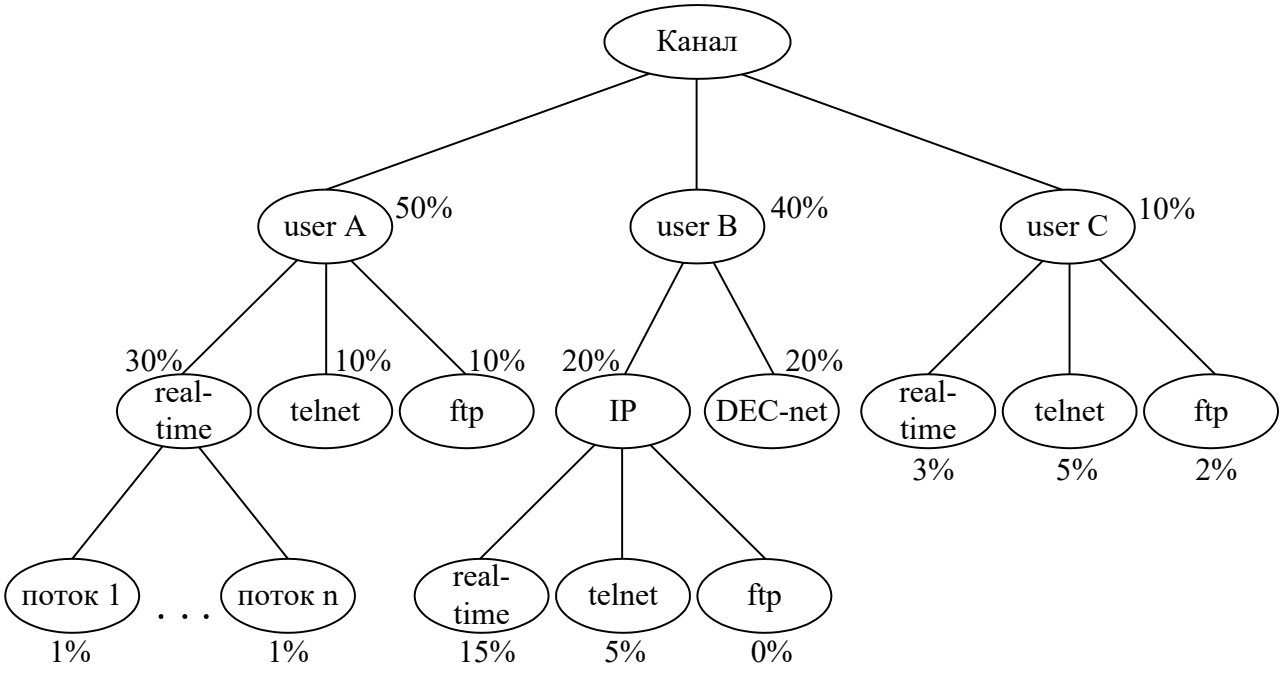

Рисунок 6.5 – Приклад ієрархічного розподілу ресурсів

Механізм обслуговування CBQ є дуже розповсюдженим, він реалізований у Packet Filter (pf) у сімействі операційних систем BSD [О: 5, 6, Д: 19].

Розглянемо приклад. Нехай маємо таку структуру черг (рис. 6.6).

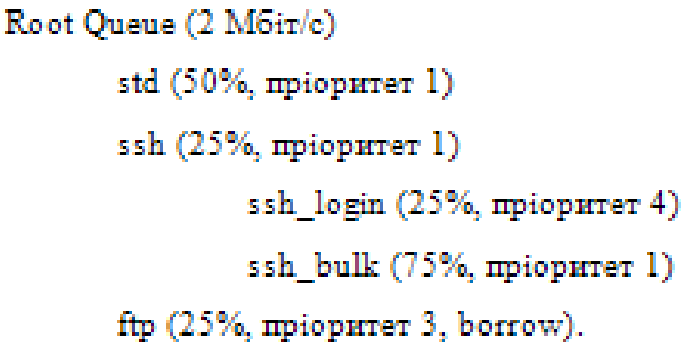

Рисунок 6.6 – Приклад ієрархічної структури черг

Тут Root Queue – коренева черга – займає смугу пропускання каналу 2 Мбіт/с. Коренева черга має три дочірні черги std, ssh і ftp, яким надається 50 %, 25 % і 25 % від сумарної смуги відповідно. Черга ssh у свою чергу має дві дочірні підчерги: ssh_login і ssh_bulk, що поділяють смугу, яку виділено для ssh, в співвідношенні 1:3 (25 % і 75 %).

У рамках CBQ можливе введення пріоритетів. Черги одного рівня ієрархії і рівного пріоритету обслуговуються за круговим принципом (WRR). Черга, що має вищий пріоритет, обслуговується першою. У даному прикладі спочатку обслуговується черга ftp (пріоритет 3), потім черги з пріоритетом 1 – std і ssh за круговим принципом. Коли обслуговуватиметься черга ssh, спочатку вилучаються пакети з ssh_login (пріоритет 4), потім з ssh_bulk (пріоритет 1). Пріоритети черг різного рівня ієрархії (ftp і ssh_login) не порівнюються. Запис borrow для черги ftp означає, що цій черзі дозволено запозичати ресурси (смугу пропускання), які виділені для інших черг цього ж рівня ієрархії, але ними не використовуються.

6.6. Схема максимум-мінімум рівномірного розподілу ресурсів

З метою забезпечення рівномірного обслуговування використовується схема максимум-мінімум рівномірного розподілу ресурсів (max-min fair-share allocation scheme). Основними її правилами є такі [О: 5, 6, Д: 19]:

- ресурси розподіляються в порядку підвищення вимог;
- користувач не може отримати обсяг ресурсів, який перевищує його потреби;

- ресурси розподіляються рівномірно між користувачами з незадоволеними вимогами.

Розглянемо приклад [О: 5, 6, Д: 19]. Припустимо, що загальний обсяг доступного ресурсу дорівнює 14 одиницям. Вимоги до ресурсу користувачів A, B, C, D та E складають 2, 2, 3, 5 і 6 одиниць, відповідно. Розподіл ресурсу починається з джерела з найменшими вимогами, який одержує обсяг ресурсу, що дорівнює відношенню всього запасу ресурсу до загальної кількості користувачів. Отже, у розглянутому нами випадку користувачам A і B буде надано 14 / 5 = 2,8 одиниць ресурсу (рис. 6.7). Однак вимоги користувачів A і B складають усього лише 2 одиниці ресурсу. У цьому випадку надлишковий ресурс обсягом 1,6 одиниць (по 0,8 одиниць з кожного користувача) рівномірно розподіляється між трьома іншими користувачами. Отже, користувачі C, D та E одержують по 2,8 + (1,6 / 3) = 3,33 одиниць ресурсу (рис. 6.8). Наступним за обсягом висунутих вимог до ресурсів є користувач C. Вимоги користувача C на 0,33 одиниці менше обсягу ресурсів, що йому пропонується. Надлишковий ресурс обсягом 0,33 одиниці рівномірно розподіляється між користувачами D і E, кожний з яких одержує по 3,33 + (0,33 / 2) = 3,5 одиниці ресурсу [О: 5, 6, Д: 19] (рис. 6.9).

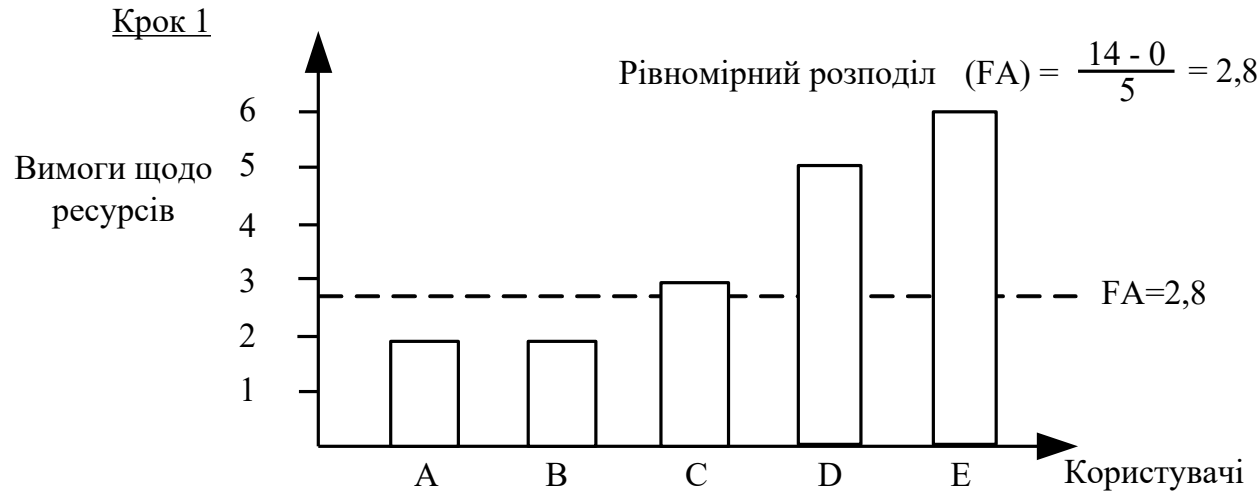

Рисунок 6.7 – Розподіл ресурсів для користувачів A і B

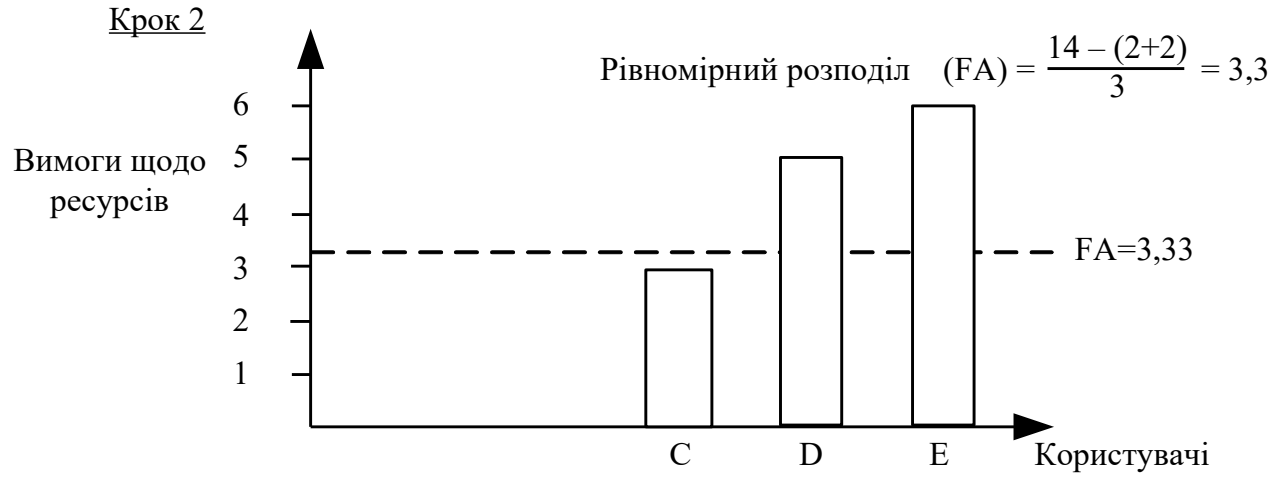

Рисунок 6.8 – Розподіл ресурсів для користувача C

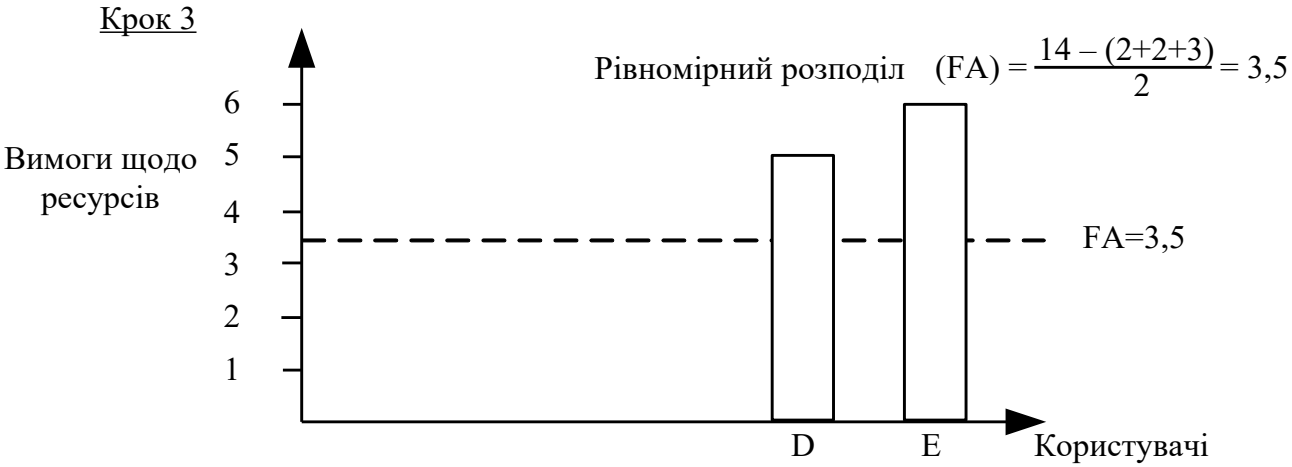

Рисунок 6.9 – Розподіл ресурсів для користувачів D та E

У загальному випадку обсяг ресурсів, наданий i-му користувачеві, розраховується за формулою [О: 5, 6, Д: 19]:

$$R_i = \frac{R_\Sigma - \overline{R}}{n},$$

де R_Σ – сумарний запас ресурсів; $\overline{R}$ – обсяг уже розподілених ресурсів;

n – кількість користувачів, яким ще потрібні ресурси.

Як видно з рис. 6.7-6.9, вимоги користувачів A та B задовольняються у повному обсязі, тому що вони не перевищують значення, отриманого в результаті рівномірного розподілу ресурсів між усіма користувачами.

На кроці 2 цілком задовольняються вимоги користувача C. А запити користувачів D і E залишаються незадоволеними.

Слід зазначити, що всі користувачі з незадоволеними вимогами (тобто з вимогами, обсяг яких перевищує їх максимінну рівномірну частку), одержують рівні обсяги ресурсів. Максимінна схема рівномірного розподілу ресурсів одержала свою назву в зв'язку з тим, що користувач з незадоволеними вимогами одержує максимум з можливих мінімальних рівномірних часток. Максимінна схема рівномірного розподілу ресурсів, у якій кожному користувачеві призначається певна вага, одержала назву зваженої максимінної схеми рівномірного розподілу ресурсів (weighted max-min fair-share allocation scheme). У відповідності до зваженої максимінної схеми рівномірного розподілу ресурсів кожному користувачеві надається рівномірна частка ресурсів, пропорційна до його ваги [О: 5, 6, Д: 19].

6.7. Алгоритм рівномірного обслуговування черг (FQ)

В основу роботи алгоритму рівномірного обслуговування черг (FQ) покладено обчислення порядкового номера кожного отриманого пакета. Цей порядковий номер пакета є службовою позначкою, яка визначає відносний порядок обробки пакетів [О: 5, 6, Д: 19].

Одним з основних понять алгоритму FQ є поняття лічильника циклів (Round Number, RN), значення якого відповідає кількості виконаних циклів побайтового планувальника кругового обслуговування в заданий момент часу. Лічильник циклів безпосередньо визначає значення порядкового номера пакета.

З метою ілюстрації принципу роботи алгоритму FQ розглянемо три потоки трафіка – A, B і C, розміри пакетів яких складають 128, 64 і 32 байт, відповідно [О: 5, 6, Д: 19]. Пакети надходять один за іншим у порядку Al, A2, A3, Bl, C1. Першим надходить пакет A1, потім –пакет A2 тощо.

Потік вважається активним (active), якщо хоча б один із пакетів з цього потоку знаходиться в очікуванні обслуговування; у іншому випадку потік вважається пасивним (inactive).

Припустимо, що системою FQ був отриманий пакет A1, який належить пасивному потокові. Повна обробка 128-байтового пакета побайтовим планувальником кругового обслуговування потребує 128 циклів. Якщо на момент одержання пакета A1 значення лічильника циклів дорівнювало 100, то після повної передачі пакета A1 це значення складе 100 + 128 = 228. По суті, порядковий номер пакета є номером циклу, в якому здійснюється передача останнього байту пакета. Оскільки в дійсності планувальник за один раз реалізує передачу всього пакета, а не його одного байту, він обслуговує весь пакет незалежно від того, чи зрівнявся лічильник циклів з порядковим номером пакета [О: 5, 6, Д: 19] (рис. 6.10).

Коли система FQ отримає пакет A2, він уже належатиме активному потокові трафіка завдяки наявності в черзі пакета A1 з порядковим номером 228. Порядковий номер пакета A2 дорівнюватиме 228 + 128 = 356, оскільки його слід передати після пакета A1. Аналогічно, пакетові A3 призначається порядковий номер 356 + 128 = 484. Оскільки пакети B1 і C1 належать пасивним потокам трафіка, їхні порядкові номери дорівнюють 164 = (100 + 64) і 132 = (100 + 32), відповідно (рис. 6.10). З урахуванням обчислених порядкових номерів пакети будуть обслуговані в такій послідовності: C1, B1, A1, A2, A3.

Отже, порядковий номер (Sequence Number) SN_L пакета довжиною L байт, що визначає черговість його обробки, можна розрахувати за формулою [О: 5, 6, Д: 19]:

$$SN_L = \begin{cases} L + RN, & \text{для пакетів пасивного потоку трафіка,} \\ L + \max\left(SN_q\right), & \text{для пакетів активного потоку трафіка,} \end{cases}$$

де RN – значення лічильника циклів на момент надходження пакета (дорівнює порядковому номерові останнього обслуговуваного пакета); $\max\left[SN_q\right]$ – значення найбільшого порядкового номера пакета, який поставлений в чергу цього потоку (для активного потоку).

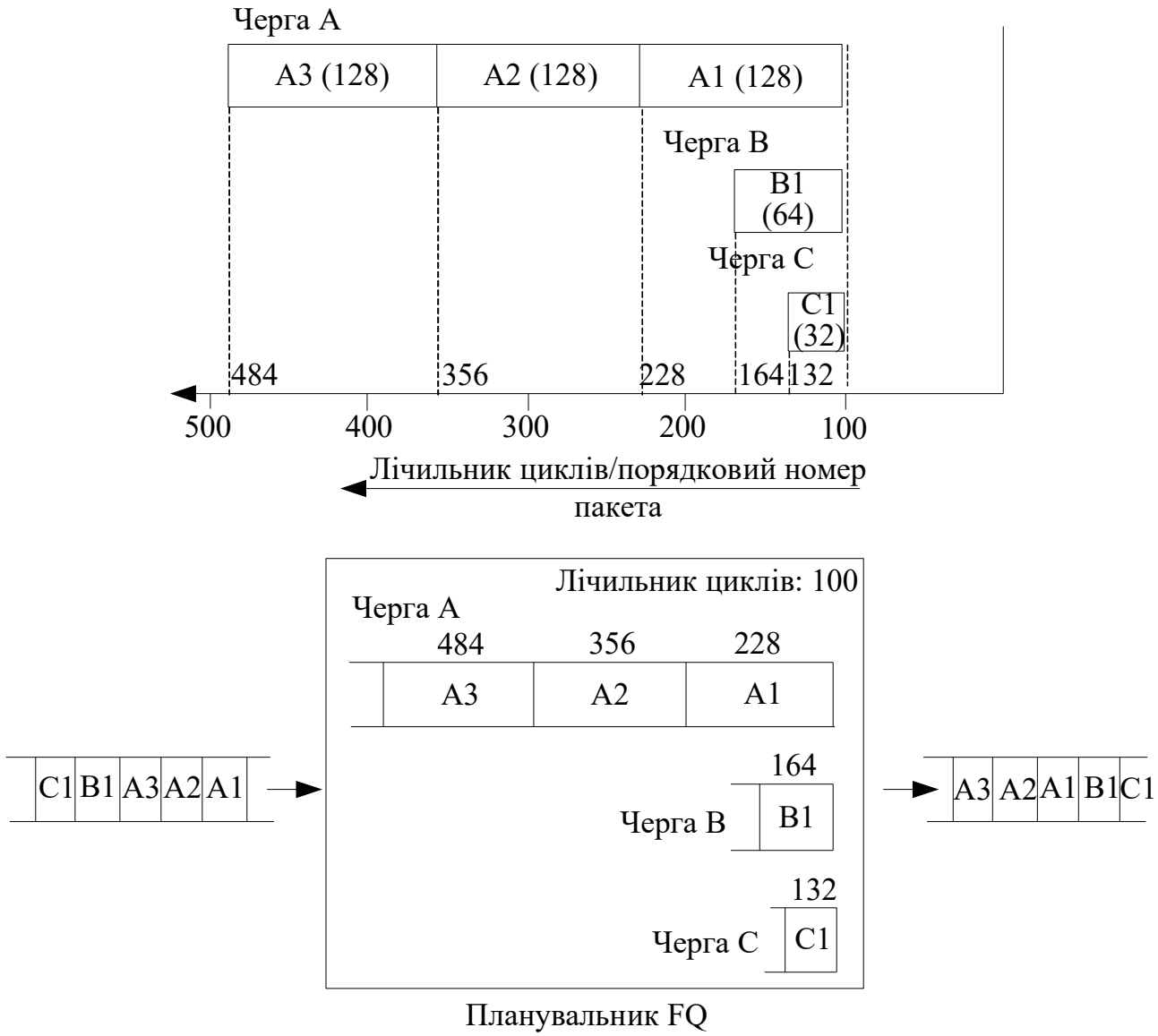

Рисунок 6.10 – Приклад моделювання побайтового планувальника кругового обслуговування за допомогою алгоритму FQ

Лічильник циклів *RN* використовується винятково для розрахунку порядкового номера пакетів, які належать новим (пасивним) потокам трафіка. Порядковий номер пакетів, що належать активним потокам трафіка, розраховується з урахуванням найбільшого порядкового номера пакета з цього потоку, який поставлений в чергу.

Слід зазначити, що лічильник циклів оновлюється при передачі кожного чергового пакета, при цьому його нове значення дорівнює порядковому номерові пакета, що передається. Отже, якщо 32-байтовый пакет D1, що належить новому потоку, буде прийнятий у момент передачі пакета A1,

значення лічильника циклів дорівнюватиме 228, а порядковий номер пакета D1 – 260 = (228 + 32). Оскільки порядковий номер пакета D1 менше порядкових номерів пакетів A2 і A3, він буде переданий раніше за цих пакетів. Зміна в порядку обслуговування пакетів схематично зображена на рис. 6.11 [О: 5, 6, Д: 19].

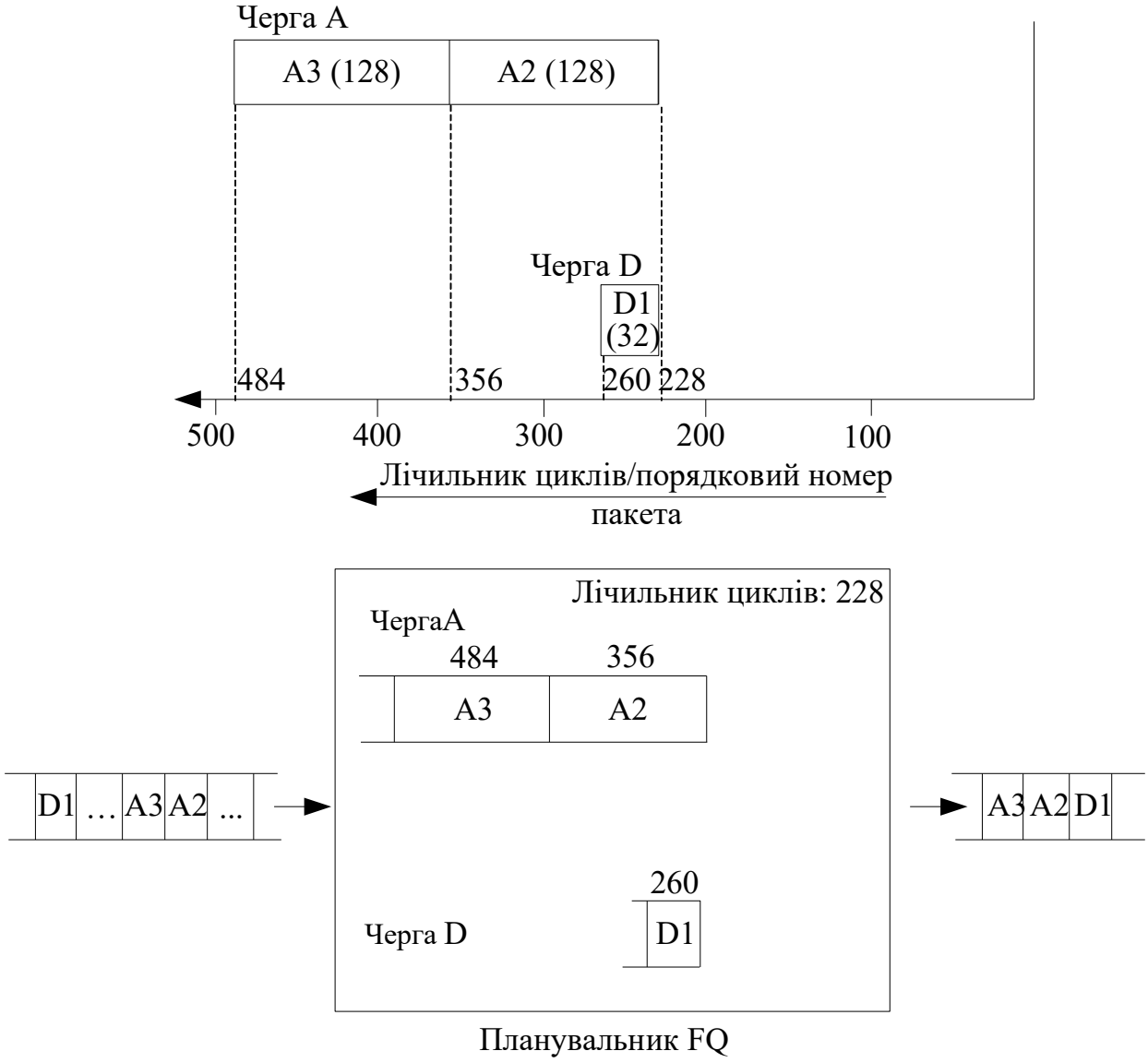

Рисунок 6.11 – Зміна в порядку обслуговування пакетів, яка викликана надходженням пакета D1 у момент передачі пакета A1

Запитання для самоперевірки та контролю засвоєння знань

1. Яким чином вирішується задача розподілу канальних ресурсів між потоками?
2. Що визначають алгоритми обслуговування черг?
3. Яким вимогам повинні задовольняти алгоритми обслуговування черг для забезпечення QoS?
4. Що характерно для механізму FIFO?
5. Який основний недолік механізму FIFO?
6. Що характерно для механізму пріоритетного обслуговування черг (PQ)?
7. За якими параметрами можна проводити класифікацію пакетів при використання механізму пріоритетного обслуговування черг (PQ)?
8. До черги з яким пріоритетом відноситься некласифікований трафік при використанні механізму пріоритетного обслуговування черг (PQ)?
9. Чи можна гарантувати при використанні механізму пріоритетного обслуговування черг (PQ) необхідну якість обслуговування для трафіку з пріоритетом менше найвищого?
10. Які основні недоліки механізму пріоритетного обслуговування черг (PQ)?
11. Що характерно для механізму зважених черг, що настроюються (CQ)?
12. За якими параметрами можна проводити класифікацію пакетів при використання механізму зважених черг, що настроюються (CQ)?
13. На скільки черг можна поділити трафік при використанні механізму зважених черг, що настроюються (CQ)?
14. Для чого використовується нульова системна черга в механізмі зважених черг, що настроюються (CQ)?
15. Яким чином обслуговуються черги в механізмі зважених черг, що настроюються (CQ)?
16. До чого зводиться задача розподілу смуги пропускання в заданих пропорціях між декількома потоками при використанні механізму зважених черг, що настроюються (CQ)?
17. Що характерно для алгоритму організації черг на основі класу (CBQ)?

18. Яку функцію виконує загальний планувальник у алгоритмі організації черг на основі класу (CBQ)?

19. Який параметр оцінюється при контролі вихідного трафіку в алгоритмі організації черг на основі класу (CBQ)?

20. На які категорії класифікується вихідний трафік у алгоритмі організації черг на основі класу (CBQ)?

21. Яку функцію виконує планувальник розподілу ресурсів у алгоритмі організації черг на основі класу (CBQ)?

22. У чому полягає ієрархічний підхід до розподілу ресурсів у алгоритмі організації черг на основі класу (CBQ)?

23. Яку задачу дозволяє вирішити схема максимум-мінімум рівномірного розподілу?

24. Які правила покладено в основу схеми максимум-мінімум рівномірного розподілу?

25. Який обсяг ресурсів може отримати користувач з незадоволеними вимогами в схеми максимум-мінімум рівномірного розподілу?

26. Що характерно для зваженої максимінної схеми рівномірного розподілу ресурсів?

27. Що покладено в основу роботи алгоритму рівномірного обслуговування черг (FQ)?

28. Чому відповідає значення лічильника циклів (Round Number, RN) у алгоритмі рівномірного обслуговування черг (FQ)?

29. Коли потік вважається активним у алгоритмі рівномірного обслуговування черг (FQ)?

30. Що визначає порядковий номер пакета в алгоритмі рівномірного обслуговування черг (FQ)?

31. Коли оновлюється лічильник циклів у алгоритмі рівномірного обслуговування черг (FQ)?

32. Чому дорівнюється нове значення лічильник циклів у алгоритмі рівномірного обслуговування черг (FQ)?

7. АРХІТЕКТУРА ДИФЕРЕНЦІЙОВАНИХ ТА ІНТЕГРОВАНИХ ПОСЛУГ

7.1. Архітектура диференційованих послуг (DiffServ)

Термін «диференційоване обслуговування» (Differentiated Service, DiffServ) застосовується як у широкому значенні для позначення рівня обслуговування, так і у вузькому – для позначення архітектури QoS, що забезпечує однойменний рівень обслуговування. DiffServ у вузькому значенні можна визначити як масштабовану архітектуру IP QoS, що забезпечує якість обслуговування на основі чітко визначених компонентів, які комбінуються з метою надання необхідних послуг, і яка орієнтована для застосування в мережах постачальників мережних послуг і в магістральних мережах. Диференціація (diffserv) послуг обумовлює наявність декількох рівнів якості обслуговування, кожному з яких відповідає власна архітектурна модель забезпечення функцій QoS [О:5,6, Д: 22, 23, 26-30].

Базові елементи архітектури DiffServ описані в декількох специфікаціях:

- RFC 2474 (Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers) задає спосіб маркування пакетів;
- RFC 2598 (An Expedited Forwarding PHB) і 2597 (Assured Forwarding PHB Group) визначають два основних різновиди сервісу;
- RFC 2475 (An Architecture for Differentiated Service) описує взаємодію компонентів в архітектурній моделі DiffServ.

Основним моментом у моделі DiffServ є диференціювання трафіка шляхом його розбивки на класи з різними пріоритетами (рівнем QoS), і, як наслідок, головною задачею DiffServ є визначення і стандартизація байту диференційованої послуги (DS, RFC 2474). Байт DS у рамках архітектури DiffServ замінює байт типу обслуговування (Type of Service, ToS) із заголовка пакета IPv4 і байт класу трафіка (Traffic Class) пакета IPv6. Специфікація RFC 2474 визначає байт DS, вводить поле коду диференційованої послуги (Differentiated Services Code Point, DSCP) і його маркування, від чого надалі

залежить політика покрокового обслуговування пакета (Per-Hop Behavior, PHB), тобто рішення про просування пакета в кожному проміжному вузлі мережі і, як результат, наданий рівень QoS.

Як правило, постачальник послуг обговорює разом із замовником профіль, згідно якого кожному рівню обслуговування відповідатиме певна інтенсивність передачі трафіку. Якщо ж об'єм трафіку, що передається перевищить встановлену квоту для даного рівня обслуговування, то надання обумовлених послуг для «надпланових» пакетів не гарантується. Після визначення послуги розробляються відповідні їй рішення про просування пакету (PHB-політика) для кожного вузла мережі, що підтримує дану послугу. PHB-політиці привласнюється певний код DSCP. PHB-політика – це політика поведінки мережевого вузла відносно пакетів з певним значенням поля коду диференційованої послуги (DSCP). Всі пакети потоку трафіку із специфічною вимогою до обслуговування несуть в собі одне і те ж значення поля DSCP.

Архітектура DiffServ (рис. 7.1) передбачає наявність класифікатора і формувача трафіка на границі DiffServ-домена, а також підтримку функції розподілу ресурсів у ядрі з метою забезпечення необхідної політики покрокового обслуговування PHB [O: 5,6].

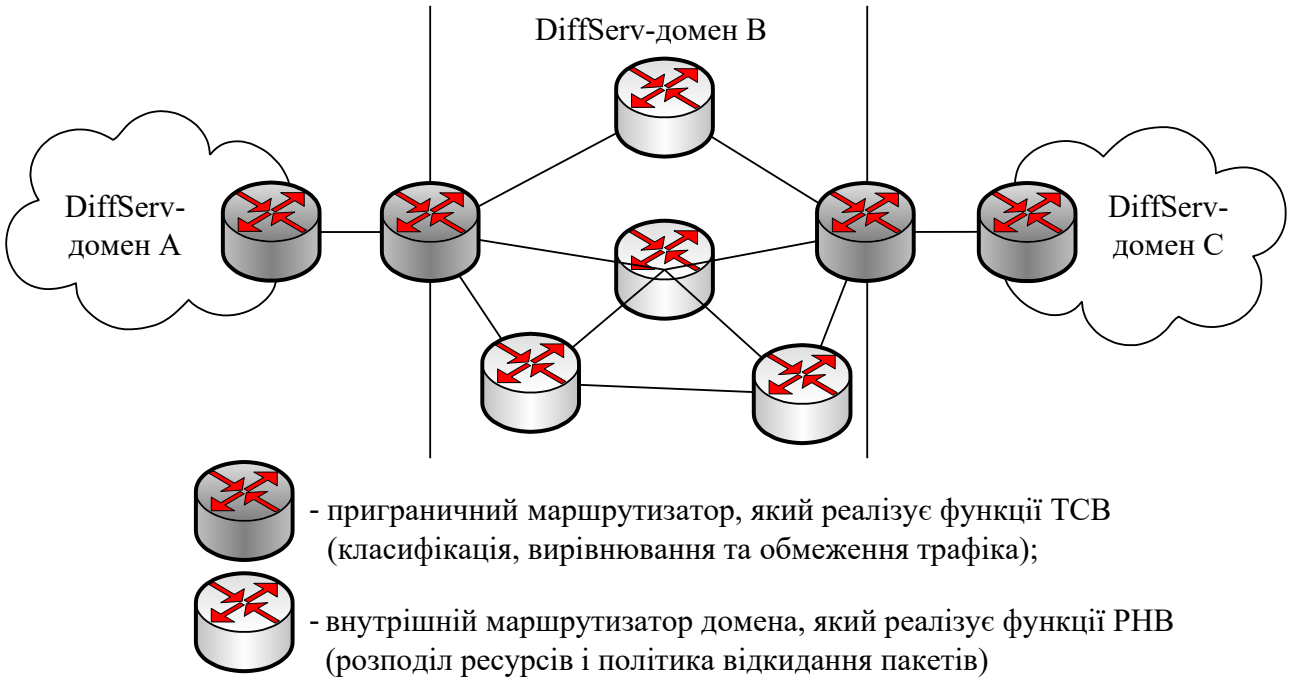

Рисунок 7.1 – Типи маршрутизаторів в архітектурі DiffServ

Всі вузли усередині diffserv-домену (сукупність вузлів, які реалізують певні диференційовані послуги) визначають PHB-політику, яка має бути застосована до пакету на основі значення поля коду диференційованої послуги, що зберігається в ньому.

Крім того, приграничні вузли diffserv-домену виконують дуже важливу функцію формування трафіку (TCB – Traffic Conditioning Block), що поступає в diffserv-домен. Формування трафіку включає виконання таких функцій, як класифікація пакетів і обмеження трафіку. Зазвичай, формування трафіку виноситься на вхідний інтерфейс вузла, пакети з якого поступають в diffserv-домен.

Формування грає вирішальну роль в управлінні трафіком, що поступає в diffserv-домен, оскільки в цьому випадку для кожного пакету мережа може визначити відповідну йому PHB-політику.

Приграничні маршрутизатори DiffServ-домену виконують функцію класифікації і встановлюють значення поля DSCP відповідно до необхідного рівня QoS. Як під час класифікації, так і потім під час формування трафіка важливу роль відіграє SLA – угода про якість обслуговування, яка попередньо укладається між постачальником послуг і замовником. Наявність SLA є необхідною умовою реалізації архітектури DiffServ. Дана угода встановлює критерії політики і визначає профіль трафіка. Очікується, що трафік буде згладжуватися (функція shaping) у вихідних точках домена відповідно до SLA, а у вхідній точці домена профілюватися (обмежуватися) відповідно до правил політики (функція policing). Будь-який трафік «поза профілем» (тобто такий, що виходить за верхні границі смуги пропускання, що зазначені в SLA) не одержує гарантій щодо обслуговування (або ж оплачується за підвищеною ціною у відповідності зі SLA). Критерії політики можуть включати час дня, адреси джерела і призначення, транспортний протокол, номери портів. У загальному випадку будь-який вміст пакета може використовуватися для застосування політики.

Внутрішні вузли DiffServ-домена (рис. 7.1) забезпечують передавання пакета з відповідним рівнем обслуговування, який визначається на підставі поля DSCP у його заголовку, тобто реалізують PHB-політику. PHB-політика – це політика дій мережного вузла по відношенню до пакетів з певним значенням поля DSCP. PHB-політику можна визначити в термінах пріоритету в наданні ресурсів відносно інших PHB-політик або ж за допомогою таких вимірюваних характеристик трафіка, як затримка пакетів, рівень втрати пакетів або тремтіння трафіка.

У табл. 7.1 наведений опис двох основних функціональних блоків цієї архітектури [О:5,6].

Таблиця 7.1 – Основні функціональні блоки архітектури диференційованих послуг

Функціональний блок	**Розташування**	**Функція, що виконується**	**Дія**
Формувачі трафіку	Вхідний інтерфейс прикордонного маршрутизатора diffserv-домена	Класифікація пакетів, вирівнювання і обмеження трафіку	Обмеження вхідного трафіку і установка значення поля DSCP на основі профілю трафіку
Пристрої, що реалізовують PHB-політику	Всі маршрутизатори diffserv-домена	Розподіл ресурсів і політика відкидання пакетів	PHB-політика обробки пакетів визначається на основі характеристик якості обслуговування, що відповідають заданому значенню поля DSCP

Отже архітектурою DiffServ з метою надання необхідної якості обслуговування передбачається така послідовність обробки пакетів.

1. На приграничних маршрутизаторах DiffServ-домена (під час входження в домен) виконуються задачі ТСВ:

– класифікація пакетів відповідно до необхідного рівня QoS;
– відповідне маркування поля DSCP у заголовку кожного IP-пакета;
– формування трафіка відповідно до параметрів, які узгоджено в SLA (функції вирівнювання й обмеження, деякі пакети можуть бути відкинуті);
– постановка пакета в чергу для передачі в опорну мережу.

2. В опорній мережі (усередині DiffServ-домену) реалізується PHB-політика:

– формування й обслуговування черг (функція розподілу ресурсів);

– відкидання пакетів з метою попередження виникнення перевантаження.

На рис. 7.2 зображена узагальнена операційна модель QoS [O:5,6].

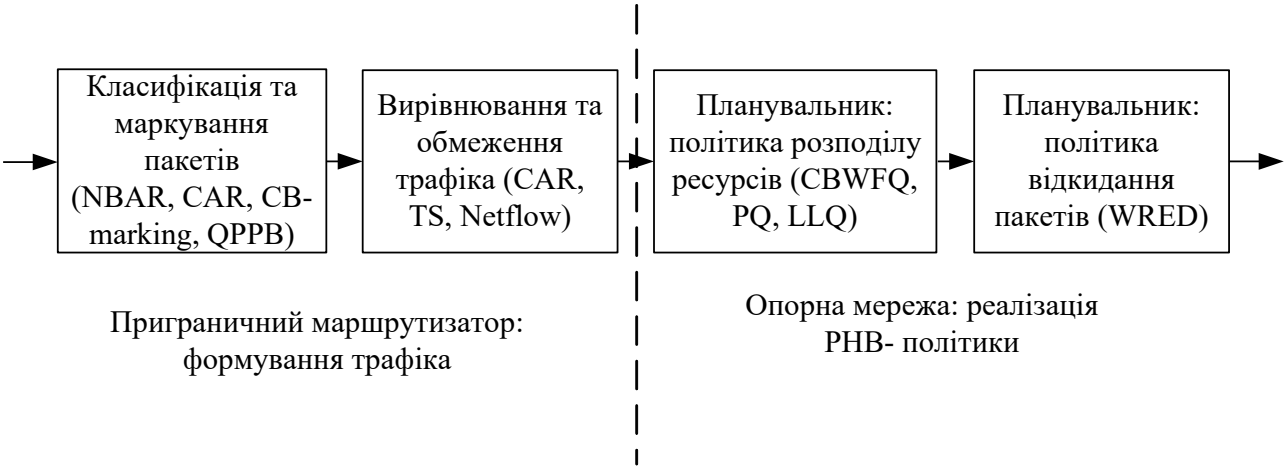

Рисунок 7.2 – Узагальнена операційна модель QoS

В рамках DiffServ не передбачена явна сигналізація про необхідну якість обслуговування, не здійснюється резервування ресурсів і усі вузли, що реалізують PHB-політику, працюють незалежно один від одного.

Архітектура диференційованих послуг визначає усього лише базові механізми, на основі яких здійснюється обслуговування пакетів. В даному випадку послуга визначає досить вагомі характеристики передачі пакетів, такі, як пропускна здатність, затримка, джитер, а також рівень втрат пакетів в одному напрямку при передачі уздовж мережного маршруту. До того ж можна охарактеризувати послугу в термінах відносного пріоритету при доступі до ресурсів мережі. Після визначення послуги розробляються відповідні їй рішення про передавання пакета (PHB-політика) для кожного вузла мережі, що підтримує дану послугу. Наприклад, компанія Cisco пропонує такий набір механізмів для реалізації архітектури диференційованих послуг [O: 5]:

– модульний QoS інтерфейс командного рядка (Cisco Modular QoS command-line interface, MQC), що допомагає спростити задачу

одночасного налаштування декількох інструментів QoS на одному мережному пристрої;

– механізм узгодження швидкості доступу CAR, що виконує функції класифікації і маркування полів IP-пріоритету і QoS-групи, вимірювання трафіка й обмеження інтенсивності трафіка (policing), забезпечуючи тим самим управління смугою пропускання;
– механізм маркування пакетів на основі класу Class-Based Packet Marking, що реалізує однойменну функцію маркування пакетів з метою диференціації трафіка;
– механізм обмеження трафіка Traffic Policing, що дозволяє обмежити швидкість на вхідному або вихідному інтерфейсі на основі сформованих користувачем критеріїв, значень IP-пріоритету, QoS-групи або поля DSCP;
– формування трафіка на основі класу Class-Based Shaping, що дозволяє настроїти GTS (Generic Traffic Shaping) на основі класу, визначити середню і пікову швидкість для кожного класу і конфігурувати CBWFQ у рамках GTS;
– механізм обслуговування черг CBWFQ, що дозволяє гарантувати мінімальне значення смуги, яка виділяється трафіку в період перевантаження на даному інтерфейсі;
– механізм обслуговування черг LLQ, у рамках якого вводиться пріоритетна черга PQ, що використовується для чутливого до затримок трафіка, наприклад, голосового;
– механізм зваженого довільного раннього виявлення перевантажень WRED, що реалізує випадкове відкидання пакетів, де імовірність відкидання розраховується на підставі IP-пріоритету або DSCP. Даний механізм має працювати в парі з CBWFQ;
– реалізація розширення MPLS Class of Service (CoS), що дозволяє постачальникам послуг установлювати поле MPLS Experimental замість IP-пріоритету.

7.1.1. Код диференційованої послуги (DSCP)

Робоча група IETF по створенню і впровадженню диференційованих послуг, що займається питаннями стандартизації поля DSCP, визначила для встановлення коду диференційованої послуги використання 6 біт з байту ToS заголовка IP-пакету. Фактично, кодом диференційованої послуги є розширення 3-бітового поля IP-пріоритету. Так само, як і IP-пріоритет, код диференційованої послуги дозволяє визначати спосіб обробки відмічених відповідним чином пакетів даних і використовується для виконання одних і тих же функцій в diffserv-мережі: маркування пакетів на кордоні мережі і реалізація певної політики обробки пакетів (наприклад, їх відкидання або постановка в чергу) усередині мережі (табл. 7.2) [О: 5,6, Д: 22, 23, 26-30].

Таблиця 7.2 – Порівняльна характеристика різних політик покрокової обробки DiffServ

PHB	**Опис**	**DSCP**
Політика негарантованої доставки пакетів (Best effort, BE)	PHB без надання всіляких гарантій щодо якості обслуговування	DSCP за замовчуванням (000 000)
Селектори класу (Class selector, CS)	Використовується вісім значень DSCP, де завжди останні три біти нульові. Уведені для зворотної сумісності з IP-пріоритетами. Засновані на логіці «більше – краще»: чим більше значення DSCP, тим вище якість обслуговування	CS1, CS2, CS3, CS4, CS5, CS6, CS7
Політика гарантованої передачі (Assured forwarding, AF)	PHB містить два компоненти: планувальник черг забезпечує мінімальну смугу для кожної з чотирьох окремих черг і три рівні порога відкидання для кожної з черг. DSCP не завжди задовольняє правилу «більше – краще»	AF11, AF12, AF13, AF21, AF22, AF23, AF31, AF32, AF33, AF41, AF42, AF43
Політика негайної передачі (Expedited forwarding, EF)	PHB містить два компоненти: планувальник черг забезпечує низький рівень затримки, тремтіння і втрат, а також гарантовану частку смуги; для виділення достатньої смуги необхідна функція обмеження (policing)	EF

Усі коди DCSP, які визначені робочою групою IETF щодо створення та впровадження DiffServ, в залежності від рівня обслуговування (застосовуваної політики PHB) можна розбити на чотири групи (табл. 7.2) [Д: 22]:

– DCSP за замовчуванням або стандартний код DCSP відповідає політиці негарантованої доставки пакетів (best effort), визначений як 000 000;
– DCSP, сумісні з кодами IP-пріоритету, так звані селектори класу (Classs Selector, CS) забезпечують розділення трафіка на 8 класів (рівнів пріоритету), були уведені для сумісності з існуючими IP-системами, що використовують поле пріоритету; мають формат xxx000, де x – бінарні 0 або 1. Ці коди наведені в табл. 7.3 [Д: 22].

Таблиця 7.3 – Коди DSCP

Селектор класу	**DSCP**
Пріоритет 1	001000
Пріоритет 2	010000
Пріоритет 3	011000
Пріоритет 4	100000
Пріоритет 5	101000
Пріоритет 6	110000
Пріоритет 7	111000

DCSP політики негайної передачі (Expedited Forwarding, EF) вказують на найвищий рівень обслуговування, який надається за додаткову вартість. Визначаються в RFC 2598. Рекомендоване значення – 101110.

DCSP політики гарантованої передачі (Assured Forwarding, AF) мають 12 припустимих значень формату AF_{ij}, де i – клас обслуговування ($j = \overline{1,4}$), j – рівень відкидання пакетів ($j = \overline{1,3}$) і описані в RFC 2597. Ці коди DSCP визначають чотири рівні обслуговування, кожен з яких, у свою чергу, може бути охарактеризований трьома рівнями пріоритету відкидання пакетів даних (табл. 7.4).

Таблиця 7.4 – Значення DCSP політики гарантованої передачі

	Низька імовірність відкидання в межах класу	**Середня імовірність відкидання в межах класу**	**Висока імовірність відкидання в межах класу**
Клас 1	AF11/10 / 001010	AF12/12/001100	AF13/14/001110
Клас 2	AF21/18/010010	AF22/20 / 010100	AF23/22/010110
Клас 3	AF31/26/011010	AF32/28/011100	AF33/30 / 011110
Клас 4	AF41/34/100010	AF42/36/100100	AF43/38/100110

На рис. 7.3 і 7.4 показане обслуговування трафіка IP-пакетів у випадку використання селекторів класу CS і у випадку використання політики гарантованої передачі.

Отже, у рамках DiffServ можна виділити три різновиди PHB-політики (рівня обслуговування) [О: 5,6, Д: 22, 23, 26-29]:

– негарантована доставка даних (DCSP за замовчуванням);
– PHB-політика негайної передачі (EF PHB);
– PHB-політика гарантованої доставки (AF PHB).

При реалізації архітектури DiffServ найвищу якість обслуговування (Premium Service), яку можна порівняти з якістю виділених каналів, забезпечує політика негайної передачі пакетів EF PHB.

PHB-політика, що відповідає певному класу трафіку, залежить від цілої низки чинників:

– інтенсивність вхідного потоку або навантаження для заданого класу трафіку. Цей параметр контролюється прикордонним формувачем трафіку;
– розподіл ресурсів для заданого класу трафіку. Цей параметр контролюється функціями розподілу ресурсів, реалізованими у вузлах diffserv-домену;
– рівень втрати трафіку. Цей параметр залежить від політики відкидання пакетів, що проводиться у вузлах diffserv-домену.

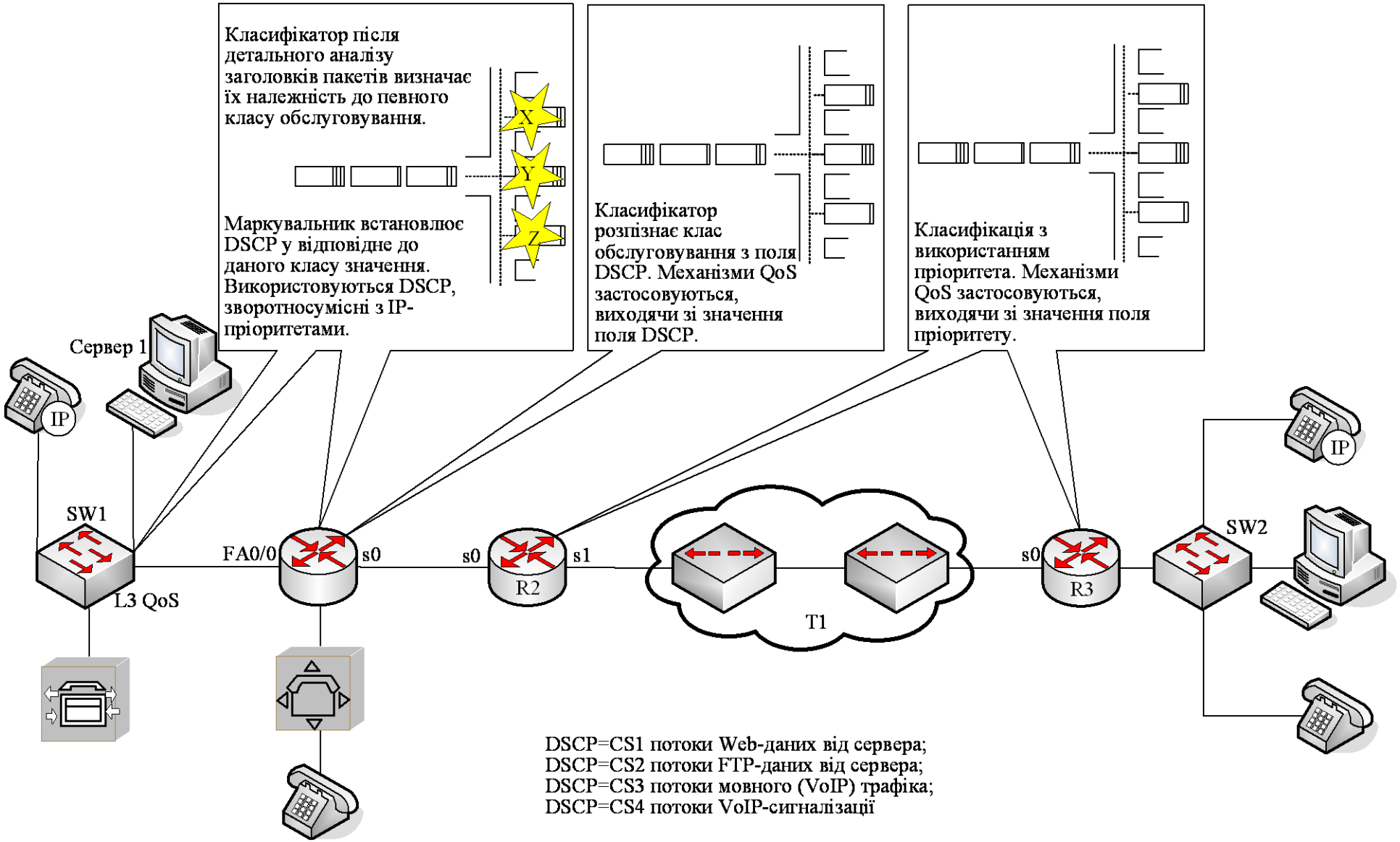

Рисунок 7.3 – Обробка IP-трафіка у випадку використання селекторів класу CS

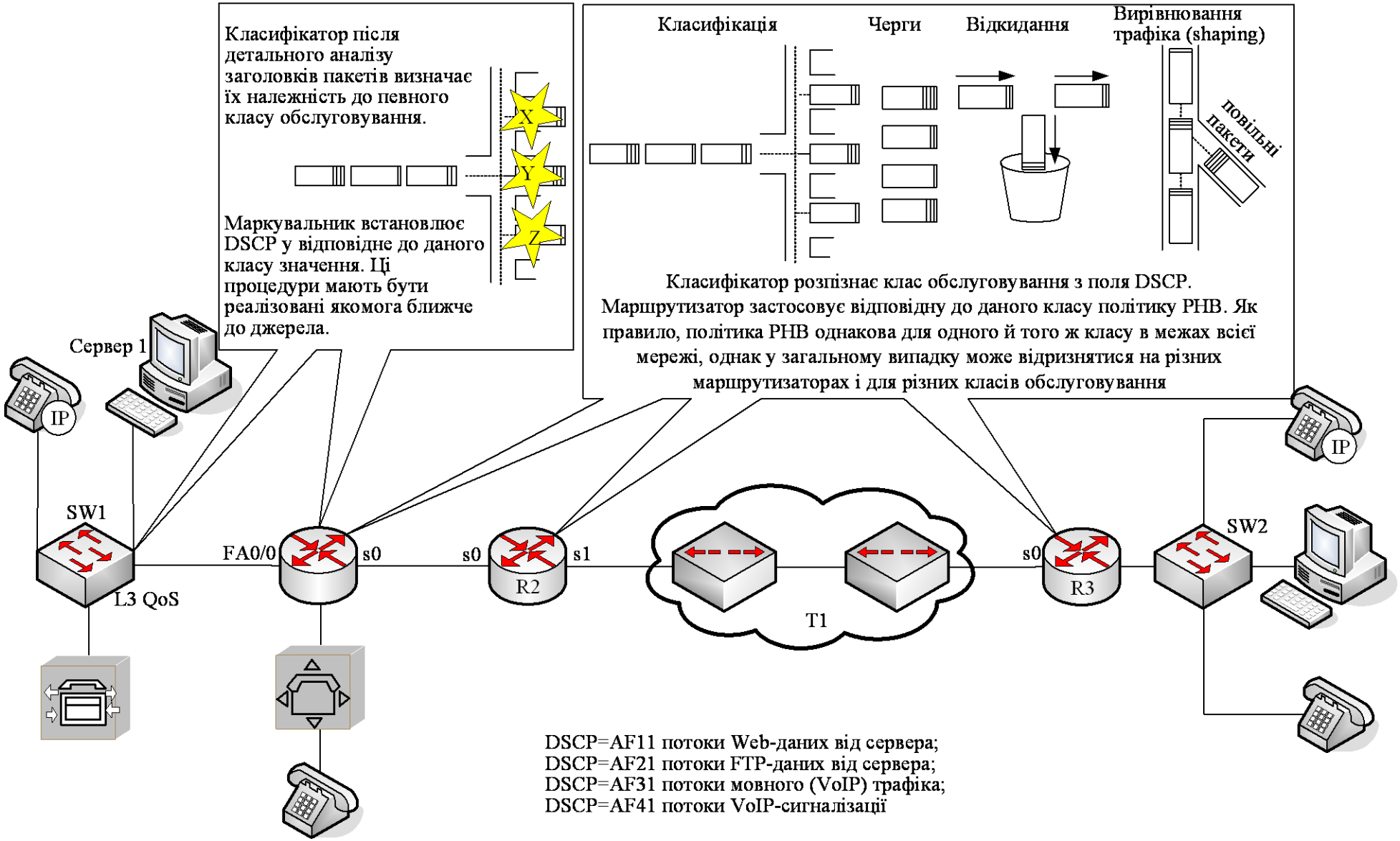

Рисунок 7.4 – Приклад обробки IP-трафіка у випадку використання політики гарантованої передачі

7.1.2. PHB-політика негайної передачі пакетів

PHB-політика негайної передачі пакетів (Expedited Forwarding PHB – EF PHB) використовується для забезпечення наскрізного обслуговування пакетів у вузлах diffserv-домена, характерними рисами якого є низький рівень втрати пакетів, мала затримка, незначне тремтіння трафіку, а також гарантована смуга пропускання [О: 5,6, Д: 28].

Політика EF PHB застосовується для обслуговування трафіку таких застосувань, як передача голосу по мережах IP (Voice over IP – VoIP), програм відеоконференцій, а також для забезпечення таких послуг, як передача інформації по віртуальних каналах, що орендуються, оскільки ця послуга є двоточковим з'єднанням кінцевих вузлів diffserv-домена. Подібний тип обслуговування доволі часто називають також послугами високого класу (premium service).

Головними чинниками, що призводять до великої затримки пакетів і тремтіння трафіку, є затримки, пов'язані з виникненням великих накопичених черг. Подібні черги характерні для переобтяжених ділянок мережі. Причиною перевантаження мережі є переважання інтенсивності вхідного потоку трафіку над інтенсивністю його вихідного потоку.

Один із способів уникнути затримки пакетів, пов'язаної з виникненням великих черг, – обмеження максимальної інтенсивності вхідного потоку трафіку мінімальною інтенсивністю його вихідного потоку. PHB-політика негайної передачі пакетів передбачає установку значення інтенсивності вихідного потоку трафіку, тоді як інтенсивність вхідного потоку контролюється формувачами трафіку, реалізованими в прикордонних пристроях мережі.

Оскільки відповідно до політики EF PHB вхідні пакети не повинні утворювати чергу (допускається черга дуже малого розміру), інтенсивність вихідного потоку трафіку має бути рівній інтенсивності вхідного потоку або перевищувати її. Маршрутизатор може виділити ресурси, достатні для

забезпечення певної інтенсивності вихідного трафіку для заданого інтерфейсу, шляхом використання різних функціональних реалізацій політики EF PHB.

Дана функціональна можливість може бути реалізована за рахунок використання різних механізмів обслуговування черг (зважений механізм рівномірного обслуговування черг на основі класу трафіку, Class-Based Weighted Fair Queuing – CBWFQ), зважений механізм кругового обслуговування (Weighted Round Robin – WRR) і механізм кругового обслуговування з дефіцитом (Deficit Round Robin – DRR). В цьому випадку EF-трафіку призначається вага, яка відповідає смузі пропускання, що набагато перевищує інтенсивність вхідного EF-трафіка.

7.1.3. PHB-політика гарантованої доставки пакетів (AF PHB)

PHB-політика гарантованої доставки пакетів (Assured Forwarding PHB – AF PHB) є засіб, за допомогою якого постачальник послуг може забезпечити декілька різних рівнів надійності доставки IP-пакетів, отриманих з difrserv – домена клієнта [О: 5,6, Д: 27]. Політика AF PHB є прийнятною для більшості TCP-прикладних програм.

PHB-політика гарантованої доставки пакетів передбачає наявність різних рівнів обслуговування для кожного з чотирьох класів AF-трафіка (AF_1, AF_2, AF_3 і AF_4 – табл. 7.4). Кожному класові виділяється певна кількість ресурсів (буферного простору і смуги пропускання) відповідно до укладеної раніше угоди SLA або прийнятої в мережі політики.

У межах кожного класу вводиться три рівні відкидання пакетів. Якщо при перевантаженні виникне необхідність відкинути пакети якого-небудь класу, наприклад, AF_1, імовірність їхнього відкидання задовольнятиме такому правилу:

$$P_d(AF_{13}) \geq P_d(AF_{12}) \geq P_d(AF_{11}),$$

де $P_d(AF_{13})$ – імовірність відкидання пакета класу обслуговування AF_{13}.

Іншими словами, в першу чергу буде відкинутий пакет класу AF_{13}, потім класу AF_{12}, потім AF_{11}. Відкидання відбувається при перевищенні класом (у даному випадку AF_1) виділеної для нього смуги. Пакети потоку-порушника можуть маркуватися з підвищенням імовірності відкидання.

Рівень обслуговування AF PHB має виділений йому діапазон припустимих значень DSCP. Поле DSCP у цьому випадку має формат *xyzab*0, де *xyz* можуть бути 001, 010, 011 або 100, а значення *ab* відображає значення P_d.

7.1.4. Методи формування трафіку на границі мережі для реалізації PHB-політик

Існує три рішення для формування трафіку на границі мережі і розподілу ресурсів в її вузлах для досягнення певної PHB-політики: ініціалізація мережі, сигналізація про якість обслуговування і диспетчер політик [O:5,6].

Ініціалізація мережі. Один із способів розподілу ресурсів полягає в ініціалізації ресурсів мережі з використанням евристичних методів. Слід зазначити, що цей метод може бути застосований тільки в мережах невеликого розміру, для яких політики QoS і профілі трафіку залишаються незмінними впродовж достатньо довгого проміжку часу.

Сигналізація про якість обслуговування. Відповідно до цього методу реалізації PHB-політики прикладні програми сповіщають мережу про вимоги до якості обслуговування за допомогою сигнального протоколу RSVP. За допомогою протоколу RSVP можна встановити відповідність між запитами до якості обслуговування і класами послуг diffserv. Наприклад, гарантованому обслуговуванню RSVP може бути поставлена у відповідність diffserv-послуга негайної передачі пакетів. Встановлення відповідності між резервуванням ресурсів протоколу RSVP і diffserv-класами проводиться на кордоні diffserv-мережі. Це рішення добре підходить для реалізації в крупних корпоративних мережах.

Диспетчер політик. QoS-політики в мережі можуть бути налагоджені шляхом внесення певних змін до конфігурації маршрутизаторів. Проте, у великій мережі цей процес стає надзвичайно складним і практично

некерованим. Для того, щоб у такій мережі забезпечити наскрізну якість обслуговування, політики, реалізовані у всіх її вузлах, мають бути несуперечливі. Отже, для швидкої і ефективної реалізації QoS-політик у всіх вузлах мережі потрібна наявність централізованого диспетчера політик. Зазвичай, такий диспетчер політик реалізує спільну відкриту службу політик (Common Open Policy Service – COPS) – це протокол розповсюдження політик, розроблений групою IETF. У термінології протоколу COPS централізований сервер політик називається точкою визначення політики (Policy Decision Point – PDP). Вузол мережі, якому нав'язується політика, отримав назву точки вживання політики (Policy Enforcement Point – PEP). PDP-сервер використовує протокол COPS для завантаження політик в PEP-вузли мережі. PEP-пристрій може згенерувати повідомлення, в якому він інформує PDP-сервер про неможливість реалізації запропонованої PDP-сервером політики.

7.2. Архітектура інтегрованих послуг IntServ

Метою цієї архітектури є підтримка QoS, у першу чергу для мережних додатків реального часу, яка б забезпечувала управління затримкою передачі пакета, а також управління розподілом смуги між потоками трафіка. Термін «інтегровані послуги» відповідно до специфікації RFC 1633 [Д: 31]. містить у собі існуючу раніше доставку best effort (негарантована доставка даних), а також додає доставку потоків трафіка реального часу з гарантованою затримкою і доставку пакетів з гарантованою швидкістю.

Основними функціональними блоками моделі IntServ є резервування ресурсів (resource reservation) і управління доступом (admission control). У рамках архітектури IntServ зроблено акцент на процесі сигналізації, за допомогою якого індивідуальні потоки повідомляють про свої вимоги щодо обсягу смуги, який потрібно зарезервувати, і припустимої величини затримки [О: 5,6, Д: 31]. Як протокол сигналізації в моделі IntServ передбачається використання протоколу резервування ресурсів RSVP (RFC 2205-2215) [Д: 24,25].

Процесові резервування передує процес управління доступом, що на підставі аналізу доступних мережних ресурсів приймає рішення про прийняття потоку до обслуговування (якщо ресурсів досить) або відхилення запиту (за нестачі ресурсів). Обов'язковою умовою прийняття запиту до обслуговування є непогіршення якості обслуговування раніше прийнятих запитів. Функція управління доступом покладається на COPS-сервер.

Отже, ключовим поняттям у IntServ є резервування ресурсів. Коли мова йде про резервування смуги для потоку трафіка, то це означає таку конфігурацію механізму обслуговування черг, що при обслуговуванні черги з даним потоком йому надається така смуга, яка запитується. Коли мова йде про забезпечення припустимої величини затримки, то тут все трохи складніше. Наприклад, IOS Cisco забезпечує низьку затримку шляхом резервування місця в черзі. Взагалі в концепції IntServ (RFC 1633) не обумовлюється спосіб забезпечення резервування ресурсів, залишаючи це питання розробникам обладнання.

Реалізація моделі IntServ згідно з RFC 1633 вимагає наявності в маршрутизаторі таких функціональних блоків (рис. 7.5) [О: 5,6, Д: 31]:

- класифікація (ідентифікація) потоків даних з метою визначення їхньої належності певному класу обслуговування;
- механізм обслуговування черги, а також механізми визначення перевантаження і відкидання пакетів з низьким рівнем пріоритетного обслуговування;
- управління доступом до ресурсів мережі з метою визначення можливості обслуговування запиту з заданим рівнем QoS на підставі аналізу наявності необхідного обсягу ресурсів;
- механізм резервування ресурсів.

Функції управління доступом, класифікації і планувальника можуть бути об'єднані в блок управління трафіком (traffic control), головна задача якого полягає в створенні різниці щодо якості обслуговування [О: 5,6, Д: 31].

Головною перевагою даної моделі є забезпечення твердих гарантій щодо якості обслуговування: мережний додаток отримує той обсяг ресурсів, який йому необхідний. Крім того, це сприяє ефективному використанню ресурсів мережі.

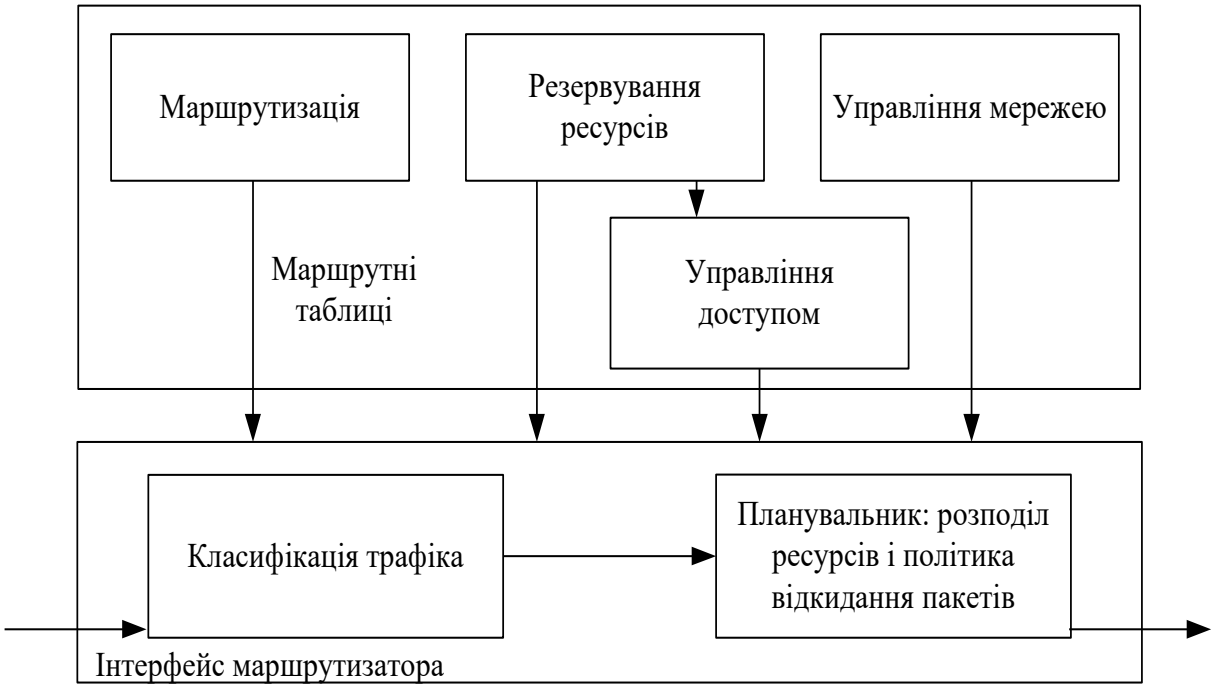

Рисунок 7.5 – Функціональні блоки моделі IntServ

Однак моделі IntServ властиві серйозні недоліки, завдяки яким вона не одержала широкого поширення на практиці. Перший і найголовніший недолік полягає в низькій масштабованості, що пов’язано із обслуговуванням кожного потоку окремо.

Другий недолік пов’язаний із утратою якості обслуговування при проходженні через ділянку мережі, у якій не підтримується IntServ. Тут відбувається або передача з якістю best effort, або відображення запиту на класи DiffServ (DSCP) при проходженні через DiffServ-домен.

7.2.1. Протокол сигналізації RSVP

ReSerVation Protocol (RSVP) – стандартизований протокол, призначений для динамічного налаштування наскрізного QoS у рамках гетерогенної мережі. RSVP дозволяє мережним додаткам посилати в мережу сигнали про свої QoS-вимоги для кожного потоку і динамічно резервувати необхідну для них

смугу пропускання. Можна дати таке визначення RSVP – це протокол сигналізації, що забезпечує резервування ресурсів і управління ними з метою надання інтегрованих сервісів, призначених для емуляції виділених каналів у IP-мережах [О: 5, 6, Д: 24, 25, 30].

RSVP працює на мережному рівні, як і протокол IP, і має свій власний тип протоколу – 46, хоча можлива інкапсуляція повідомлень RSVP у датаграми UDP. Для такої інкапсуляції використовуються UDP-порти 1698 і 1699.

Спочатку протокол RSVP розроблявся для мультимедійного трафіка (аудіо- та відео-), орієнтуючись на групове мовлення. Однак RSVP успішно застосовується для резервування ресурсів для односпрямованого трафіка, наприклад, трафіка мережної файлової системи (Network File System, NFS) і управляючого трафіка віртуальних приватних мереж (Virtual Private Network, VPN).

Протокол RSVP сигналізує про запити резервування ресурсів доступним шляхом в мережі, який маршрутизується. При цьому RSVP не виконує власну маршрутизацію, а використовує для цього інші протоколи маршрутизації. Подібна модульність допомагає протоколу RSVP ефективно функціонувати разом з будь-якою службою надання інформації про маршрути. RSVP забезпечує явну передачу повідомлень для управління трафіком та політиками.

7.2.2. Принципи функціонування протоколу RSVP

Кінцеві системи використовують протокол RSVP для запиту у мережі певного рівня QoS від імені потоку даних прикладної програми.

Функціонування RSVP пов’язане з виконанням таких задач [О: 5, 6, Д: 24, 25, 30]:

– резервування ресурсів і підтримка резервування;
– звільнення зарезервованих ресурсів;
– сигналізація про помилки.

Резервування ресурсів здійснюється шляхом посилання в мережу RSVP-запитів від імені потоку даних, в яких закладено інформацію про необхідний рівень QoS.

RSVP-запити передаються уздовж наперед визначеного маршруту і на кожному із пройдених вузлів (маршрутизаторів) здійснюється спроба зарезервувати необхідні ресурси. Для цього в кожному з маршрутизаторів мережі необхідна реалізація RSVP-агентів (демонів), взаємодія яких з іншими функціональними блоками відображена на рис. 7.6 (RFC 2205).

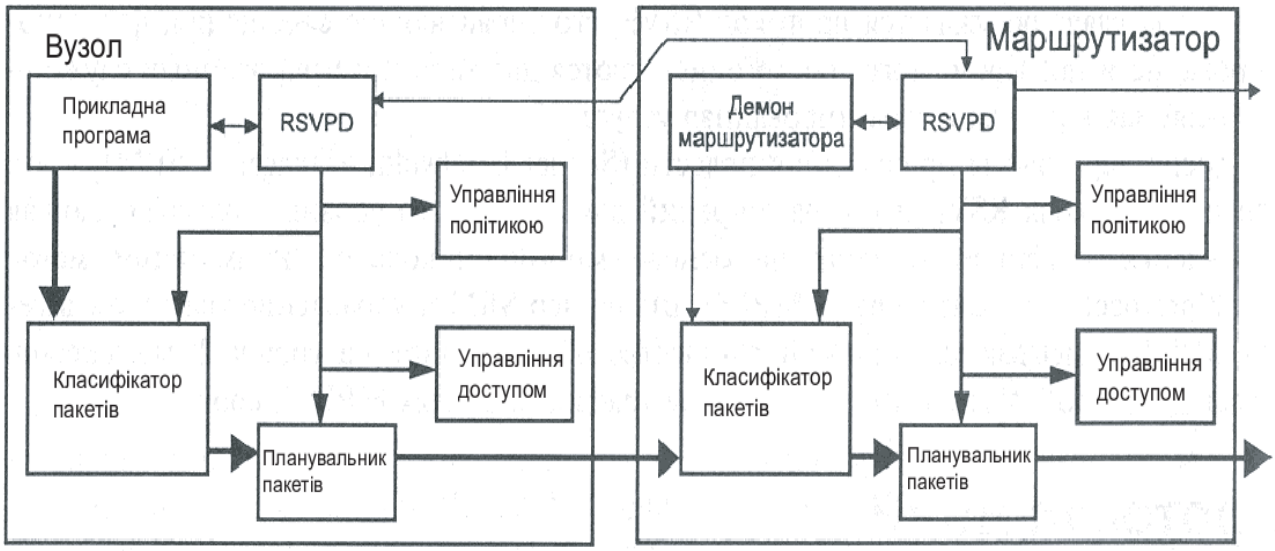

Рисунок 7.6 – Функціональні блоки RSVP

Перш ніж зарезервувати ресурси, RSVP-демон маршрутизатора з'єднується з двома локальними модулями ухвалення рішення – модулем управління доступом (admission control) і модулем управління політикою (policy control) [О: 5, 6, Д: 24, 25, 30].

Модуль управління доступом визначає, чи має вузол досить вільних ресурсів для забезпечення запитаного рівня QoS. Модуль управління політикою визначає, чи є у користувача адміністративні права, для того, щоб провести резервування. Якщо яка-небудь з перевірок не пройшла, RSVP-демон відправляє повідомлення про помилку процесу прикладної програми, який створив запит. Якщо обидві перевірки пройшли нормально, RSVP-демон встановлює параметри класифікатора пакетів (packet classifier) і планувальника пакетів (packet scheduler) для отримання потрібного рівня QoS.

Класифікатор пакетів визначає клас QoS для кожного пакету, а планувальник пакетів управляє передачею пакетів, ґрунтуючись на їх класі

QoS. Зважений алгоритм рівномірного обслуговування черг (Weighted Fair Queuing– WFQ) і зважений алгоритм довільного раннього виявлення (Weighted Random Early Detection – WRED) забезпечують підтримку QoS на рівні планувальника [О: 5, 6, Д: 24, 25].

Під час процесу прийняття рішення модулем управління доступом резервування смуги пропускання, що замовляється, проводиться тільки в тому випадку, якщо для запрошуваного класу трафіку досить частки, що залишилася. Інакше запит на доступ відхиляється, але трафік все одно передається з якістю обслуговування, визначеною за замовчуванням для даного класу трафіку.

У багатьох випадках, навіть якщо запит на доступ був відхилений на одному або декількох маршрутизаторах, модуль все ще може реалізувати прийнятну якість обслуговування, встановивши резервування на перевантажених маршрутизаторах. Це можливо через те, що інші потоки даних можуть не повністю використовувати замовлену ними смугу пропускання.

Резервування завжди повинне слідувати по одному і тому ж наперед визначеному маршруту. В разі виходу з ладу лінії зв’язку маршрутизатор повинен повідомити про це RSVP-демон, щоб RSVP-повідомлення, що генеруються ним, передавалися по новому маршруту.

Процес **встановлення резервування** складається з п’яти окремих кроків [О: 5, 6, Д: 24, 25].

1. Джерело даних посилає за унікальною або груповою адресою одержувача спеціальне повідомлення PATH, у якому воно вказує параметри свого трафіка – специфікацію потоку даних відправника (Sender_TSpec). Повідомлення PATH передається маршрутизаторами мережі у напрямку від джерела (або джерел) з використанням таблиць маршрутизації, які отримані за допомогою будь-якого протоколу маршрутизації.

2. Кожен RSVP-маршрутизатор перехоплює PATH-повідомлення, зберігає IP-адресу попередньої точки призначення, записує замість нього свою власну адресу і відправляє оновлене повідомлення далі тим же шляхом, яким

передаються дані. Тим самим у мережі утвориться фіксований маршрут передачі повідомлень у рамках сесії RSVP.

3. Після одержання повідомлення PATH приймач відправляє маршрутизаторові, від якого він одержав це повідомлення, запит резервування ресурсів – повідомлення RESV (reservation request). До повідомлення RESV крім інформації Receiver_TSpec (специфікація потоку даних) включається специфікація запиту (RSpec), в якій указуються необхідні приймачеві параметри якості обслуговування, і специфікація фільтра (FilterSpec), що визначає, до яких пакетів сесії застосовується дане резервування (наприклад, за типом транспортного протоколу і номером порту). Разом RSpec і FilterSpec складають дескриптор потоку, який маршрутизатор використовує для ідентифікації кожного резервування ресурсів (RFC 2205). Фільтр може визначити потік пакетів з будь-яким ступенем деталізації і на підставі будь-якої інформації, у тому числі і прикладному рівні. Запитувані параметри QoS у специфікації RSpec можуть відрізнятися від зазначених у специфікації TSpec.

4. Коли кожен маршрутизатор, який підтримує RSVP, уздовж зворотного шляху одержує повідомлення RESV, то він визначає прийнятність зазначених у запиті параметрів резервування як було описано вище з використанням процесів управління доступом і управління політикою. Якщо запит не може бути задоволений (через недолік ресурсів або помилки авторизації), маршрутизатор повертає повідомлення про помилку джерелу. Якщо запит приймається, то маршрутизатор посилає повідомлення RESV уверх наступному маршрутизаторові (у напрямку джерела). Прийом запиту резервування маршрутизатором означає також передачу параметрів QoS на відпрацьовування у відповідні блоки маршрутизатора. Конкретний спосіб відпрацьовування параметрів QoS маршрутизатором у протоколі RSVP не описується. Якщо технологія канального рівня, що обслуговує вихідний інтерфейс, не підтримує управління якістю обслуговування, то відпрацьовування виконується механізмом управління чергами, таким як WFQ або CQ. Якщо ж ця технологія підтримує QoS, то параметри специфікації RSpec відображаються на параметри QoS даної технології.

5. Коли останній маршрутизатор одержує повідомлення RESV і приймає запит, то він посилає повідомлення-підтвердження назад вузлу-одержувачу (останній маршрутизатор розташований найближче до джерела, а для групових потоків – це точка об'єднання резервування). При виконанні групового резервування враховується той факт, що в точках розгалуження дерева доставки декілька потоків резервування зливаються в один. Якщо для всіх потоків запитується однакова пропускна здатність, то вона потрібна і для загального потоку. Якщо запитуються різні величини пропускної здатності, то для загального потоку обирається максимальна.

Після встановлення резервування джерело починає відправляти дані, що обслуговуються на всьому шляху до приймача (приймачів) із заданою якістю обслуговування.

Механізм RSVP-резервування схематично показаний на рис. 7.7.

По мережі, що складається з RSVP-сумісних маршрутизаторів (RSVP-enabled router network), прокладається шлях між RSVP-відправниками і RSVP-одержувачами.

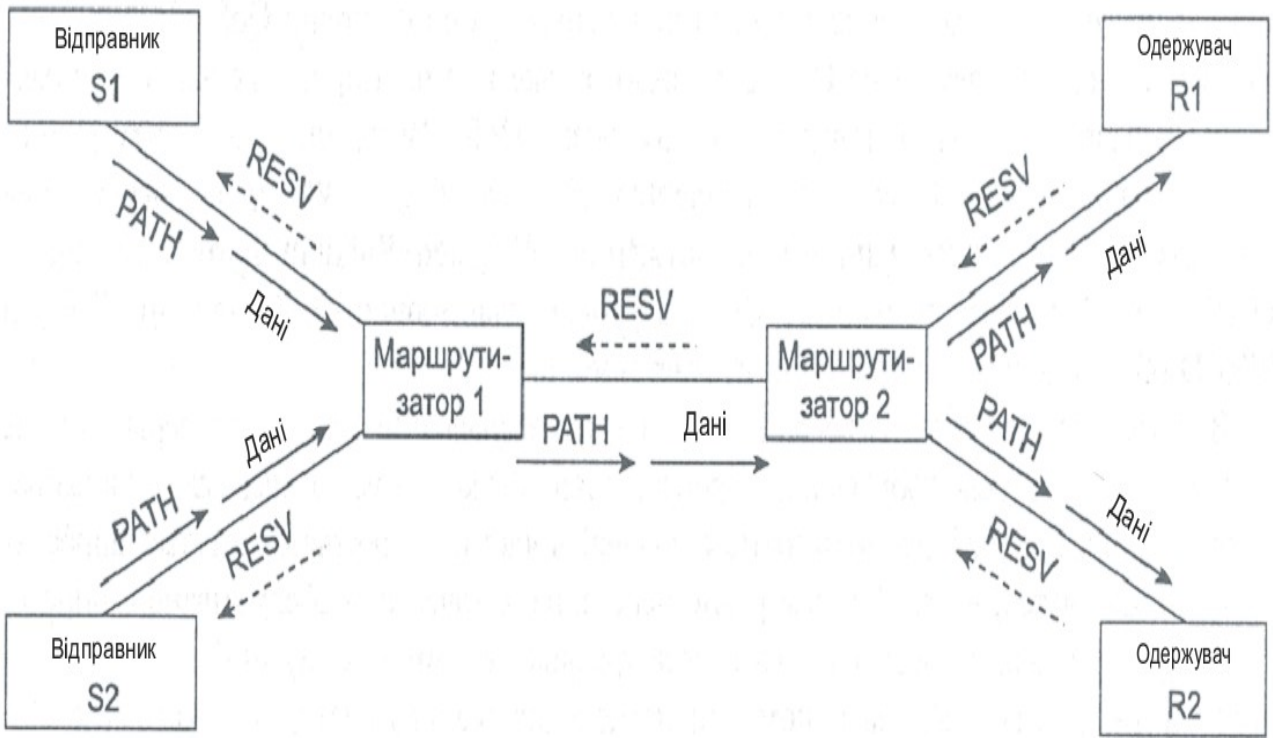

Рисунок 7.7 – Механізм RSVP-резервування

RSVP-компоненти

RSVP-компоненти виконують такі функції.

RSVP-джерело (RSVP sender) ініціює відправлення трафіка в RSVP-сеансі. RSVP-джерела передають параметри свого трафіка – специфікацію потоку даних відправника Sender_TSpec – проміжним маршрутизаторам і RSVP-одержувачеві. Специфікація Sender_TSpec є частиною об'єкта RSVP SENDER_TSPEC повідомлення PATH (див. нижче). Під час передавання RSVP-мережею зміст специфікації Sender_TSpec не змінюється.

Нижче перераховані специфікації потоку, які RSVP-відправники можуть передавати по RSVP-мережі [О: 5, 6, Д: 24, 25, 30]:

– середня швидкість передачі даних;
– максимальний розмір сплеску.

RSVP-одержувач (RSVP-receiver) одержує трафік у RSVP-сеансі. У процесі резервування ресурсів у відповідь на повідомлення PATH формує повідомлення, до якого включено об'єкт RSVP FLOWSPEC (див. нижче), що містить інформацію Receiver_TSpec і RSpec. Під час передавання RSVP-мережею зміст даних специфікацій може змінюватися в результаті злиття декількох RSVP-запитів або з інших причин.

Нижче перераховані специфікації потоку, які RSVP-одержувачі можуть передавати по RSVP-мережі [О: 5, 6, Д: 24, 25, 30]:

– середня швидкість передачі даних.
– максимальний розмір сплеску.
– QoS, включаючи:
  – гарантоване обслуговування – в PATH-повідомленнях також описуються максимально можливі затримки в мережі;
  – обслуговування з керованим навантаженням – маршрутизатори гарантують тільки те, що мережеві затримки будуть мінімальними.

Специфікації потоку даних Sender_TSpec та Receiver_TSpec містять у собі таку інформацію про трафік (RFC 2210): середню швидкість передачі даних, розмір «кошика маркерів» (узгоджений розмір сплеску), пікову швидкість

(максимальний розмір сплеску), мінімальний і максимальний розміри пакетів. Специфікація RSpec містить вимоги щодо виділених ресурсів у термінах смуги пропускання і затримки.

Підтримка резервування. RSVP є протоколом гнучких станів (soft-state). Це означає, що потрібне періодичне оновлення резервування в мережі шляхом посилання додаткових повідомлень, чим власне протоколи даного типу відрізняються від hard-state протоколів, в яких запит посилається один раз і виконується доти, поки не буде явно скасований.

Для відновлення резервування усі компоненти, що беруть участь у резервуванні (RSVP-джерело, RSVP-одержувач, RSVP-сумісні маршрутизатори), періодично через кожні 30 с відсилають своїм сусідам повідомлення PATH (у прямому напрямку) і RESV (у зворотному напрямку). Якщо маршрутизатор відправляє чотири повідомлення PATH підряд і не одержує протягом цього часу жодного повідомлення RESV, резервування вважається втраченим і одержувачеві відправляється повідомлення про розрив з'єднання [О: 5, 6, Д: 24, 25, 30].

Скасування резервування. Резервування можна скасувати прямо або непрямо. Пряме скасування виконується з ініціативи RSVP-джерела або RSVP-одержувача за допомогою відповідних повідомлень протоколу RSVP (PATHTEAR уздовж шляху, яким передавалося повідомлення PATH, і RESVTEAR уздовж шляху, яким передавалося повідомлення RESV). Повідомлення RESVTEAR відправляється у відповідь на отримане PATHTEAR, що сигналізує про те, що RSVP-одержувач скасував резервування. Cisco IOS Software очікує від RSVP-джерела відсилання RSVP-одержувачеві підтвердження про одержання RESVTEAR – повідомлення RESVTEARCONF.

Непряме скасування відбувається за тайм-аутом: стан резервування має термін життя, і RSVP-компоненти мають періодично підтверджувати резервування. Якщо ж повідомлення-підтвердження перестають надходити, то резервування скасовується після закінчення його терміну життя [О: 5, 6, Д: 24, 25, 30].

Сигналізація помилок. В процесі поширення мережею або в процесі обробки RSVP-повідомлень іноді виникають помилки. Зазвичай це відбувається через порушення цілісності повідомлення. Для повідомлення RSVP-компонентів про виникнення помилок використовуються повідомлення PATHERR і RESVERR.

7.2.3. RSVP-повідомлення

У протоколі RSVP використовуються сім типів повідомлень: два обов'язкових – PATH і RESV і п'ять опціональних – PATH ERROR, PATH TEARDOWN, RESV ERROR, RESV CONFIRM і RESV TEARDOWN [О: 5, 6, Д: 24, 25, 30]. У табл. 7.5 наведено типи повідомлень, які використовуються в RSVP.

Таблиця 7.5 – Типи RSVP-повідомлень

Повідомлення	**Функція**	**Напрямок**	**Адреса призначення**
PATH	Використовується для ініціалізації встановлення і підтримки резервування	Униз (до одержувача)	Вузол-одержувач
RESV	Інформує про успішне проходження повідомлення PATH; резервує ресурси	Уверх (до джерела)	Наступний вузол
PATHERR	Посилається в напрямку джерела у випадку виявлення помилки в PATH (наприклад, коли розірване з'єднання або пакет PATH пошкоджений)	Уверх	Наступний вузол
RESVERR	Посилається в напрямку до кінцевого вузла, якщо в процесі обробки повідомлення RESV виявлена помилка	Униз	Наступний вузол
PATHTEAR	Посилається до кінцевого вузла для розриву існуючого резервування	Униз	Вузол-одержувач
RESVTEAR	Посилається в напрямку до вузла-джерела для розриву існуючого резервування	Уверх	Вузол-джерело
RESVCONF	Посилається у відповідь на повідомлення RESV або RESVTEAR для запиту підтвердження одержання повідомлення	Униз	Вузол-одержувач
RESVTEARCONF (Cisco)	Посилається в підтвердження на RESVTEAR	Униз	Вузол-одержувач
HELLO (RFC 3209)	Посилається сусіднім RSVP- вузлам з якими існує пряме з'єднання з метою підтримки встановленого з'єднання	Уверх / униз	Наступний вузол

Документом RFC 2205 визначено сім типів повідомлень: PATH, RESV, PATHERR, RESVERR, PATHTEAR, RESVTEAR, RESVCONF. Повідомлення HELLO введене в RFC 3209 як розширення RSVP і призначене для підтримки встановленого резервування. Повідомлення RESVTEARCONF не описане в документах RFC, а є запатентованою компанією Cisco розробкою.

Повідомлення RSVP мають чіткий формат: кожне повідомлення RSVP обов'язково має заголовок (рис. 7.8) і наступний за ним один або кілька об'єктів.

Рисунок 7.8 – Формат RSVP-повідомлення

У табл. 7.6 описані поля заголовка RSVP-повідомлення.

Таблиця 7.6 – Поля заголовка RSVP-повідомлення

Поле	**Опис**
Версія (Version)	Версія протоколу RSVP. Поточна версія 1
Прапорці (Flags)	Прапорці не визначені
Тип повідомлення (Message Type)	1 = повідомлення PATH 2 = повідомлення RESV 3 = повідомлення PATHERR 4 = повідомлення RESVERR 5 = повідомлення PATHTEAR 6 = повідомлення RESVTEAR 7 = повідомлення RESVCONF 10 = повідомлення RESVTEARCONF 20 = повідомлення HELLO
RSVP Checksum	Контрольна сума в RSVP-повідомленні
TTL передачі	Вказує на те, з яким значенням позначки часу життя (TTL) відправлений даний IP-пакет
Зарезервоване (Reserved)	Не використовується
Довжина (RSVP Length)	Довжина RSVP повідомлення в байтах, включаючи заголовок. Мінімальна довжина 8 байт

Що стосується об’єктів, які розміщуються після заголовка, то їхня кількість і типи залежать від призначення повідомлення. Різними документами RFC визначено 23 класи об’єктів RSVP. Всі об’єкти мають однаковий формат (RFC 2205), що складається з заголовка (32 біта) і змісту об’єкта (рис. 7.9).

Рисунок 7.9 – Формат об’єктів RSVP-повідомлення

У відповідності зі стандартним форматом у заголовку кожного об’єкта вказується його загальна довжина в байтах (величина обов’язково кратна чотирьом); номер класу об’єкта (кожен з 23 класів об’єктів має як свою назву, так і номер); С-тип (тип об’єкта, унікальний в рамках одного класу). Номер класу і С-тип використовуються разом як 16-бітовий ідентифікатор унікального типу для кожного об’єкта.

Об’єкт несе в собі різну інформацію, наприклад, про стиль резервування ресурсів (клас STYLE), специфікацію потоку даних, переданих джерелом (клас Sender_TSpec), специфікацію потоку даних і вимог до ресурсів, запитуваних одержувачем (клас Flowspec, RFC 2210) [Д:32]. Об’єкт розміщується в повідомленні з урахуванням його типу, наприклад, об’єкт класу Sender_TSpec розміщається тільки в PATH-повідомленнях, об’єкт класу Flowspec тільки в повідомленнях RESV, RESVERR, RESVTEAR, RESVCONF, RESVTEARCONF.

7.2.4. Стилі резервування протоколу RSVP

Запит на резервування містить у собі набір опцій, що у сукупності називається стилем. Одна опція резервування визначає спосіб резервування різними джерелами в рамках однієї сесії : індивідуальне (distinct) резервування або загальне (shared) резервування. Інша опція резервування контролює вибір джерел: явний (explicit) або довільний (wildcard) вибір [О: 5, 6, Д: 24, 25, 30].

У випадку явного вибору кожному джерелу ставиться у відповідність певна специфікація фільтра, у випадку довільного вибору – таких специфікацій не потрібно зовсім.

В даний час визначені наступні три стилі резервування [О: 5, 6, Д: 24, 25, 30] (табл. 7.7).

Таблиця 7.7 – Стилі резервування

Кількість джерел	**Стилі резервування**	
	Індивідуальне	**Загальне**
Явне резервування	FF (резервування з фіксованим фільтром)	SE (загальне явне резервування)
Групове резервування	Не визначено	WF (резервування з груповим фільтром)

Індивідуальне резервування (distinct reservations) застосовується в тих прикладних програмах, в яких декілька джерел даних можуть відправляти інформацію одночасно. У відео програмах кожен відправник генерує індивідуальний потік даних, для якого необхідно здійснювати окреме управління доступом і планування черги на всьому шляху до одержувача. Отже, для такого потоку необхідно здійснювати окреме резервування ресурсів для кожного відправника і для кожного каналу на маршруті.

Індивідуальне резервування відбувається явно для відправника і встановлюється за допомогою стилю резервування з фіксованим фільтром (Fixed Filter – FF). У символьному вигляді запит на резервування в стилі FF можна представити як FF(S{Q}) , де S – це відправник, а Q – об’єкт FlowSpec. Пара S{Q} є дескриптором потоку. RSVP дозволяє застосовувати стиль резервування FF одночасно для декількох потоків, при цьому формується список дескрипторів потоків FF (S1{Q1}, S2{Q2}, ...). Повне резервування в каналі для даної сесії характеризується сумою Q1, Q2, ... для усіх відправників.

Найпростіший випадок індивідуального резервування ресурсів спостерігається на прикладі прикладної програми з одноадресним трафіком, де є тільки один відправник і один одержувач.

Загальне резервування (shared reservations) застосовується в тих прикладних програмах, в яких декілька джерел даних не передають інформацію одночасно, наприклад цифрові аудіо-програми, такі, як програми VoIP. У цьому випадку, оскільки в будь-який окремо узятий проміжок часу розмову веде невелика кількість людей, інформація передається лише невеликою обмеженою кількістю відправників. Такий потік не потребує окремого резервування ресурсів для кожного відправника, для нього необхідне всього лише одне резервування, яке при необхідності можна буде застосувати до будь-якого відправника в групі.

В термінах протоколу RSVP такий потік називається загальним потоком (shared flow); він встановлюється за допомогою загального явного (Shared Explicit – SE) або групового (Wildcard Filter – WF) резервування.

При загальному явному (Shared Explicit – SE) резервуванні потоки, що резервують мережні ресурси, вказуються окремо. Іншими словами, створюється резерв ресурсів, що використовується спільно декількома відправниками, перелік яких задається безпосередньо.

У символьному вигляді запит на резервування в стилі SE можна представити як

SE{(S1,S2){Q}},

де S1, S2 ... – окремі відправники, що вимагають резервування ресурсів, а Q – об’єкт FlowSpec.

За допомогою групового фільтру (Wildcard Filter – WF) смуга пропускання і характеристики затримки можуть бути зарезервовані для будь-якого відправника. Такий фільтр не дозволяє вказати відправників окремо – він приймає всіх відправників, на що вказує установка адреси джерела і порта в нуль. Резервування здійснюється в найбільшому серед запитаних одержувачами обсязі ресурсів, що не залежить від кількості відправників. Зарезервований груповий ресурс поділяється поміж потоками усіх відправників.

У символьному вигляді запит на резервування в стилі WF можна представити як WF (*{Q}), де символом «*» є груповий символ вибору відправників, а Q – об'єкт FlowSpec.

Використання стилю резервування FF аналогічно встановленню з'єднання «точка – точка», SE і WF – «група точок – точка».

Стиль резервування вказується в об'єкті класу STYLE повідомлень RESV, що передаються в напрямку від одержувача до джерела. При об'єднанні запитів на резервування як результуючою необхідною смугою обирається найбільша величина з усіх запитаних одержувачами.

7.2.5. Типи інтегрованих послуг, які надаються RSVP

Протокол RSVP надає два типи інтегрованих послуг, які одержувачі можуть запрошувати за допомогою повідомлень RSVP RESV: службу регульованого навантаження (Controlled-Load Service, RFC 2211) [Д: 33] і службу гарантованого обслуговування (Guaranteed Service, RFC 2212) [Д: 34] (служба гарантованої бітової швидкості – guaranteed bit rate service) .

Регульоване навантаження

Служба регульованого навантаження (controlled load service) забезпечує гарантію того, що зарезервований потік досягне свого пункту призначення з мінімальним втручанням з боку трафіку, що доставляється без гарантій. Більш того, в реалізації цієї послуги компанією Cisco передбачена ізоляція окремих зарезервованих потоків. Ізоляція потоку дозволяє виключити вплив інших присутніх в мережі зарезервованих потоків при резервуванні ресурсів.

Як правило, служба регульованого навантаження застосовується при передачі трафіку Internet – прикладних програм, чутливих до перевантажень в мережі. Такі прикладні програми відмінно працюють в ненавантажених мережах, але відразу «приходять в непридатність» при перевантаженні. Прикладом таких застосувань може служити прикладна програма, що працює по протоколу FTP (File Transfer Protocol – протокол передачі файлів).

Гарантоване обслуговування

Служба гарантованого обслуговування (guaranteed service) забезпечує обмеження затримки без відкидання пакетів, що задовольняють параметрам трафіку, в умовах відсутності збоїв в роботі мережних компонентів або змін в інформації про маршрути під час життя потоку. Ця служба гарантує мінімальне втручання з боку трафіку, що доставляється без гарантій, ізоляцію зарезервованих потоків і числове вираження максимальної затримки.

Служба гарантованого обслуговування може забезпечити тільки максимальну, але не мінімальну або середню затримку пакетів. Більш того, щоб підрахувати максимальну затримку пакетів, спочатку потрібно визначити фіксовану затримку шляху (затримка поширення і затримка передачі) і додати її до гарантованої максимальної затримки черги.

Максимальна затримка черги – це кумулятивна затримка передачі PATH-повідомлень від джерела до одержувача. PATH-повідомлення містить інформацію про затримку на всьому шляху від джерела до одержувача і у будь-який час надає одержувачеві її точну оцінку. Одержувач використовує інформацію про затримку під час запиту гарантованого обслуговування.

Служба гарантованої бітової швидкості краще всього відповідає тим прикладним програмам масштабу реального часу, які дозволяють відтворювати аудіо і відеофайли. Подібні прикладні програми використовують для своєї нормальної роботи буфер тремтіння з метою компенсації нерівномірності прибуття пакетів. Визначаючи максимальну затримку черги, служба гарантованої бітової швидкості допомагає оцінити необхідний розмір буфера тремтіння. Таким чином, прикладні програми масштабу реального часу можуть розраховувати на гарантовану смугу пропускання і максимальну затримку.

Вказівка на тип обслуговування надається в спеціальному полі об’єкта FlowSpec, причому сам формат об’єкта FlowSpec залежить від типу сервісу: у випадку вибору гарантованого обслуговування в об’єкт FlowSpec входять специфікації RSpec, яких немає у випадку регулювання навантаження.

Нагадаємо, що специфікація RSpec містить інформацію про необхідну смугу і припустиму величину затримки.

Служби регульованого навантаження і гарантованої бітової швидкості використовують корзину маркерів для опису параметрів потоку даних [О: 5, 6]. Корзина маркерів – це механізм регулювання інтенсивності трафіку, що визначає середню швидкість (середній об'єм даних, який можна передати за одиницю часу), розмір сплеску (об'єм даних, який можна відправити протягом заданого проміжку часу без втрат для планування черги) і інтервал вимірювання (квант часу).

При використанні обох служб одержувач запрошує в RESV-повідомленні певну бітову швидкість і розмір сплеску. Планувальник WFQ (зважений алгоритм рівномірного обслуговування черг) і механізм управління чергою WRED (зважений алгоритм довільного передчасного виявлення) гарантують, що трафік досягне одержувача через строго визначений час. Служба регулювання навантаження обіцяє тільки «хороше обслуговування», а служба гарантованої бітової швидкості надає інформацію (у PATH-повідомленнях), з якої можна обчислити значення затримки [О: 5, 6].

7.2.6. Середовище передачі і протокол RSVP

Для ефективної роботи протоколу RSVP найбільше підходить двоточкове середовище передачі (канал між парою вузлів). У разі спільного використання каналів передачі багатьма вузлами в мережах з комутацією кадрів (технології Ethernet, ATM, Frame Relay, X.25) існують певні проблеми використання протоколу RSVP [О:5, 6].

У спільно використовуваному середовищі передачі немає способу дати гарантію, що інші пристрої з того ж сегменту мережі не передають в даний момент трафік, який може привести до перевантаження сегменту. Проте протокол RSVP все ж може нормально працювати на не дуже завантажених сегментах або на сегментах з невеликою кількістю джерел. Щоб вирішити проблему резервування ресурсів в спільно використовуваному середовищі

передачі, був розроблений протокол SBM (Subnet Bandwidth Manager – диспетчер пропускної спроможності підмережі) – протокол управління доступом на основі RSVP для мереж IEEE сімейства 802.3 [Д: 26]. Протокол SBM визначає деякі RSVP-розширення, які надають метод відображення протоколу RSVP на пристрої і мережі канального рівня.

7.2.7. Масштабованість протоколу RSVP

Недоліком протоколу RSVP є те, що об’єм необхідної інформації про стан потоків збільшується із зростанням числа резервувань ресурсів для потоків трафіку.

Оскільки в Internet-магістралі у будь-який час можуть існувати сотні тисяч одноадресних і багатоадресних потоків, використання інформації про стан кожного потоку вважається рішенням, що не масштабується, для магістралей Internet.

При застосуванні у великих intranet-мережах і магістралях постачальників послуг Internet протокол RSVP добре масштабується за умови використання великих багатоадресних груп, або об’єднання потоків на кордонах мережі. Агрегація RSVP-резервувань передбачає злиття декількох наскрізних резервувань, що мають спільні маршрутизатори входу і виходу, в одне велике наскрізне резервування [О: 5, 6].

Інший підхід до вирішення проблеми масштабованості протоколу RSVP в ядрі крупної мережі полягає у використанні протоколу RSVP на кордонах мережі і реалізації diffserv-услуг в її магістралі.

Запитання для самоперевірки та контролю засвоєння знань

1. Що передбачає диференціація послуг при реалізації диференційованого обслуговування?
2. Що розуміють під diffserv-доменом?
3. За допомогою чого вузли diffserv-домену визначають PHB-політику просування конкретного пакету?
4. Яку функцію виконують прикордонні вузли diffserv-домену?
5. Які функції реалізують при формування трафіку?
6. Що дозволяє визначити код диференціальної послуги IP-пакету?
7. Які типи PHB-політик визначають коди DSCP?
8. Що характерно для політики негайної передачі?
9. Що характерно для політики гарантованої передачі?
10. Які параметри характеризують PHB-політику певного класу?
11. Коли застосовують PHB-політику негайної передачі?
12. Які основні фактори впливають на затримку пакетів та тремтіння?
13. Який підхід використовують для уникнення затримку, пов’язаної з виникненням черг?
14. Що передбачає PHB-політика негайної передачі для уникнення виникнення черг?
15. Яке завдання вирішується за допомогою PHB-політику гарантованої передачі?
16. Скільки черг буде використано для пакетів 4-х класів при реалізації PHB-політики гарантованої передачі?
17. Що розуміють під сигналізацією про якість обслуговування при формуванні трафіку на кордоні мережі?
18. Для чого використовують диспетчер політик?
19. Який протокол використовують для розповсюдження політик на вузли diffserv-домену?
20. Що розуміють під точкою визначення політики в протоколі COPS?

21. Що розуміють під точкою вживання політики в протоколі COPS?

22. Яке завдання вирішується за допомогою архітектури інтегрованих послуг?

23. Які функціональні блоки виділяють у архітектурі інтегрованих послуг?

24. За допомогою якого процесу індивідуальні потоки повідомляють про свої вимоги до QoS?

25. Для чого використовується процес управління доступом?

26. Які функціональні блоки необхідно реалізувати в маршрутизаторі для підтримки архітектури інтегрованих послуг?

27. Яка головна перевага архітектури інтегрованих послуг перед архітектурою диференційованих послуг?

28. Наведіть основні недоліки архітектури інтегрованих послуг.

29. Для чого використовується протокол RSVP?

30. Який маршрут для передачі повідомлень про резервування ресурсів вибирає протокол RSVP?

31. Які задачі вирішуються в процесі роботи протоколу RSVP?

32. Яким чином здійснюється резервування ресурсів за допомогою протоколу RSVP?

33. Що робить RSVP-демон маршрутизатора перш ніж зарезервувати ресурси?

34. Яку функцію виконує модуль управління доступом?

35. Яку функцію виконує модуль управління політикою?

36. При виконання якої умови RSVP-демон може встановити параметри класифікатора пакетів і планувальника пакетів для отримання потрібного рівня QoS?

37. Яку функцію виконує класифікатор пакетів?

38. Яку функцію виконує планувальник пакетів?

39. При виконання якої умови модуль управління доступом приймає рішення про резервування ресурсів?

40. Яке повідомлення відправляє джерело даних для резервування ресурсів?

41. Що роблять проміжні маршрутизатори з PATH- повідомленнями?

42. Яке повідомлення відправляє приймач при одержання PATH- повідомлення?

43. Яку інформацію містить повідомлення RESV приймача?

44. Що роблять проміжні маршрутизатори при одержання повідомлень RESV?

45. Що робить останній маршрутизатор при одержанні повідомлень RESV?

46. Які виділяють компоненти RSVP процесу?

47. Які функції виконує RSVP-джерело?

48. Які специфікації потоку передає RSVP-джерело?

49. Які функції виконує RSVP-одержувач?

50. Які специфікації потоку передає RSVP-одержувач?

51. Як реалізовано процес підтримки та відновлення резервування?

52. Як реалізовано скасування резервування?

53. Як реалізовано сигналізацію про помилки?

54. На які типи поділяють повідомлення протоколу RSVP?

55. Які повідомлення є обов'язковими?

56. Які повідомлення є опціональними?

57. Яке повідомлення використовується для ініціалізації встановлення резервування?

58. Яке повідомлення резервує ресурси?

59. Які повідомлення використовуються для інформування про помилки?

60. Які повідомлення використовуються для розриву резервування?

61. Скільки існує класів об'єктів, інформація про які передається в RSVP-повідомленнях?

62. Які поля в RSVP-повідомленнях використовуються для ідентифікації типу об'єкта відповідного класу?

63. Що розуміють під стилем резервування в протоколі RSVP?

64. Які існують способи резервування різними джерелами?

65. Які існують способи вибору джерел резервування?

66. Що характерно для явного вибору джерела?

67. Що характерно для довільного вибору джерела?

68. Скільки існує стилів резервування?

69. Що характерно для індивідуального резервування?

70. Що характерно для загального резервування?

71. Чим відрізняється явне загальне резервування від групового?

72. В об’єкті якого класу вказується стиль резервування?

73. Які типи інтегрованих послуг надає протокол RSVP?

74. Що характерно для служби регульованого навантаження?

75. Що характерно для служби гарантованого обслуговування?

76. В об’єкті якого класу вказується тип обслуговування?

77. Чому в технологіях, які спільно використовують середовище передачі, можливі проблеми при використанні протоколу RSVP?

78. У чому полягає проблема масштабованості протоколу RSVP?

79. Які підходи можуть бути використані для вирішення проблеми масштабованості протоколу RSVP?

8. ЯКІСТЬ ОБСЛУГОВУВАННЯ НА КАНАЛЬНОМУ РІВНІ

8.1. Підтримка QoS в технології ATM

Технологія ATM (асинхронний режим передачі) представляє собою комунікаційну технологію, що поєднує принципи комутації пакетів і каналів для передачі даних різного типу (звичайні комп'ютерні дані, голосові дані, відеодані) [О: 7, 8, Д: 35].

Серед основних властивостей даної технології щодо передачі даних виділяють наступні.

1. Гарантія визначеної пропускної здатності мережі на всьому маршруті передачі даних.

2. Гарантія визначеної величини затримки передачі.

3. Гарантія мінімального рівня помилок передачі.

Технологія ATM надає QoS-гарантії шляхом явного визначення кінцевими ATM – станціями контракту, що описує характеристики їх трафіку. Дескриптор потоку несе такі параметри QoS, як пікова швидкість передачі комірки (Peak Cell Rate – PCR), стабільна швидкість передачі комірки (Sustainable Cell Rate – SCR) і розмір сплеску.

Кінцеві ATM – системи відповідають за те, щоб переданий трафік відповідав контракту QoS. Вони формують трафік шляхом буферизації даних і передачі його в рамках параметрів контракту QoS. ATM – комутатори перевіряють кожну характеристику призначеного для користувача трафіку і порівнюють її з контрактом QoS. Якщо певна частина трафіку не задовольняє контракту QoS, комутатор може встановити CLP – біт для всього трафіку, що не узгоджується із заданими параметрами. Комірка із встановленим бітом CLP має більшу ймовірність бути відкинутою при перевантаженні мережі, ніж комірка із значенням біта CLP, рівним 0.

8.1.1. Класи послуг ATM

Задачі забезпечення QoS вирішуються на адаптаційному рівні ATM (AAL) архітектурної моделі технології ATM. Цей рівень забезпечує необхідні протокольні перетворення для коректної передачі даних різного типу, використовуючи різні протоколи рівня AAL для замовлення і підтримки якості обслуговування.

Послуги шо до передачі даних по ATM мережі, поділяють на 5 класів (табл. 8.1) по наступним трьом характеристикам потоку даних [О:7, 8, Д:35]:

– синхронізація пристроїв між кінцевими точками передачі;

– постійна чи змінна швидкість передачі;

– необхідність встановлення з'єднання.

Таблиця 8.1 – Класи обслуговування в технології ATM

Характеристики	Клас A	Клас B	Клас C	Клас D	Клас E
Синхронізація	+	+	–	–	Описується користувачем або виробником устаткування ATM
Постійна швидкість передачі	+	–	–	–	
Встановлення з'єднання	+	+	+	–	

Клас A призначений для забезпечення сервісу з синхронізацією з обох сторін, постійною швидкістю передачі (CBR) і встановленням з'єднання. Клас A використовується для емуляції цифрових магістралей при передачі голосових потоків через ATM мережу. Забезпечується протоколом AAL1.

Клас B – сервіс із синхронізацією, змінною швидкістю передачі (VBR) і встановленням з'єднання. Застосовується для передачі голосу і відео. Забезпечується протоколом AAL2.

Клас C – сервіс без синхронізації, з змінною швидкістю передачі (VBR) і з встановленням з'єднання. Використовується для передачі звичайних цифрових даних. Забезпечується протоколами AAL3, AAL5.

Клас D – сервіс без синхронізації, зі змінною швидкістю передачі (VBR) і без встановлення з'єднання. Використовується для передачі цифрових даних. Забезпечується протоколами AAL4, AAL5.

Існуючі послуги АТМ ґрунтуються на синхронізації між джерелом і пунктом призначення, інтенсивності трафіка і режимі з'єднання. Форум АТМ визначає п'ять категорій послуг [О:7, 8, Д:35].

• Обслуговування з постійною бітовою швидкістю (CBR). Призначене для даних масштабу реального часу, які не допускають затримок, тремтіння, і вимагають широку смугу пропускання. CBR – це синонім виразу «емуляція каналу» (Circuit Emulation). Для такого типу трафіку потрібна зарезервована смуга пропускання, яка повинна надаватися на вимогу. Як приклад можна привести передачу стандартного голосового трафіку на швидкості 64 Кбіт/с і трафіку відеозображень (стандарт H.320).

Змінна швидкість передачі в режимі реального часу (Real-Time Variable Bit Rate – RT-VBR). Як випливає з назви, вказана послуга, як і CBR, призначається для передачі даних масштабу реального часу, проте в цьому випадку джерела можуть відправляти дані нерівномірно. Об'єднаний в пакети трафік відео зображень і даних з вимогами режиму реального часу (наприклад, інтерактивний мультимедійний трафік) відноситься саме до цієї категорії. Послуга RT-VBR визначається параметрами PCR, SCR і максимальним розміром сплеску (maximum burst size – MBS).

Змінна швидкість передачі не в режимі реального часу (Non-Real Time Variable Bit Rate – nRT-VBR). Дана послуга призначена для прикладних програм, які передають трафік сплесками і допускають наявність затримок і тремтіння. У цю категорію потрапляють інтерактивні транзакції даних. Послуга nRT-VBR визначається параметрами PCR, SCR і MBS.

Доступна швидкість передачі (Available Bit Rate – ABR). ABR-джерела регулюють швидкість передачі відповідно до інформації, що надається мережею. Послуга ABR не резервує смугу пропускання як таку, а адаптує трафік до мережі, грунтуючись на механізмі зворотного зв'язку, який реалізується за допомогою комірок управління ресурсами (Resource Management – RM). ABR використовує метод оцінювання, що полягає в тому, що швидкість передачі на кожному віртуальному каналі підбирається

відповідно до явної вказівки швидкості, яка передається RM-комірками. RM-комірки передаються перед комірками з даними. Швидкість джерела контролюється RM-комірками, що повернулися. RM-комірки є новим джерелом витрат, пов'язаних з передачею службової інформації. Послуга ABR добре працює з протоколом управління передачею (Transmission Control Protocol – TCP), в якому реалізований адаптивний алгоритм управління потоком.

Послуга ABR визначається параметром PCR і мінімальною швидкістю передачі комірок (Minimum Cell Rate – MCR), що гарантується, на кожному віртуальному каналі.

Невизначена бітова швидкість (Unspecified Bit Rate – UBR). UBR є ATM – еквівалентом послуги передачі даних методом негарантованої доставки протоколу IP. Відповідно до цієї послуги резервування смуги пропускання не проводиться, а також немає ніяких обмежень, затримки, що стосуються, і тремтіння. На рівні ATM також не здійснюється моніторинг за перевантаженням мережі. Прикладом трафіка такого типу може служити передача файлів великого розміру. Протокол TCP не дуже добре працює в зв'язці з послугою UBR, оскільки передача даних по UBR-каналу може характеризуватися втратою комірок. Проте становище можна виправити за допомогою стратегій відкидання комірок.

8.1.2. Стратегії відкидання комірок

У заголовку ATM – комірки є CLP-біт для вказівки пріоритету втрати комірки. Насамперед відкидаються комірки з CLP = 1, а в другу чергу – комірки з CLP = 0. Механізм обмеження ATM – трафіку може встановити CLP-біт рівним 1 у комірці, яка порушила специфікацію контракту трафіку віртуального каналу. Коли в якійсь частині ATM-мережі відбувається перевантаження, можуть бути відкинуті будь-які комірки, але комірка зі встановленим CLP-бітом відкидається насамперед. Часткове відкидання пакетів (Partial Packet Discard – PPD) і раннє відкидання пакетів (Early Packet Discard – EPD) – це стратегії

відкидання АТМ – комірок, які здатні ефективно підвищити продуктивність АТМ-мережі [О: 7, 8, Д: 35].

В архітектурі АТМ підрівень SAR (підрівень збирання/розбирання) адаптаційного рівня (AAL) відповідає за розбиття великих пакетів на комірки. Якщо відкинута АТМ – комірка є частиною великого пакету, немає потреби посилати комірки, що залишилися, які належать сегментованому пакету, оскільки в цьому випадку виникне непотрібний трафік на і без того перевантаженому каналі. В цьому випадку повторна збірка комірок в первинний пакет в кінцевому пункті неможлива, і необхідно повторно переслати весь пакет цілком. Отже, коли відкидається комірка великого пакету, функція PPD відкидає і решту комірок, що належать цьому пакету. Перевага такого підходу обмежена, оскільки комутатор починає відкидати комірки тільки в разі переповнювання буфера.

Реалізація функції PPD безпосередньо залежить від протоколу AAL5. Ця функція може бути використана на кожному віртуальному каналі. Комутатор відкидає комірки на певному віртуальному каналі до тих пір, поки встановлений в заголовку АТМ – комірки параметр не вкаже на те, що ALL-пакет закінчився. Саме комірка, що є ознакою кінця пакету, не відкидається. Оскільки AAL5 не підтримує одночасне мультиплексування пакетів на одному віртуальному каналі, цю комірку можна використовувати для визначення границь пакету.

Оскільки відкидання комірок великого пакету (і, як наслідок, вживання функції PPD) може статися після того, як попередні комірки великого пакету були поставлені в чергу на відправку, функція PPD проводить тільки часткове відкидання пакету. Тому, по перевантаженому каналу все ще передаватиметься значна частина комірок, що належать пошкодженим пакетам. Слід зазначити, що якщо відкинута комірка була першою в пакеті, то PPD відкидає весь пакет. З іншого боку, функція EPD застосовується ще перед тим, як яка-небудь комірка надходить до черги на відправку. Під час надходження нових пакетів алгоритм EPD перевіряє коефіцієнт завантаження вихідного буфера. Якщо цей коефіцієнт нижче встановленого порогового значення, АТМ-комутатор вважає,

що розмір буфера не буде перевищений, і всі комірки пакету можуть бути поміщені в чергу. Інакше комутатор припускає, що буфер близький до переповнювання і весь пакет не може бути поміщений в чергу, внаслідок чого пакет відкидається. Таким чином, функція EPD або ставить в чергу всі комірки пакету, або відкидає весь пакет. У зв'язку з тим, що функція EPD проводить повне відкидання пакету перед тим, як будь-яка з комірок пакету буде поміщена в чергу, цей процес називається раннім відкиданням пакету (early packet discard).

Як і на віртуальному каналі ATM, аналогічні послуги з вирівнювання трафіку можна застосувати з метою встановлення обмежень на контракт для цілого віртуального шляху (VP). В межах VP по всіх віртуальних каналах може передаватися трафік UBR-типу без яких-небудь суворих обмежень по параметрах. Послуги ATM, вживані на рівні віртуального шляху, можуть підтримувати всі класи послуг ATM, вживані на рівні віртуального каналу.

8.1.3. Міжмережний обмін ATM і IP QoS

У IP-мережі, деяка частина якої використовує ATM як основну технологію передачі даних канального рівня, IP-трафік передається по ATM-магістралі за допомогою віртуального каналу, який характеризується певним рівнем послуг ATM QoS, відповідним IP-трафіку, що передається. Проте, через перевантаження на вході в ATM-мережу розмір черг може зрости, що призведе до відкидання пакетів. Без IP QoS такі відкидання відбуваються випадковим чином без визначення пріоритету IP-пакету, тим самим перешкоджаючи забезпеченню наскрізних функцій IP QoS в ATM-мережі. Тому реалізація функцій IP QoS в ATM-мережі є необхідною. З цією метою на вхідних чергах ATM-мережі можуть бути застосовані такі механізми QoS, як зважений алгоритм довільного раннього виявлення (Weighted Random Early Detection – WRED) і зважений алгоритм рівномірного обслуговування черг (Weighted Fair Queuing – WFQ). Після цього відкидання пакетів і планування вихідної черги ґрунтуватиметься на запитаному рівні якості обслуговування пакету. З точки

зору IP QoS IP-трафік передається по ATM-мережі без втрат (втрати в ATM-мережі не відстежуються на рівні IP QoS) з використанням послуги ATM, яка задовольняє всім запитам IP-трафіку до якості обслуговування. На рівні IP QoS кожному віртуальному каналу ATM-мережі відповідає окрема черга, так що алгоритми WRED і WFQ можна застосувати до будь-якого віртуального каналу (VC) [О: 5, 6, Д: 35].

Можливі наступні сценарії забезпечення функцій IP QoS при передачі трафіку по ATM-мережі [О:5, 6, Д:35].

1. Один канал PVC (постійний віртуальний канал) передає весь IP-трафік в кінцеву точку призначення. IP-трафік, що перевищує параметри ATM PVC, ставиться в чергу на вході в ATM-мережу. По мірі зростання перевантаження черга обслуговується такими механізмами IP QoS, як WRED і WFQ. Алгоритм WRED гарантує, що втрати пріоритетнішого трафіку будуть значно менше, ніж втрати менш пріоритетного трафіку. Алгоритм WFQ гарантує, що пріоритетнішому трафіку буде надана більш велика смуга пропускання в порівнянні з менш пріоритетним трафіком, оскільки пріоритетніший трафік обслуговується частіше. Слід також відзначити, що коли трафік каналу PVC обслуговується відповідно до зваженого механізму рівномірного обслуговування черг на основі класу (Class-Based Weighted Fair Queuing – CBWFQ), можна призначати смуги пропускання залежно від класу трафіку.

2. Використання віртуального шляху (VP), який складається з декількох PVC для переносу IP-трафіку в кінцеву точку призначення. Коли трафік з різними вимогами QoS (трафік масштабу реального часу; трафік, що передається не в режимі реального часу; трафік негарантованої доставки) доставляється в одне і те ж місце, краще всього створити віртуальний шлях з декількох PVC-каналів в ATM-мережі з метою передачі кожного класу IP-трафіку по окремому каналу. Кожен PVC відповідає класу послуг ATM залежно від IP-трафіку, що передається по ньому. На рис. 8.1 зображена схема міжмережевої взаємодії IP – ATM з використанням віртуального шляху.

Рисунок 8.1 – Передача з використанням віртуального шляху з декількох PVC

Кожен віртуальний канал, який входить до віртуального шляху, при такому сценарії призначений для передачі трафіку з певним значенням IP-пріоритету. Для кожного віртуального каналу в рамках віртуального шляху існує окрема черга. Для кожної з них можна застосувати алгоритм IP QoS WRED і WFQ.

Віртуальний канал можна конфігурувати для передачі трафіку з одним або декількома значеннями IP-пріоритету. За цілісністю віртуального каналу можна стежити за допомогою набору протоколів ATM OAM (Operation and Maintenance) або ILMI (Interim Local Management Interface – проміжний інтерфейс локального управління). Якщо віртуальний канал з високим пріоритетом перестає працювати, можна або «перекинути» його трафік в канал з меншим пріоритетом з цього ж віртуального шляху, або оголосити про обрив всього віртуального шляху.

Клас послуг ATM для IP-трафіку є вищим за клас UBR (передача методом негарантованої доставки), в результаті чого пакети (комірки) не відкидатимуться усередині ATM-мережі в результаті використання ATM QoS. На рис. 8.2 зображені два сценарії міжмережевої взаємодії IP – ATM QoS.

Рисунок 8.2 – Міжмережна взаємодія IP-ATM QoS

Прийняття рішення про використання одного каналу PVC або віртуального шляху для забезпечення IP QoS при міжмережному обміні залежить від ціни і важливості трафіку в мережі. Не дивлячись на те, що віртуальний шлях може бути дорожче одного каналу PVC, він здатен ізолювати критичні класи трафіку (наприклад, голосовий трафік). З іншого боку, для віртуального шляху необхідно заздалегідь спланувати передачу трафіку для того, щоб всі PVC-канали використовувалися оптимально. Інакше може виникнути ситуація, коли PVC-канал, по якому передається високопріоритетний трафік, виявиться перевантаженим, а PVC-канал, по якому передається менш пріоритетний трафік, залишиться відносно вільним. В разі перевантаження PVC-каналу не можна автоматично перекинути його високопріоритетний трафік на інший PVC-канал цього віртуального шляха. При використанні одного PVC можна застосувати механізм CBWFQ з пріоритетною чергою, забезпечуючи, таким чином, першість голосового трафіку серед інших типів даних, що передаються по цьому PVC.

8.2. Підтримка QoS в технології Frame Relay

Технологія Frame Relay являє собою спрощений різновид пакетної комутації, який можна використовувати на якісних каналах передачі даних з використанням інтелектуальних транспортних протоколів (або протоколів/додатків) прикладного рівня для виправлення помилок передачі. При виявленні помилки передачі (на канальному рівні) протокол Frame Relay відкидає помилкові дані і чекає, поки ці помилки не будуть виправлені протоколом більш високого рівня. Тим не менш, технологія Frame Relay забезпечує підтримку таких показників якості обслуговування як середня швидкість передачі і об'єм пульсації трафіку [О: 7, 8, Д: 35].

У протоколі Frame Relay механізм запобігання перевантаженню забезпечують три біта заголовка Frame Relay – кадру. Це біти прямого явного повідомлення про перевантаження (Forward Explicit Congestion Notification – FECN), зворотнього явного повідомлення про перевантаження (Backward Explicit Congestion Notification – BECN) і DE-біт (Discard Eglible) – рис. 8.3.

Значення FECN-біта, яке дорівнює 1, встановлюється, для того, щоб повідомити кінцеве термінальне устаткування (data terminal equipment – DTE), наприклад маршрутизатор, що на шляху передачі кадру від джерела до кінцевої точки сталося перевантаження мережі [О:7, 8, Д:35].

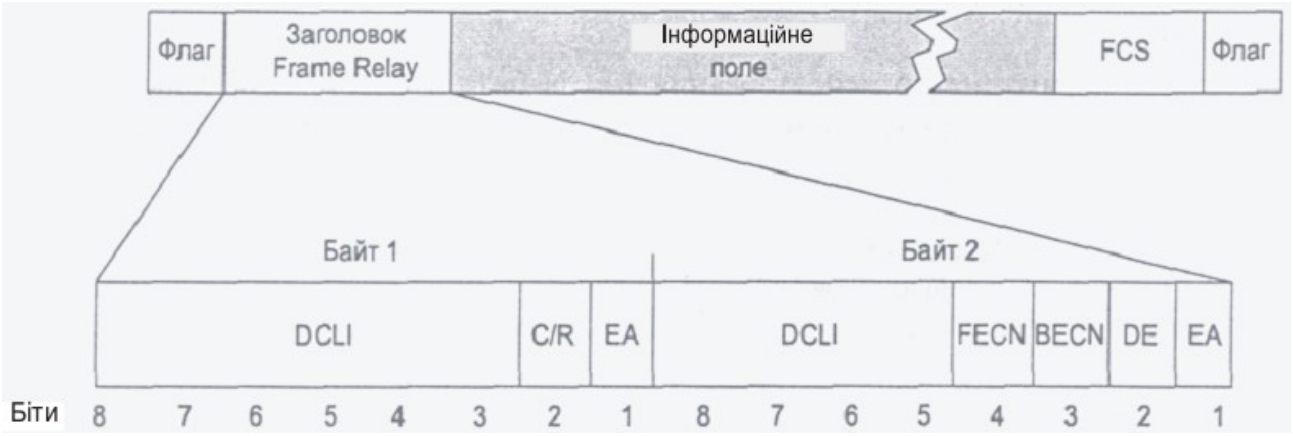

Рисунок 8.3 – Заголовок Frame Relay – кадру

Значення BECN-біта встановлюється комутатором Frame Relay в 1, для того, щоб повідомити маршрутизатор пункту призначення, що в мережі

сталося перевантаження по напряму, зворотному до напряму передачі кадру від джерела до пункту призначення.

DE – «прийнятність відкидання» – встановлюється DTE пристроями або комутаторами мережі, щоб сповістити мережу про те, що даний кадр має більш низький пріоритет у порівнянні з іншими, і може бути відкинутим мережею у випадку перевантажень на каналах передачі. Таким чином, забезпечується базовий механізм призначення пріоритетів в Frame Relay-мережах. Кадри із встановленим DE-бітом відкидаються в першу чергу в порівнянні з кадрами, в яких цей прапор не встановлений.

На основі біта FECN приймаюча сторона за допомогою протоколів більш високих рівнів (наприклад, TCP) сповістить сторону, яка передає, про те, що вона повинна знизити інтенсивність відправки пакетів в мережу. Біт BECN сповіщає сторону, яка передає, про перевантаження в мережі та про необхідність зменшити темп передачі. Біт BECN обробляється на рівні пристроїв доступу до мережі Frame Relay (маршрутизаторів, мультиплексорів та ін.). Цей біт вказує протоколам більш високих рівнів на необхідність зниження темпу передачі пакетів.

Механізм вирівнювання трафіку в мережі Frame Relay (FRTS)

Механізм FRTS (Frame Relay Traffic Shaping) вирівнює трафік, що передається по віртуальному каналу, відповідно до визначеної швидкості передачі інформації. FRTS намагається згладити пульсуючий трафік, буферизуючі пакети, що перевищили середню інтенсивність [О: 7, 8, Д: 35]. Як тільки механізм обслуговування черг визнає об’єм ресурсів, що звільнився, достатнім, пакети з буфера вилучаються з черги для передачі в канал. Алгоритм обслуговування черг може бути конфігурований на кожному віртуальному каналі тільки для трафіку, що надсилається з інтерфейсу в канал.

Принцип роботи механізму FRTS наступний. Для кожного віртуального каналу визначається кілька параметрів:

– CIR (Committed Information Rate) – погоджена інформаційна швидкість, з якою мережа буде передавати дані користувача;

– Bc (Committed Burst Size) – погоджений обсяг пульсацій – максимальна кількість байтів, які мережа буде передавати від користувача за інтервал часу Tc;
– Be (Excess Burst Size) – додатковий обсяг пульсацій, тобто максимальна кількість байтів, що мережа буде намагатися передати понад встановлене значення Bc за інтервал часу Tc.

Дані параметри вказуються в угоді на обслуговування, яка укладається між користувачем та оператором мережі. Якщо ці величини визначені, то інтервал часу Tc визначається виразом:

$$\boldsymbol{Tc = Bc / CIR}.$$

Гарантій по затримках передачі кадрів технологія Frame Relay не дає.

Основним параметром є CIR. Для PVC ця величина встановлюється на етапі конфігурування. Для SVC (комутований віртуальний канал) угода про якість обслуговування укладається автоматично за допомогою протоколів Q.931/933 – необхідні параметри CIR, Bc та Be передаються в пакеті запиту на встановлення з'єднання.

Інтервал Tc є контрольним інтервалом, на якому перевіряються умови угоди. У загальному випадку користувач не повинен передати за цей інтервал у мережу дані із середньою швидкістю, що перевищує CIR. Якщо він порушує угоду, то мережа не тільки не гарантує доставку кадру, але позначає цей кадр ознакою DE = 1 – кадр, що підлягає видаленню. Але такі кадри видаляються з мережі тільки у випадку перевантаження комутаторів (якщо перевантажень нема, то кадр буде доставлено). Така поведінка мережі відповідає випадку, коли загальна кількість даних, переданих користувачем у мережу за період Tc, не перевищує обсягу Bc + Be. Якщо цей поріг перевищений, то кадр не позначається ознакою DE, а негайно видаляється з мережі.

Для кожної угоди про параметри якості обслуговування всі комутатори мережі виконують алгоритм «дірявого відра» (Leaky Bucket) [О: 7, 8, Д: 35]. Алгоритм використовує лічильник C байт, які надійшли від користувача. Кожні Tc секунд лічильник зменшується на величину Bc (або скидається в нуль, якщо

значення лічильника менше, ніж Bc). Всі кадри, дані яких не збільшили значення лічильника вище порогу Bc, пропускаються в мережу зі значенням DE = 0. Кадри, дані яких змінили значення лічильника в інтервалі Bc; Bc + Be, передаються в мережу з ознакою DE = 1. Кадри, які призвели до значення лічильника > Bc + Be, відкидаються комутатором.

Можна включати в угоду не всі параметри якості обслуговування на даному віртуальному каналі. Наприклад, тільки CIR та Bc. Такий варіант передбачає більш якісне обслуговування – кадри ніколи не відкидаються комутаторами відразу, хоча позначаються ознакою DE = 1. Існує ще один вид замовлення – обумовлюється поріг Be, а швидкість CIR покладається рівною нулю. Всі кадри відразу ж позначаються ознакою DE = 1, але відправляються в мережу, а при перевищенні порогу Be, вони відкидаються. Контрольний інтервал Tc у цьому випадку обчислюється як Be / R, де R – пропускна здатність каналу.

Зазвичай доступ до мережі здійснюється по каналах із пропускною здатністю, більшою ніж CIR. Але при цьому користувач платить за замовлені величини CIR, Bc та Be. Швидкість каналу буде впливати на верхню межу можливої пульсації Bc + Be. Параметри якості обслуговування можуть бути різними для різних напрямків передачі у віртуальному каналі.

Механізм замовлення середньої пропускної здатності та максимальної величини пульсації є основним механізмом керування потоками кадрів у мережах Frame Relay. Угоди повинні складатися таким чином, щоб сума середніх швидкостей віртуальних каналів не перевищувала пропускної здатності фізичного каналу і порту комутатора. При замовленні PVC за це відповідає адміністратор, а при встановленні SVC – програмне забезпечення комутаторів.

Адаптивний механізм FRTS

Механізм FRTS, що працює в адаптивному режимі, дозволяє зменшити інтенсивність трафіку, який надходить від DTE-пристрою в віртуальний канал, ґрунтуючись на отриманих мережевих індикаторах перевантаження BECN [О: 7, 8, Д:35]. Він вирівнює інтенсивність трафіку, що передається через PVC-канал, залежно від доступної смуги пропускання Frame Relay-мережі.

Таким чином, інтенсивність трафіку може складати X, проте коли мережа почне посилати BECN-індикатори, вона буде зменшена до Y.

Протягом кожного часового інтервалу Tc процес перевіряє, чи отримано від мережі Frame Relay повідомлення BECN. Якщо повідомлення BECN було отримане протягом останнього Tc-інтервалу, швидкість передачі зменшується на 25 % в порівнянні з поточною швидкістю. Скидання швидкості може продовжуватися до тих пір, поки не буде досягнутий нижній поріг CIR (MINCIR). Незалежно від завантаження Frame Relay-мережі, маршрутизатор не скидатиме швидкість нижче за значення MINCIR. Значення MINCIR за замовчуванням дорівнює половині значення CIR.

Після адаптації швидкості передачі трафіку до перевантаження в Frame Relay-мережі механізм вирівнювання трафіку чекає 16 Tc – інтервалів без повідомлень BECN і починає збільшувати швидкість передачі трафіку до визначеної швидкості CIR.

Адаптивний механізм ефективно працює при обміні даними в режимі зі встановленням з’єднання та підтвердженням доставки (наприклад, при використанні TCP протоколу). При використанні дейтаграмного режиму передачі (наприклад, протоколи UDP або IP) підтвердження не надсилаються, що не дає можливості повідомити вузол-відправник про перевантаження на каналі шляхом встановлення біту BECN у відповідні кадри. Якщо застосувати механізм вирівнювання UDP-трафіка, то єдине повідомлення про перевантаження мережі полягатиме в установці FECN-біта, що отримується разом з кадрами, що приходять в кінцевий маршрутизатор. Для врахування можливих перевантажень на каналі при дейтаграмному режимі передачі в адаптивному механізмі передбачена команда на відправку тестового кадру Q.922 із встановленим BECN-бітом у відповідь на отримання кадру із встановленим FECN-бітом. В BECN-кадрі, який відправляється у відповідь на отримане повідомлення FECN, має бути скинутий DE-біт. Не дивлячись на те, що при отриманні маршрутизатором-джерелом тестового кадру Q.922 останній відразу ж відкидається, BECN-біт указує на необхідність зменшення інтенсивності трафіку.

8.2.1. Фрагментація в мережах Frame Relay

Великі пакети, поставлені у вихідну чергу віртуального каналу маршрутизатора перед маленькими, чутливими до затримок пакетами, сприяють збільшенню затримки й тремтіння маленьких пакетів. Це відбувається через те, що затримка при передачі великих пакетів, звичайно, більше затримки при передачі маленьких пакетів. Для вирішення цієї проблеми форумом Frame Relay (Frame Relay Forum – FRF) було затверджено стандарт фрагментації Frame Relay – FRF.12 BECN [Д:35, 36, 37] .

Стандарт FRF.12 гарантує, що голосові пакети й інші подібні їм невеликі пакети даних не будуть затримуватися через передачу великих пакетів. У цьому стандарті також передбачене гарантоване рівномірне відправлення маленьких пакетів, що дозволяє зменшити тремтіння трафіка. Така можливість дозволяє передавати по одному PVC-каналу голосовий трафік і трафік інших прикладних програм, таких, як ERP-програм (Enterprise Resource Planning – планування ресурсів підприємств), програм, необхідних для рішення критично важливих завдань, програм, які не чутливі до затримок, тощо.

Важливим практичним застосуванням стандарту FRF.12 є передача голосової інформації через мережу Frame Relay з якістю обслуговування міжміського телефонного зв’язку. Фрагментація, особливо якщо вона використовується разом з механізмами QoS, такими, як WFQ та WRED, гарантує послідовну безперервну передачу голосової інформації.

До реалізації стандарту FRF.12 єдиним способом розбити пакети в IP-середовищі було встановлення маленького значення MTU (maximum transmission unit – максимального розміру одиниці передачі інформації). Однак цей спосіб є досить неефективним через підвищення завантаженості процесора й збільшення числа пакетів. У випадку передачі голосу по мережі IP (Voice over IP – VoIP) фрагментація пакетів стає абсолютною необхідністю в мережах з невеликою пропускною здатністю [Д: 35, 36, 37].

Активізація механізму FRTS є необхідною умовою при переході до фрагментації FRF.12. При вирівнюванні голосового трафіка FRTS використовує

зважений алгоритм рівномірного обслуговування черг (Weighted Fair Queuing – WFQ) на основі потоку із пріоритетною чергою (priority queue – PQ). Кожен PVC-канал Frame Relay має свою власну структуру PQ-WFQ, що використовується для планування обслуговування пакетів на основі значення IP-пріоритету відповідно до алгоритму WFQ на основі потоку. На відповідному інтерфейсі механізм FRTS використовує дві FIFO-черги: одну для пріоритетних пакетів (голосовий трафік, LMI-трафік (Local Management Interface – інтерфейс локального керування) тощо), іншу – для іншого трафіка. Механізм FRTS забороняє на інтерфейсі будь-який інший метод обслуговування черг, крім подвійного FIFO [Д: 35, 36, 37].

На рис. 8.4 зображені три різних потоки – два потоки даних й один потік голосового трафіка, які надходять на інтерфейс для їхньої наступної передачі.

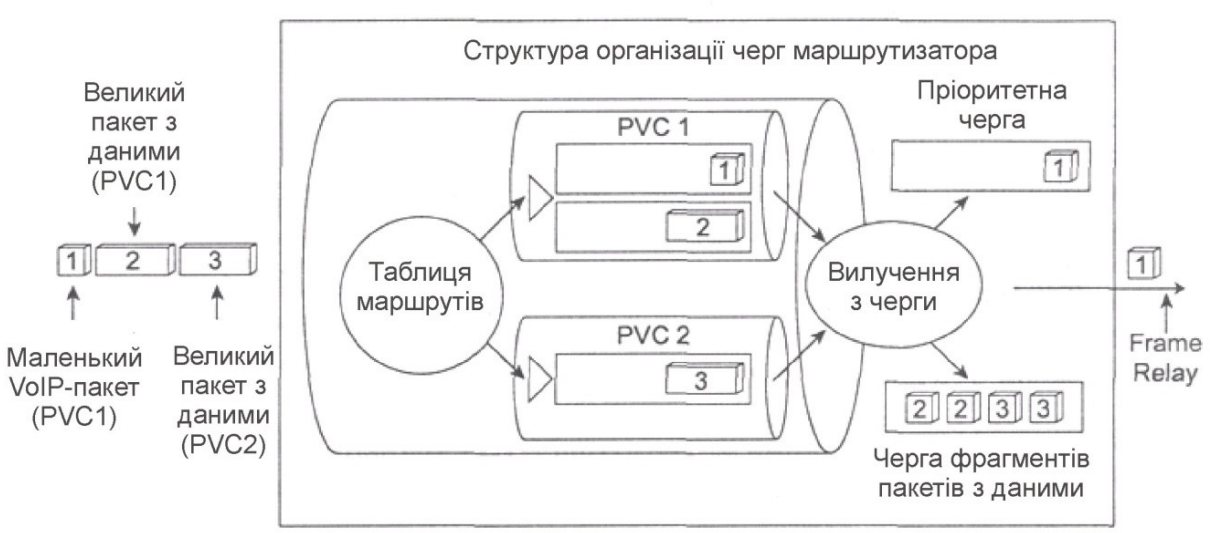

Рисунок 8.4 – Робота механізму FRTS з декількома каналами PVC

Потоки маршрутизуються до відповідних структур PVC-каналу на основі інформації в заголовку пакета. У нашому випадку потоки 1 й 2 направляються до каналу PVC 1, а потік 3 – до каналу PVC 2. Кожному каналу PVC відповідає своя черга вирівнювання трафіка. Так, на каналі PVC 1 сконфігурований механізм PQ – WFQ, що передбачає першочергове обслуговування пакетів голосового потоку за допомогою пріоритетної FIFO-черги інтерфейсу. Пакети з даними, що обслуговуються за допомогою черги з вирівнюванням трафіка, розбиваються на

фрагменти залежно від встановленого граничного значення й поміщаються у звичайну FIFO-чергу інтерфейсу. Граничне значення розміру фрагмента повинне бути встановлене таким чином, щоб жоден голосовий пакет не був фрагментований. Незважаючи на те, що для кожного каналу PVC існує окрема черга вирівнювання трафіка, всі канали PVC використовують подвійну FIFO-чергу інтерфейсу. Пакети вилучаються із пріоритетної FIFO-черги у відповідності зі строгим пріоритетом у порівнянні з пакетами звичайної FIFO-черги інтерфейсу.

8.2.2. Міжмережний обмін Frame Relay й IP QoS

Міжмережна взаємодія IP – Frame Relay QoS, як і міжмережна взаємодія IP – ATM QoS, необхідна для забезпечення наскрізної якості обслуговування в мережах IP. На вході в мережу Frame Relay черги IP-трафіка створюються для кожного віртуального каналу (VC) Frame Relay. Для кожної VC-черги можна застосувати алгоритми WFQ й WRED. На додаток до WFQ, як механізм обслуговування VC-черги можна вибрати механізм PQ-WFQ, механізм із пріоритетною схемою обслуговування й механізм із замовленою схемою обслуговування [Д: 35, 36, 37].

Як згадувалося раніше, для забезпечення базової диференціації трафіка в технології Frame Relay використовується біт DE. Імовірність відкидання кадру із установленим прапором DE більше, ніж імовірність відкидання кадру зі значенням DE-біта, рівним 0. У зв’язку з тим, що в протоколі IP для визначення пріоритету використовується три біти, прямо відобразити DE-біт на рівні IP-пріоритету не можна. Проте можна скористатися досить простим відображенням, а саме – поставити у відповідність IP-трафіку, що передається методом негарантованої доставки (IP-пріоритет «О»), установлений DE-біт, а іншому, більше важливому трафіка – скинутий DE-біт. Таким чином, трафік, що передається методом негарантованої доставки, буде відкидатися раніше, ніж більше важливий трафік.

8.3. Протоколи локальних мереж IEEE 802 й IP QoS

Можливість встановлення пріоритетів для кадрів даних різних мережних додатків з подальшою їх пріоритетною передачею через Ethernet – мережі визначена як частина стандарту 802.1p, який використовується спільно зі стандартом тегування кадрів для визначення їх належності різним VLAN – 802.1Q [О: 7, 9]. Згідно стандарту 802.1Q для визначення приналежності кадру конкретній VLAN використається 4-байтова мітка, яка додатково містить і біти пріоритету 802.1p (рис. 8.5.).

Рисунок 8.5 – Кадр 802.1Q з бітами 802.1p

У стандарті 802.1Q визначається новий тип маркування кадру шляхом додавання 4-байтової мітки, що складається з наступних частин [О: 7, 9]:

– два байти ідентифікатора маркірованого протоколу (Tagged Protocol Identifier – TPID) – значення в цьому полі вказує, який протокол використовується для тегування (значення 0x8100 використовується при маркуванні пакета за протоколом 802.1Q);
– два байти маркірованої контрольної інформації (Tagged Control Information – TCI):
  – три біти 802.1 p – використовуються згідно стандарту IEEE 802.1p для завдання пріоритету трафіку що передається;

– один біт ідентифікатора канонічного формату (Canonical Format Identifier – CFI) – вказує на формат MAC-адреси (1 – канонічний – кадр Ethernet, 0 – не канонічний – кадр Token Ring, FDDI);
– 12-бітовий ідентифікатор VLAN (ID) – вказує, до якої VLAN належить кадр.

У стандарті 802.1p передбачений метод збереження інформації про пріоритет при передачі кадру по локальній мережі [О: 7, 9]. Усього існує 8 пріоритетів на основі значення трьох бітів 802.1p (від 0 до 7, де 7 – найвищий пріоритет), які використовуються в стандарті 802.1p для призначення рівнів пріоритету кадрам різних мережних додатків. З метою підтримки стандарту 802.1p канальний рівень повинен підтримувати множинні черги – по однієї на кожен пріоритет або клас трафіка. Комутатор зберігає значення пріоритету при комутації кадру.

Зовсім нескладно відобразити 3-бітове поле IP-пріоритету на три біти пріоритету стандарту IEEE 802.1p і навпаки, забезпечуючи таким чином взаємодію між IP QoS й IEEE 802.1p.

У сімействі комутаторів Cisco Catalyst біти класу обслуговування (Class of Service – COS) 802.1p використаються для диференціації трафіка на основі таких механізмів QoS, як WRR (Weighted Round Robin – зважений алгоритм кругового обслуговування) і WRED [О: 5]. Незважаючи на те що в стандарті 802.1p використовується мітка, визначена IEEE 802.1Q – стандартом для мереж VLAN, 802.1p можна використовувати й при відсутності віртуальної локальної мережі. Ідентифікатор VID, рівний 0, зарезервований і використовується для вказівки на відсутність VLAN.

Через додаткову 4-байтову мітку, додану специфікаціями 802.1Q й 802.1p, Ethernet-кадр може перевищити свій максимальний розмір в 1518 байт. Саме тому організація IEEE розробляє специфікацію 802.3ac (модифікацію стандарту 802.3) з метою збільшити максимальний розмір кадру з 1518 до 1522 байт.

8.4. Диспетчер SBM

SBM (Subnet Bandwidth Manager – диспетчер смуги пропускання підмережі) – сигнальний протокол, що підтримує керування доступом на основі протоколу RSVP у сімействі мереж Ethernet 802.3 [Д: 26]. Як було показано вище, RSVP – це механізм наскрізної сигналізації, що використовується для запитів резервування ресурсів мережі. У мережі Ethernet гарантоване резервування смуги пропускання неможливо для жодного вузла в сегменті, тому що Ethernet-сегмент є мережею з поділюваним середовищем передачі. Вузол Ethernet-сегмента не має відомостей про резервування ресурсів, зроблених іншими вузлами, і може відправляти трафік у сегмент зі швидкістю, що не відповідає існуючому розподілу ресурсів. SBM – це засіб для підтримки резервування ресурсів на основі протоколу RSVP у мережах сімейства 802.3.

Ethernet-сегмент із одним або декількома пристроями, що підтримують протокол SBM, називається керованим Ethernet-сегментом (managed Ethernet segment) [Д: 26]. Керований сегмент може бути як загальним сегментом з однієї або декількома SBM-станціями, так і комутованим сегментом з максимум двома SBM-станціями. У керованому сегменті один із пристроїв з працюючим протоколом SBM відіграє роль призначеного SBM (designated SBM – DSBM) [Д: 26]. Станція DSBM може обиратися автоматично, або бути призначеною адміністративно. Всі інші SBM-станції керованого сегмента (крім станції DSBM) відіграють роль DSBM-клієнтів. DSBM-станція відповідає за керування доступом для всіх запитів на резервування ресурсів, що відправляються DSBM-клієнтами керованого сегмента. Маршрутизатори Cisco й операційна система Windows 2000 є прикладами станцій, що підтримують DSBM-функціональність.

Для забезпечення роботи протоколу SBM використовується дві широкомовні адреси [Д: 26].

– AllSBMAddress. Використовується для вибору DSBM і передачі всіх DSBM-повідомлень. Всі SBM-сумісні станції прослуховують цю адресу, а DSBM-станція відправляє на неї свої повідомлення.

– DSBMLogicalAddress. Використовується для передачі DSBM-клієнтами повідомлень RSVP PATH DSBM-станції. Цю адресу прослуховують тільки станції-кандидати DSBM.

Після активізації протоколу RSVP на SBM-сумісній станції поділюваного сегмента він реєструє інтерфейс цього сегмента для прослуховування широкомовної адреси AllSBMAddress. Якщо станція одержала на цю адресу повідомлення I_AM_DSBM, інтерфейс вважається належний до керованому сегменту. Протокол RSVP на керованому сегменті повинен працювати відповідно до SBM-протоколу. DSBM-клієнт керованого сегмента прослуховує повідомлення RSVP PATH, передані йому через DSBM-станцію за адресою AllSBMAddress [Д: 26].

Нова SBM-станція керованого сегмента протягом якогось періоду часу працює в режимі прослуховування, для того щоб визначити, чи обраний DSBM даного сегмента. Якщо SBM-станція одержує повідомлення I_AM_DSBM на адресу AllSBMAddress, то вона не бере участь у виборах DSBM, поки існуюча DSBM-станція не вийде з ладу й не стане необхідно провести нові вибори DSBM.

При роботі протоколу SBM DSBM-станція перехоплює всі вхідні і вихідні повідомлення RSVP PATH й RSVP-TEAR, додаючи тим самим один додатковий перехід (hop). Всі вихідні від SBM-клієнта керованого сегмента повідомлення RSVP PATH відправляються DSBM-пристрою цього сегмента (за адресою DSBMLogicalAddress), а не за адресою призначення RSVP-сеансу. Після обробки DSBM-станція направляє повідомлення PATH за адресою призначення RSVP-сеансу. Під час обробки DSBM-станція створює й обновляє блок стану шляхів (Path State Block – PSB) сеансу й зберігає адресу L2/L3 попереднього переходу повідомлення PATH [Д: 26].

Отримане DSBM-станцією повідомлення RSVP PATH вимагає аналізу додаткових SBM-об’єктів й оновлення потрібної SBM-інформації в блоці PSB. Повідомлення RSVP PATH-TEAR використовується для видалення з блоку PSB відповідного RSVP-сеансу.

Повідомлення RSVP RESV не вимагають ніяких змін. SBM-клієнт, якому необхідно зробити резервування ресурсів після обробки отриманого повідомлення RSVP PATH, дотримується стандартних правил RSVP і відправляє повідомлення RSVP RESV по L2/L3-адресі попереднього переходу (адреса DSBM-станції сегмента) отриманого повідомлення PATH, ґрунтуючись на інформації з PSB-блоку сеансу [Д: 26].

DSBM-станція обробляє повідомлення RSVP RESV SBM-клієнта, ґрунтуючись на доступній пропускній здатності мережі. Якщо запит задовольнити неможливо, SBM-клієнтові, що запитує резервування, відправляється повідомлення RSVP RESVERR. Якщо запит можна задовольнити, повідомлення RESV відправляється за адресою попереднього переходу отриманого повідомлення PATH, узятого з PSB-блоку сеансу. Так само, як й у стандартному протоколі RSVP, DSBM, при можливості, поєднує запити про резервування [Д: 26].

Запитання для самоперевірки та контролю засвоєння знань

1. Які основні властивості технології ATM щодо передачі даних ?
2. Яким чином технологія ATM надає QoS-гарантії?
3. Які параметри QoS містить дескриптор потоку в технології ATM?
4. Яка процедура дотримання параметрів контракту QoS використовується в технології ATM?
5. Які характеристики покладено в основу поділу послуг на класи в технології ATM?
6. Що характерно для послуг класу A?
7. Що характерно для послуг класу B?
8. Що характерно для послуг класу C?
9. Що характерно для послуг класу D?
10. Що характерно для обслуговування з постійною бітовою швидкістю (CBR) в технології ATM?
11. Що характерно для обслуговування зі змінною швидкістю передачі в режимі реального часу (Real-Time Variable Bit Rate – RT-VBR) в технології ATM?

12. Що характерно для обслуговування зі змінною швидкістю передачі не в режимі реального часу (Non-Real Time Variable Bit Rate – nRT-VBR) в технології ATM?

13. Що характерно для обслуговування з доступною швидкістю передачі (Available Bit Rate – ABR) в технології ATM?

14. Що характерно для обслуговування з невизначеною бітовою швидкістю (Unspecified Bit Rate – UBR) в технології ATM?

15. Який біт у заголовку ATM-комірки вказує на пріоритет втрати комірки?

16. Що характерно для функції часткового відкидання пакетів (Partial Packet Discard – PPD) у технології ATM?

17. Що характерно для функції раннього відкидання пакетів (Early Packet Discard – EPD) у технології ATM?

18. Як забезпечуються функції IP QoS при передачі трафіку через один віртуальний канал по ATM-мережі?

19. Як забезпечуються функції IP QoS при передачі трафіку через віртуальний шлях з декількох віртуальних каналів по ATM-мережі?

20. В чому, з точки зору забезпечення IP QoS , полягає перевага використання віртуального шляху перед використанням віртуального каналу при передачі трафіку по ATM-мережі?

21. Які показники якості обслуговування підтримує технологія Frame Relay?

22. Які біти в заголовку Frame Relay кадра використовуються для реалізації механізму запобігання перевантаженню?

23. Про що свідчить значення FECN-біта, яке дорівнює 1?

24. Про що свідчить значення BECN-біта, яке дорівнює 1?

25. Про що свідчить значення DE-біта, яке дорівнює 1?

26. Яку функцію виконує механізм вирівнювання трафіку в мережі Frame Relay?

27. Які параметри використовуються в механізмі вирівнювання трафіку в мережі Frame Relay?

28. Як визначається контрольний інтервал часу в механізмі вирівнювання трафіку в мережі Frame Relay?

29. Чи надає гарантії щодо затримки передачі кадрів технологія Frame Relay?

30. Які умови перевіряються за контрольний інтервал часу в механізмі вирівнювання трафіку в мережі Frame Relay?

31. У чому полягає принцип функціонування алгоритму «дірявого відра» в технології Frame Relay?

32. Якому правили повинні відповідати угоди про якість обслуговування « в технології Frame Relay?

33. Що характерно для адаптивного механізму вирівнювання трафіку в мережі Frame Relay?

34. Який біт заголовку кадру Frame Relay використовується в адаптивному механізму вирівнювання трафіку?

35. Який принцип функціонування адаптивного механізму вирівнювання трафіку в мережі Frame Relay?

36. У чому полягає проблема використання адаптивного механізму вирівнювання трафіку в мережі Frame Relay в дейтаграмному режимі передачі?

37. Як вирішується проблема використання адаптивного механізму вирівнювання трафіку в мережі Frame Relay в дейтаграмному режимі передачі?

38. Яким чином вирішується проблема передачі високопріоритетних пакетів малого розміру одночасно с іншими пакетами великого розміру в технології Frame Relay?

39. Для вирішення якої задачі необхідна міжмережна взаємодія IP – Frame Relay QoS?

40. За допомогою яких протоколів виконується задання пріоритету Ethernet-кадру?

41. Чи можливо вказати пріоритет нетегованого Ethernet-кадру?

42. Наведіть приклади ділянок Ethernet-мережі, де можливо використати пріоритети кадрів для їх подальшого обслуговування.

43. Чи можливе гарантоване резервування смуги пропускання в Ethernet-мережі і чому?

44. Яку задачу вирішує протокол SBM в Ethernet-мережі?

45. Наведіть принцип функціонування протоколу SBM в Ethernet-мережі.

9. ОСОБЛИВОСТІ ТЕХНОЛОГІЇ MPLS

Технологія MPLS (Multiprotocol Label Switching – MPLS – мультипротокольна комутація міток) була розроблена для організації єдиного протоколу передачі різного виду трафіку через проміжні мережі, як з комутацією каналів, так і з комутацією пакетів, в яких використовуються різні протоколи мережного і канального рівнів [О: 10, 11, Д: 38, 39].

Мультипротокольна комутація міток (Multiprotocol Label Switching – MPLS) – це стандарт, розроблений робочою групою по створенню інтегрованих послуг (Internet Engineering Task Force – IETF) для нового типу комутації, що дозволяє комутувати пакети на канальному рівні (рівень 2 моделі OSI), використовуючи при цьому інформацію про просування даних мережного рівня (рівень 3 моделі OSI). Таким чином, у протоколі MPLS сполучаються високоефективні можливості комутації канального рівня й масштабованість передачі даних мережного рівня. Мультипротокольність технології MPLS проявляється у можливості роботи з різними протоколами мережного рівня – як протоколами передачі (наприклад, IP, IPX) так і протоколами маршрутизації.

Технологія MPLS об’єднує техніку віртуальних каналів з функціональністю стека TCP/IP. Об’єднання проходить за рахунок того, що один і той же мережний пристрій, який називається комутуючим по міткам маршрутизатором, виконує функції як IP-маршрутизатора, так і комутатора віртуальних каналів.

Основними функціями MPLS є [О: 10, 11, Д: 38, 39]:

– прискорена комутація пакетів з використанням міток;
– забезпечення гарантованого якості обслуговування (QoS) – розвиток існуючих механізмів IP QoS, що надає пріоритетну обробку певних інформаційних потоків на основі атрибутів QoS;
– керування трафіком (TE) – забезпечення ефективного використання доступної пропускної здатності мережі й захищених послуг;

– віртуальні приватні мережі (MPLS VPN) – побудова приватних мереж на основі MPLS як третього рівня (з використанням протоколу BGP) так і другого рівня (з використанням технології Any Transport over MPLS – Atom).

Ефективність роботи MPLS забезпечується за рахунок додавання міток до IP-пакетів. Передача інформації здійснюється за принципом, аналогічному, що використовується в мережах ATM, – комутація по заздалегідь прорахованій таблиці. Таблиця комутації складається із правил комутації, що визначають вихідний інтерфейс для кожного значення мітки, що символізує певний префікс мережі. Використання міток дозволяє суттєво розвантажити ядро мережі, позбавивши його від необхідності підтримувати громіздкі таблиці маршрутизації, і дозволяє здійснювати незалежну маршрутизацію для декількох замовників. ПО Cisco IOS використовує два протоколи для складання таблиць комутації: Label Distribution Protocol (LDP) і Resource Reservation Protocol / Traffic Engineering (RSVP/TE).

9.1. Особливості архітектури

В архітектурі MPLS виділяють два функціональні рівні [О: 10, 11, Д: 38, 39]:

– рівень керування – забезпечує побудову таблиці комутації (Forwarding Table);
– рівень комутації – передає пакети на вихідний інтерфейс, використовуючи записи побудованої таблиці комутації.

В результаті, для підтримки технології MPLS необхідно реалізувати два компоненти: керуючий та передавальний. Керуючий компонент (control component) використовує протокол поширення міток для підтримки інформації про просування міток для всіх точок призначення MPLS-мережі. Передавальний компонент (forwarding component) комутує пакети шляхом заміни мітки з використанням інформації, що передається в пакеті, і інформації про просування міток, підтримуваної керуючим компонентом. Як видно з назви, технологія MPLS підтримує різні протоколи мережного рівня.

Тому передавальний компонент не залежить від якогось конкретного протоколу. Для того щоб MPLS можна було використовувати з декількома протоколами мережного рівня, керуючий компонент повинен підтримувати поширення міток для різних протоколів рівня 3 моделі OSI.

9.2. Структура MPLS мережі

Мережа MPLS складається з ядра на основі маршрутизаторів з комутацією по мітках (Label Switch Router, LSR), оточеного прикордонними маршрутизаторами з комутацією по мітках (Edge Label Switch Router, Edge LSR або LER) – рис. 9.1. LER на вході в MPLS мережу додають мітку MPLS до пакету даних, а LER на виході з MPLS мережі видаляє мітку MPLS з пакету даних. У деяких випадках пакет даних який надійшов на порт LER вже може містити мітку, тоді новий LER додає другу мітку в пакет даних [О:10, 11, Д: 38, 39].

Мітки між LER і LSR розподіляються за допомогою LDP (Label Distribution Protocol – протокол розподілу міток) [Д:40, 41]. Для того, щоб отримати повну картину MPLS мережі LSR постійно обмінюються мітками та інформацією про кожного сусіднього вузла, використовуючи стандартну процедуру. Шлях, по якому проходить пакет в MPLS мережі, називають Label Switching Path (LSP).

При описі віртуальних приватних мереж, заснованих на технології MPLS, розташовані на вході або виході мережі LER зазвичай називаються PE-маршрутизатором (Provider Edge – маршрутизатори на кордоні мережі провайдера), а вузли, що працюють як транзитні маршрутизатори, називаються P-маршрутизаторами (Provider – маршрутизатори провайдера). Мережі клієнтів (Мережі С) підключаються до операторської мережі (Мережа Р) за допомогою маршрутизаторів CE (Customer Edge). Маршрутизатор CE являє собою звичайний IP маршрутизатор і, як правило, не виконує функцій MPLS. Маршрутизатори LSR забезпечують комутацію інформації на основі міток, призначених маршрутизаторами LER [О: 10, 11, Д: 38, 39].

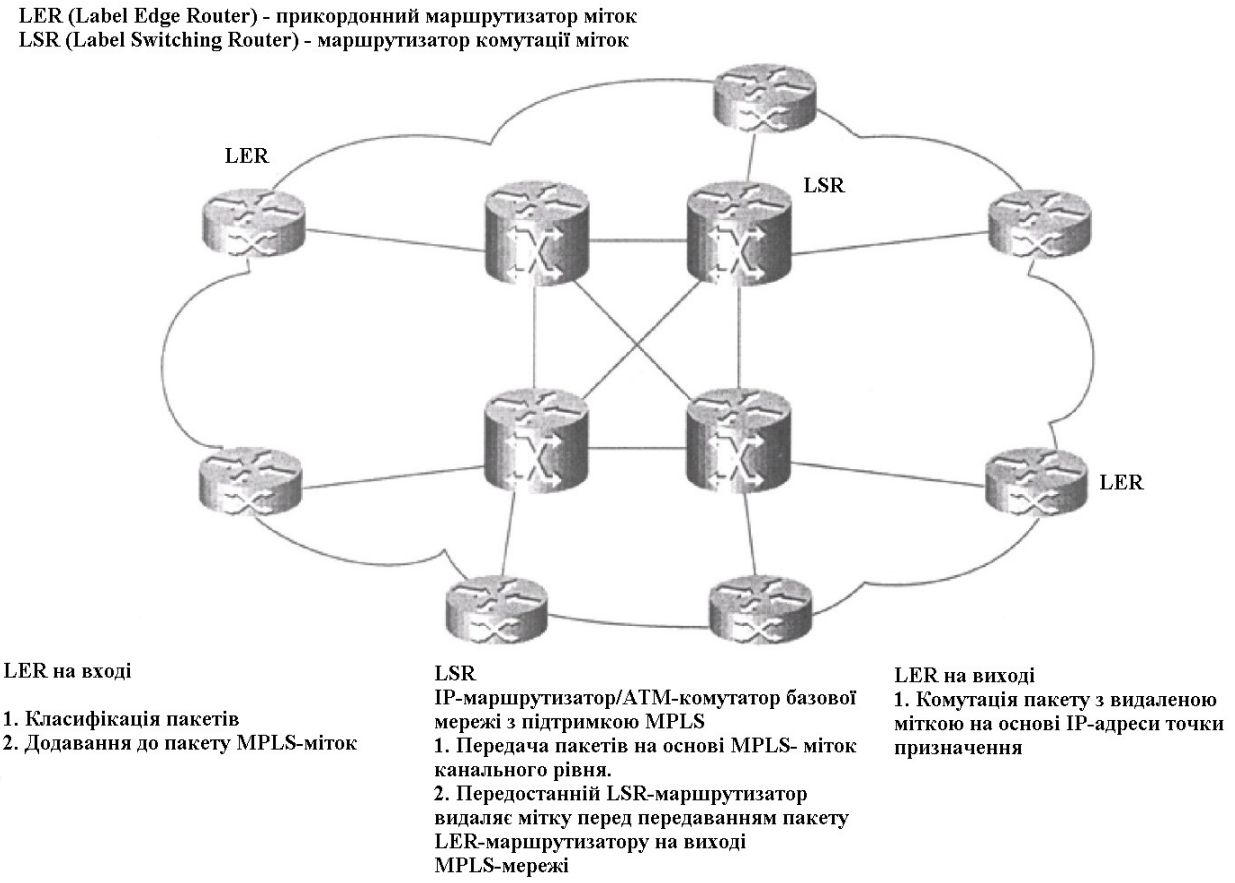

Рисунок 9.1 – Структура MPLS мережі

LER-маршрутизатор (Label Edge Router – LER) є прикордонним маршрутизатором MPLS-хмари. Деякі з його інтерфейсів беруть участь у просуванні пакетів з використанням MPLS, а деякі – ні. LER додає MPLS-мітку до усіх пакетів, що надходять в MPLS-мережу через не MPLS-інтерфейси. На виході з мережі, він видаляє MPLS-мітку з пакетів, які залишають MPLS-мережу.

Не MPLS-маршрутизатор маршрутизує пакети на основі адреси пункту призначення з використанням найбільш оптимального маршруту з таблиці маршрутизації. MPLS-маршрутизатор використовує коротку мітку, що розміщається перед заголовком 3-го рівня, і ухвалює рішення щодо подальшої передачі пакета, ґрунтуючись на точному збігу мітки із записом у власній базі LFIB (Label Forwarding Information Base). У MPLS маршрутизаторі пакет з MPLS-міткою комутується на наступний порт після пошуку відповідного запису в таблиці комутації замість пошуку в таблиці маршрутизації.

9.3. Принцип роботи

В технології MPLS використовується поняття класу еквівалентності пересилки (FEC – Forwarding Equivalency Class) [О: 10, 11, Д: 38, 39]. Клас еквівалентності використовується для визначення групи пакетів з однаковими вимогами до напряму їх передачі (усі пакети групи однаково будуть оброблятися маршрутизатором і однаково направляються до пункту призначення). Прикладом FEC є усі IP-пакети з адресами вузлів призначення з однаковим префіксом мережі. Можливі також варіанти FEC на основі юнікастової адреси і ще якогось поля заголовку IP-пакету, наприклад, поля типу обслуговування (ToS).

Кожен LSR розподіляє усю множину пакетів, що надходять, на класи еквівалентності і ставить у відповідність кожному FEC напрямок пересилки пакетів. З точки зору вибору наступного маршрутизатора для пересилки, усі пакети, які належать одному класу, однакові. В MPLS – мережі пакет приписується до одного класу FEC тільки один раз, коли він надходить до мережі. Цьому FEC привласнюється мітка – ідентифікатор фіксованої довжини, яка передається разом з пакетом. Завдяки цьому, у наступних маршрутизаторах немає необхідності аналізувати заголовок 3-го рівня пакету.

Просування пакетів у мережі MPLS здійснюється за допомогою методу заміни міток [О: 10, 11, Д: 38, 39]. Коли пакет з міткою попадає в маршрутизатор комутації міток (Label Switching Router – LSR), останній використає мітку як покажчик у своїй інформаційній базі міток (LFIB). Для кожної мітки, в базі LFIB знаходиться точний запис про відповідну вихідну мітку, інтерфейс й інформацію про інкапсуляції канального рівня яка, необхідна для просування пакета. Опираючись на інформацію з бази LFIB, LSR заміняє отриману мітку на вихідну й передає пакет на вихідний інтерфейс із відповідною інкапсуляцією канального рівня.

Технологія MPLS основана на обробці заголовка MPLS, який додається до кожного пакету даних. Заголовок MPLS може складатися з однієї або кількох «міток». Кілька записів (міток) в заголовку MPLS називаються стеком міток.

Існують дві основні архітектури MPLS: cell-mode (мітки додаються у заголовок комірки, наприклад VPI/PCI в ATM) та frame-mode (мітки додаються між заголовками Layer2 і Layer3 в пакеті) [О: 10, 11, Д: 38, 39] (рис. 9.2).

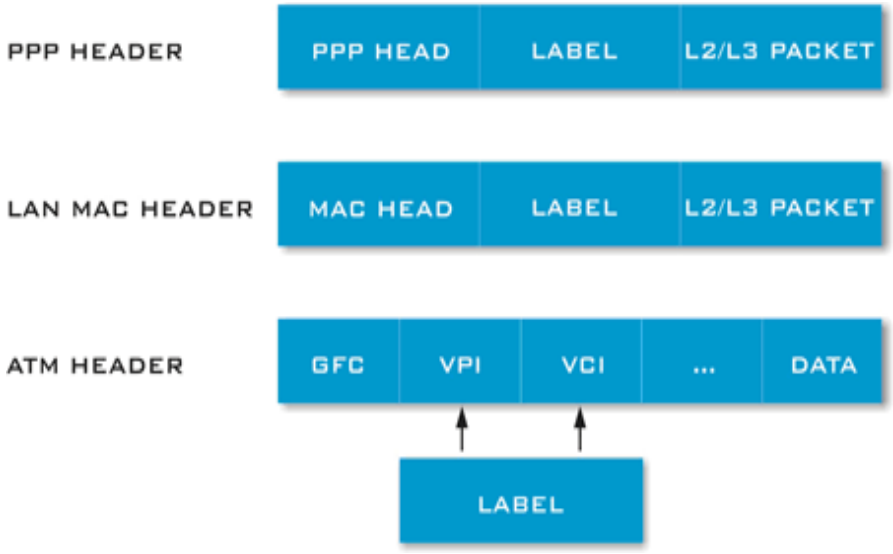

Рисунок 9.2 – Підходи до додавання MPLS – міток

Кожен запис в стеку міток складається з наступних чотирьох полів:

1) значення мітки (займає 20 біт);

2) поле класу трафіку (Traffic Class), необхідного для реалізації механізмів QoS і явного повідомлення про перевантаження (Explicit Congestion Notification, ECN) (займає 3 біти);

3) прапор дна стеку (Bottom of stack). Якщо прапор встановлений, то це означає, що поточна мітка остання в стеці. Займає 1 біт;

4) поле TTL (Time To Live) (займає 8 біт).

Формат 4-байтової MPLS-мітки представлений на рис. 9.3.

20 біт	3 біти	1 біт	8 біт
Мітка	CoS	S	TTL

Рисунок 9.3 – Формат 4-байтової MPLS-мітки

Поле CoS необхідне для надання диференційованих послуг в MPLS-мережі. Для наскрізного забезпечення послуг IP QoS на границі MPLS-мережі можна

скопіювати поле IP-пріоритету в поле CoS. Поле CoS в MPLS-заголовку має тільки 3 біти. Таким чином, у ньому може передаватися тільки 3-бітове поле IP-пріоритету, а 6-бітове поле коду диференційованої послуги (Differentiated Services Code Point- DSCP) – ні. При необхідності, інформація CoS може передаватися у вигляді однієї з міток MPLS-стека. Поле мітки має розмір 20 біт і здатний умістити як поле IP-пріоритету, так і поле DSCP.

Кожний LSR підтримує таблицю LFIB (Label Forwarding Information Base), в якій відображені відповідності вхідної мітки, вхідного інтерфейсу, маршруту і вихідної мітки та вихідного інтерфейсу. Значення мітки є унікальним лише для участку шляху між сусідніми LSR.

Вхідний LSR на границі MPLS домену (Edge LSR) отримує пакет без міток і виконує операцію додавання мітки. На шляху проходження пакету виконується послідовний обмін мітоk на проміжних LSRax (Core LSR). На прикінці, вихідний LSR (Edge LSR) виконує видалення мітки.

Обмін інформацією про мітки між LSRами виконується попарно, за допомогою спеціального протоколу розповсюдження міток – Label Distribution Protocol (LDP) [Д: 40, 41], або модифікації протоколу RSVP (TE-RSVP) (рис. 9.4).

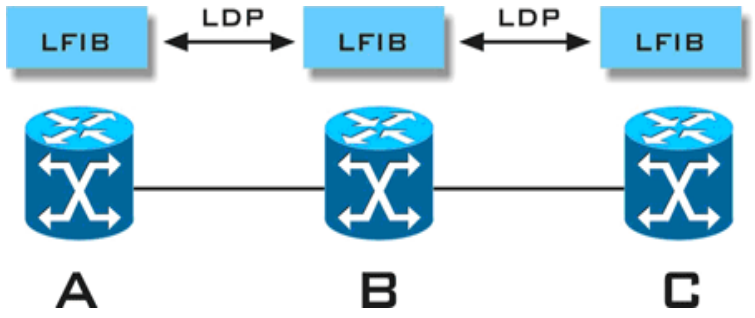

Рисунок 9.4 – Процедура обміну інформацією про мітки

Процес призначення міток

Розглянемо процес призначення міток в MPLS мережі [О: 10, 11, Д: 38-41]. Припустимо, в MPLS домені знаходяться LSR маршрутизатори A, B і C (рис. 9.5). Всі вони отримали топологічну інформацію про IP-мережі за допомогою протоколів маршрутизації (OSPF, IS-IS, BGP та ін.).

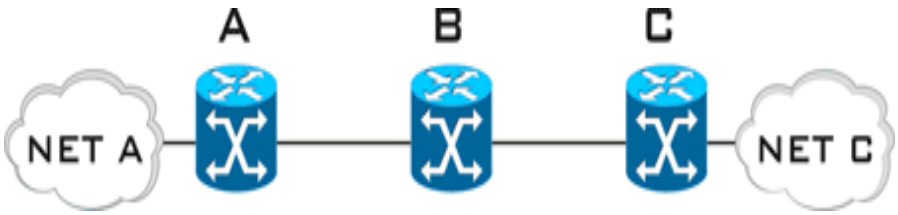

Рисунок 9.5 – Спрощена структура MPLS домену

LSR C знає (має маршрут) про підключену безпосередньо до нього мережу NET C. LSR C довільно призначає для мережі NET C вхідну мітку (наприклад, LABEL = X) і прописує це в свою LFIB (рис. 9.6).

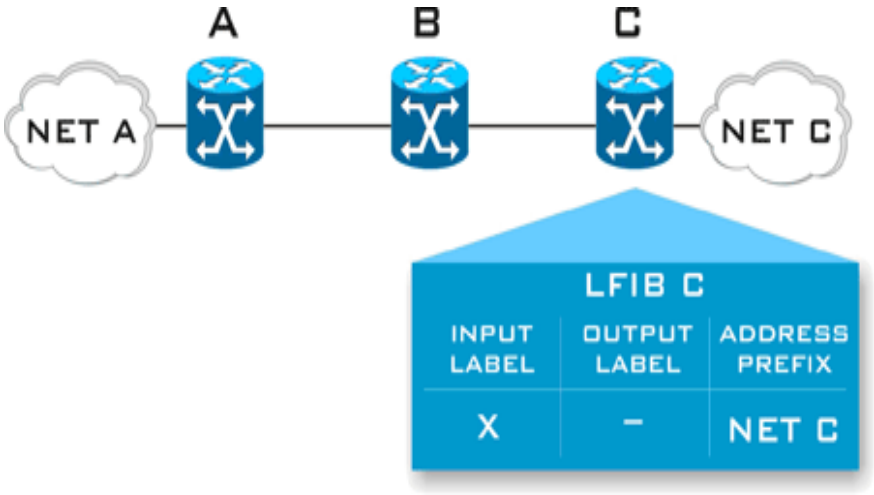

Рисунок 9.6 – Етап призначення міток маршрутизатором C

За допомогою протоколу LDP LSR C повідомляє сусіднього LSR B про призначення мітки X для мережі NET C, яку LSR B вписує в свою LFIB як вихідну для інтерфейсу (рис. 9.7).

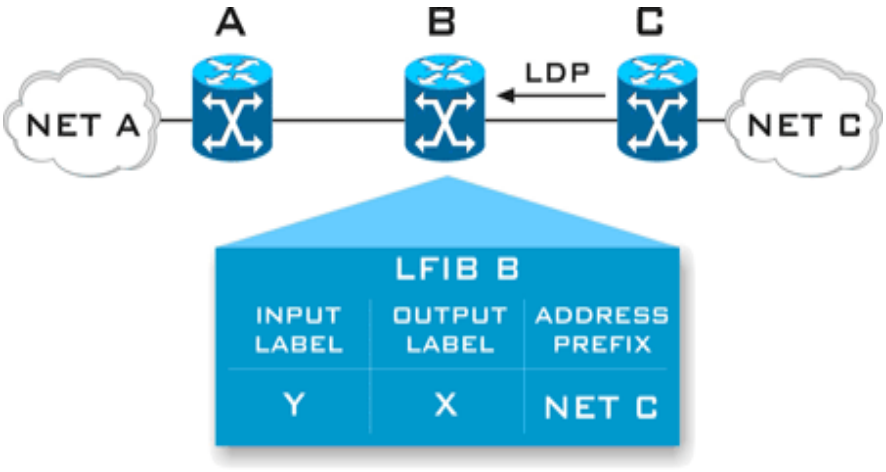

Рисунок 9.7 – Етап призначення міток маршрутизатором B

Додатково, LSR B довільно призначає вхідну мітку для NET C (Y) і за допомогою LDP повідомляє про неї LSR A. LSR A заповнює свою LFIB записом про мережі NET C (рис. 9.8). Розподіл міток між LSRами в підсумку призводить до встановлення шляхів LSP в MPLS домені.

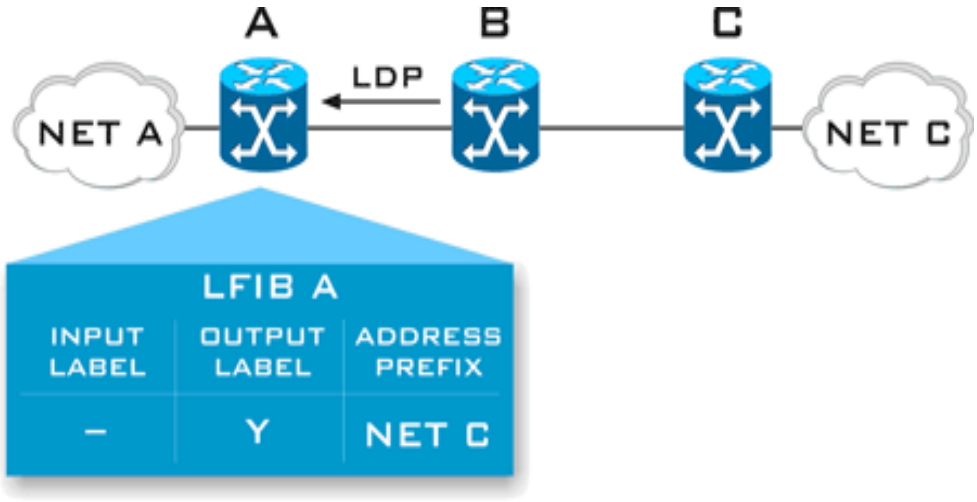

Рисунок 9.8 – Етап призначення міток маршрутизатором A

Комутація пакетів за мітками

Розглянемо процес комутації пакетів в MPLS мережі [О: 10, 11, Д: 38, 39]. Припустимо, LSR A отримує пакет адресований в мережу NET C (DESTINATION = NET C). LSR A вивчає свою LFIB і знаходить там запис з NET C і відповідні їй мітку Y і вихідний інтерфейс. До отриманого пакету додається мітка Y і пакет передається далі через відповідний інтерфейс (рис. 9.9).

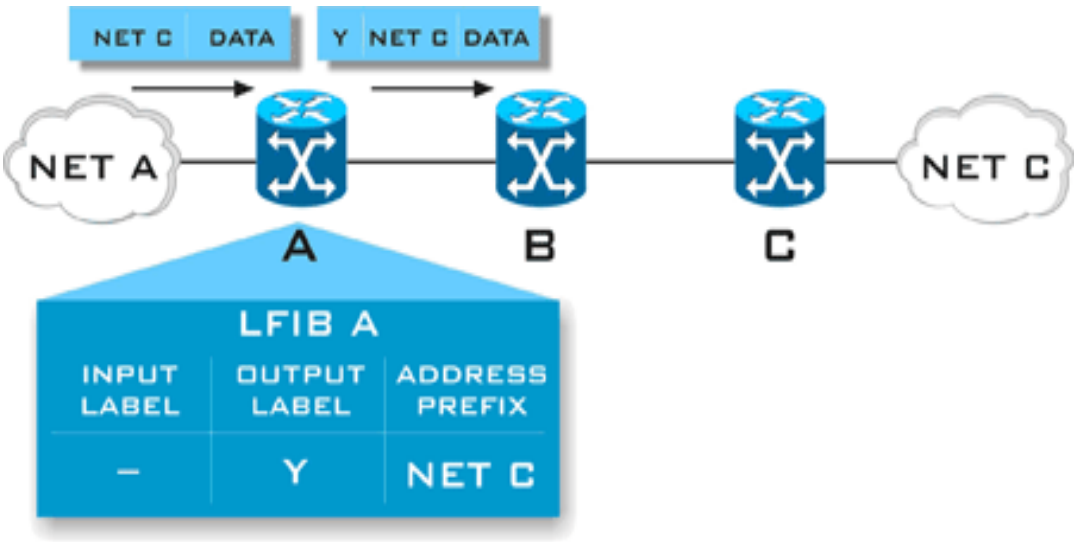

Рисунок 9.9 – Процес комутації пакетів в MPLS мережі

LSR B отримує пакет з міткою Y на певному інтерфейсі. За своєю LFIB LSR B з'ясовує через який інтерфейс і з якою міткою (X) необхідно просувати пакет далі і виконує необхідну заміну (swapping) мітки. Зауважимо, що на даному етапі заголовок мережевого рівня не аналізується – відбувається комутація по мітці (рис. 9.10).

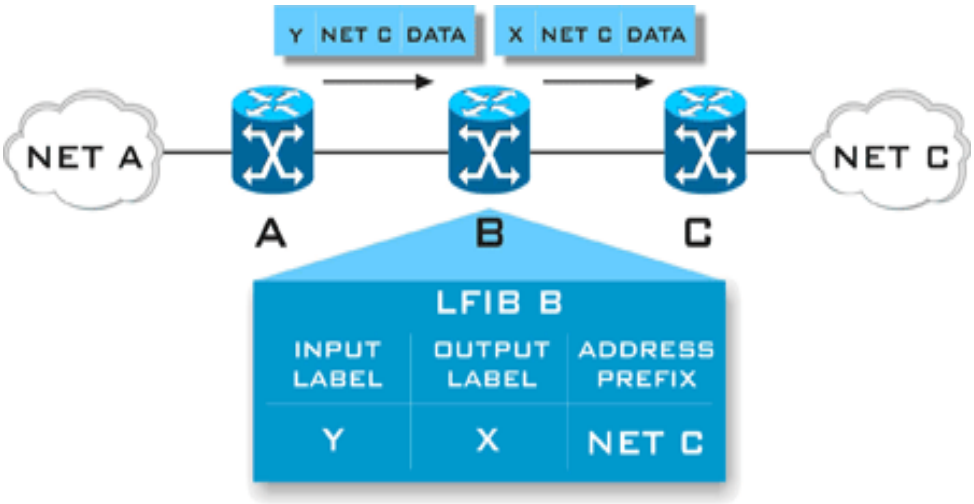

Рисунок 9.10 – Заміна міток при комутації

І, нарешті, LSR C отримує пакет з міткою X на вхідному інтерфейсі. LSR C видаляє мітку з пакету і відправляє його в мережу призначення як звичайний IP-пакет (рис. 9.11).

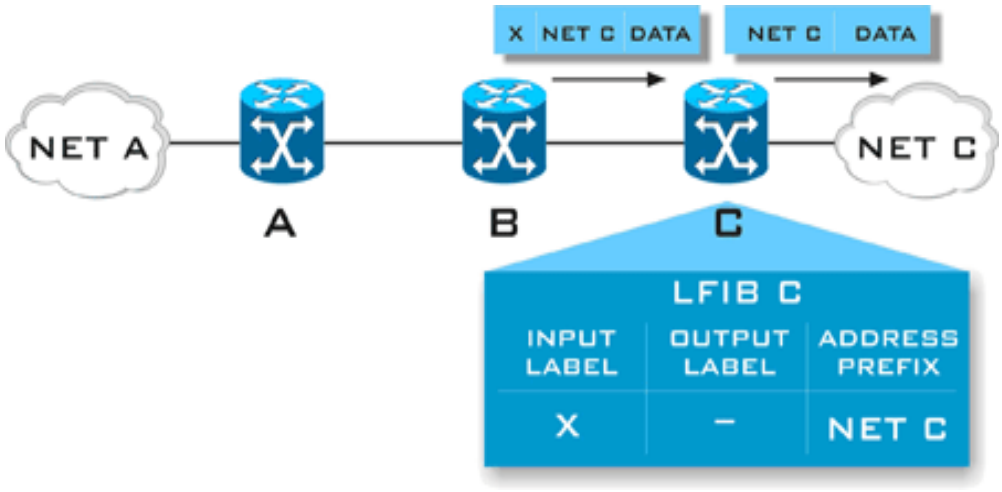

Рисунок 9.11 – Передача пакету в мережу призначення на виході MPLS мережі

Стекування міток

MPLS допускає використання не однієї мітки, а цілого набору послідовних ідентифікаторів – стека міток. У стеку розрізняються верхня мітка (Top Label) і всі нижчерозташовані. Стандартний механізм комутації оперує тільки з верхньої міткою. Стек використовується у випадку розширень MPLS, таких як тунелі, управління трафіком та інших [О: 10, 11, Д: 38, 39].

Тунелі

MPLS тунель застосовується для визначення заданого (детермінованого) маршруту для пакетів – LSP . MPLS тунель можна розглядати як інкапсуляцію MPLS в MPLS (за аналогією з IP-IP тунелюванням – MPLS всередині MPLS).

Для прикладу розглянемо MPLS домен, що містить LSRи A, B, C, D, E і створимо в ньому LSP ABDE з використанням механізму тунелювання (рис. 9.12).

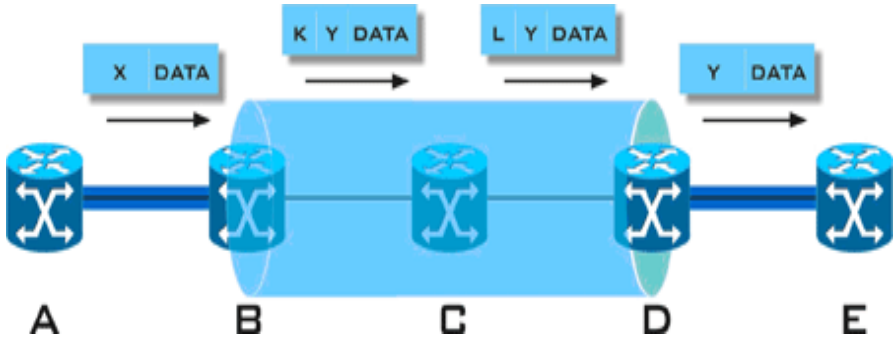

Рисунок 9.12 – Приклад створення тунелю через MPLS мережу

LSR A посилає до LSR B пакет з однією міткою X (Level 0). LSR B визначає , що пакет повинен увійти в тунель і заміщає мітку X на мітку Y для кінця LSP – LSR E , тобто як – ніби наступним є LSR E. LSR B додає в стек ще одну мітку K (Level 1), яка використовується всередині тунелю для подальшого просування у бік LSR D звичайним чином. Далі LSR D видаляє мітку верхнього рівня L і до LSR E пакет приходить вже з міткою Y , яку вставляв ще LSR B при входженні пакета в тунель. Таким чином, пакет долає кілька ділянок, комутація на яких відбувається за допомогою міток одного рівня (можливо різного) . У нашому конкретному випадку LSP нульового рівня – ABDE, LSP першого рівня – B-C-D. З точки зору трафіку нульового рівня, LSP першого рівня є прозорим тунелем . Звідси назва – MPLS тунель.

9.4. Методи призначення й поширення міток

Прив’язка мітки (label binding) – це зв’язок між міткою й інформацією про досяжність кінцевої точки мережного рівня, який не залежить від ступеня деталізації відомостей про просування даних.

MPLS пропонує три методи призначення й поширення міток [О: 10, 11, Д: 38-41]:

- спадне призначення міток;
- спадне призначення міток на вимогу;
- висхідне призначення міток.

Для всіх цих типів розміщення міток використовується протокол поширення міток (Label Distribution Protocol – LDP), що поширює мітки серед маршрутизаторів. Терміни «спадний» й «висхідний» використовуються стосовно напрямку потоку даних.

Спадне призначення міток

Спадне призначення міток (downstream label allocation) відбувається у зворотному напрямку по відношенню до потоку даних (від отримувача до джерела). Мітка, передана в пакеті, генерується й прив’язується до префікса мережі LSR-маршрутизатором наприкінці каналу по напрямку потоку даних. Таким чином, кожен LSR створює мітки для своїх безпосередньо підключених мереж, прив’язує їх як вхідні мітки префіксів і поширює інформацію про мітки серед усіх маршрутизаторів, розташованих вище по потоку даних. Маршрутизатор, що перебуває вище по потоку, заносить отриману прив’язку до таблиці комутації як вихідну мітку префіксу й, у свою чергу, створює для нього вхідну мітку, передаючи її наступному, розташованому вище по потоку даних маршрутизатору.

При спадному призначенні міток MPLS-маршрутизатор розміщає вхідну мітку для кожного елемента таблиці комутації і поширює її на кожний зі своїх інтерфейсів. Розташований вище по потоку маршрутизатор одержує прив’язку мітки й використовує її як вихідну мітку для свого асоційованого префіксу. У цьому методі призначення міток вхідна мітка призначається локально,

а вихідна – надходить від розташованого нижче по потоку маршрутизатора (рис. 9.13).

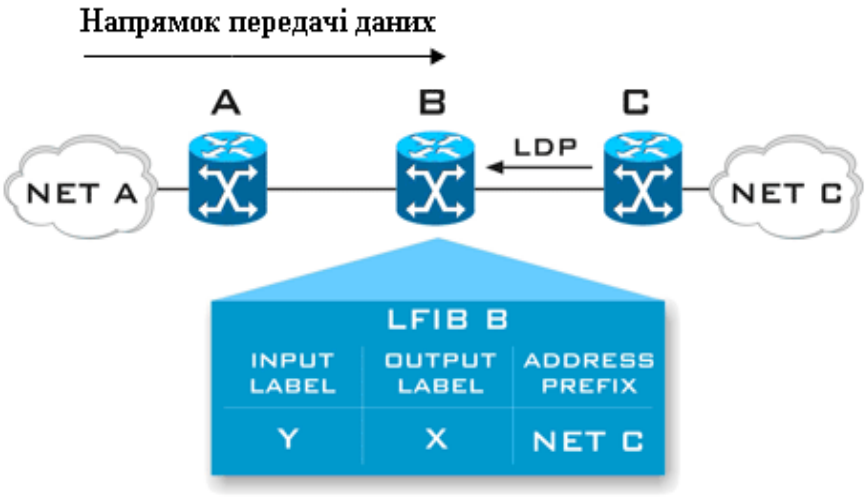

Рисунок 9.13 – Спадне призначення міток

Спадне призначення міток на вимогу

Цей процес призначення міток схожий на спадне призначення, однак він ініціюється на вимогу маршрутизатора, розташованого вище по потоку. Даний маршрутизатор ідентифікує наступну точку призначення для кожного префіксу з таблиці комутації і відправляє в неї запит на створення прив'язки мітки для цього маршруту. Остання частина процесу призначення мітки збігається із процесом спадного призначення міток.

Висхідне призначення міток

Висхідне призначення міток (upstream label allocation) відбувається по напрямку потоку даних (від джерела до отримувача). Мітка, передана в заголовку пакета даних, генерується й прив'язується до префікса мережі LSR-маршрутизатором на початку каналу по напрямку потоку даних. Для кожного запису таблиці комутації LSR-маршрутизатора вихідна мітка призначається й поширюється серед маршрутизаторів, розташованих нижче за потоком, як вхідна мітка. У цьому випадку вхідні мітки поширюються префіксами мереж (рис. 9.14).

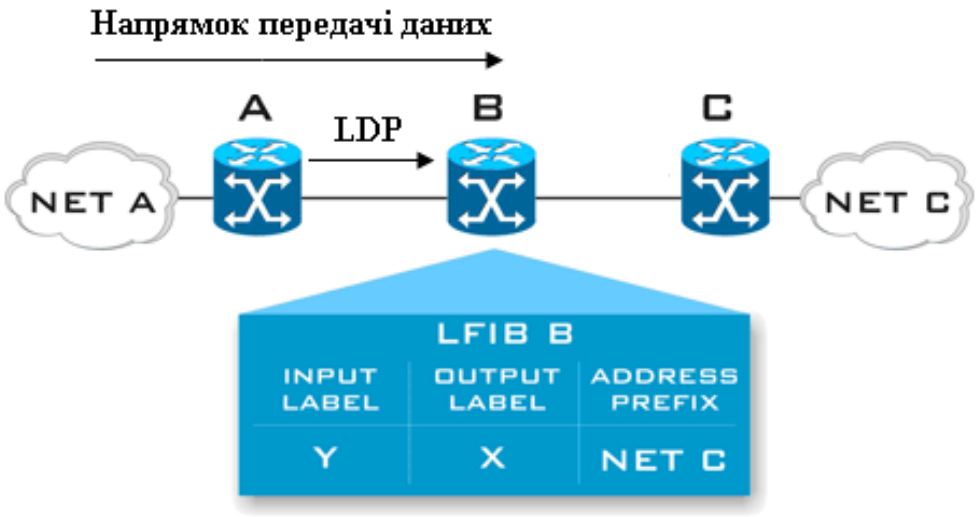

Рисунок 9.14 – Висхідне призначення міток

Коли у LSR-маршрутизатора є й вхідна і вихідна мітка префіксу, він може почати комутувати пакети шляхом заміни міток. Коли LSR створює прив'язку вихідної мітки до маршруту, він, крім заповнення бази LIB, також оновляє свою таблицю комутації інформацією про прив'язку. Це дозволяє LSR-маршрутизатору додавати мітки до непомічених пакетів при їхньому появленні.

Метод висхідного поширення міток розроблявся, перш за все для ATM-мереж, тоді як метод спадного поширення міток орієнтований на IP-мережі.

9.5. Особливості протоколу LDP (Label Distribution Protocol)

Протокол LDP регламентується стандартом IETF «LDP Specification» RFC 3036. Протокол LDP функціонує в рамках архітектури MPLS [О: 10, 11, Д: 40, 41].

Призначення протоколу LDP

Протокол LDP призначений для побудови цілісних LSP-маршрутів комутації за мітками.

Встановлення сусідських відносин

Встановлення сусідських відносин між маршрутизаторами здійснюється за двома фазами:

1) обмін повідомленнями Hello;

2) встановлення сесії LDP.

Фаза 2 виконується тільки за успішного виконання фази 1.

Обмін повідомленнями Hello

Повідомлення Hello надсилаються маршрутизатором через усі інтерфейси, на яких включений LDP, кожні 15 секунд на адресу 224.0.0.2 (all – routers), порт 646, транспортний протокол UDP.

Можливий обмін повідомлень Hello і між LSR-ами, які не сполучені безпосередньо. У цьому випадку повідомлення надсилається з використанням юнікастової адреси.

Повідомлення Hello містять таку інформацію:

Holddown timer – проміжок часу, протягом якого сусіди повинні послати хоча б одне повідомлення Hello. Якщо сусіди пропонують різне значення, то приймається мінімальне. Враховуючи, що протокол UDP не забезпечує гарантії доставки, то рекомендується посилати повідомлення Hello з періодом в три рази меншим, ніж Holddown timer. Якщо Holddown timer дорівнює 0, то приймаються наступні значення за замовчуванням:

– 15 с – для звичайних hello повідомлень на адресу all – routers;
– 45 с – для повідомлень посилаються на конкретну адресу (Targeted Hello).

Значення Holddown timer рівне 0xFFFF означає нескінченність, але навіщо воно – RFC замовчує.

T-bit – (Targeted Hello) якщо даний біт 1, то це означає, що повідомлення відправлено на конкретну (unicast) адресу, інакше повідомлення відправлено на адресу 224.0.0.2.

R-bit – (Request Send Targeted Hellos) якщо даний біт 1, то це означає, що одержувач повинен відповісти на це повідомлення (Hello), інакше не повинен. Даний біт може бути встановлений в 1, тільки у випадку якщо T-bit = 1.

Transport Address – дане поле в Hello пакеті не обов’язкове. Якщо воно присутнє, то адреса, вказана в ньому, використовується надалі для встановлення

LDP-сесії між пристроями. Якщо дане поле відсутнє, то для встановлення сесії повинна використовуватися адреса джерела Hello-пакета. Адресу, яка використовується для встановлення LDP-сесії, називають «транспортною адресою».

Configuration Sequence Number – поле містить номер конфігурації. При зміні налаштувань на LSR, відповідно змінюється цей номер. Зміна номера може викликати переустановлення LDP-сесії.

Повідомлення hello можуть відкидатися, і, відповідно, фаза 1 встановлення сусідських відносин може бути не виконана, в силу наступних причин:

– LSR-у адміністративно заборонено приймати повідомлення Hello певного типу (наприклад, TargetHello) або з певної адреси;
– не співпали типи транспортних адрес (наприклад, IPv4 і IPv6).

Встановлення LDP сесії

LDP сесія функціонує поверх стеку TCP/IP (порт 646 UDP) [О: 10, 11, Д: 40, 41]. LSR1 і LSR2 при обміні повідомленнями hello дізнаються транспортні адреси один одного. Якщо транспортна адреса LSR1 більше ніж транспортна адреса LSR2, то LSR1 стає «активним» сусідом , а LSR2 «пасивним». Якщо ні, то навпаки. Далі LDP-сесія встановлюється за таким сценарієм:

1. «Активний» сусід встановлює TCP/IP-сесію на порт 646.

2. «Активний» сусід посилає повідомлення Init з власними параметрами LDP-сесії.

3. «Пасивний» сусід перевіряє параметри LDP-сесії в повідомленні Init на предмет сумісності з локальними налаштуваннями LDP.

4. «Пасивний» сусід відповідає повідомленням Init з власними параметрами LDP-сесії.

5. «Активний» сусід перевіряє параметри LDP-сесії в повідомленні Init на предмет сумісності з локальними налаштуваннями LDP.

6. Сесія встановлена.

Якщо на якомусь етапі відбувається щось не передбачуване (приходить не той тип пакета, очікуване повідомлення не надходить взагалі, або не збігаються параметри LDP-сесії в повідомленні Init та ін.), то сесія вважається

не встановленою. LSR, який виявив помилку, посилає повідомлення Shutdown або Reject своєму сусідові.

Повідомлення Init

Повідомлення Init містить наступну інформацію:

Protocol Version – версія протоколу.

KeepAlive Time – максимальний час між службовими повідомленнями KeepAlive. Обидві сторони можуть пропонувати різні значення – використовуватися повинно мінімальне.

A-bit – Label Advertisement Discipline – режим обміну інформацією про мітки. Можливе використання двох режимів обміну інформацією про мітки:

– 1 – Downstream On Demand;

– 0 – Downstream Unsolicited.

D-bit – Loop Detection – механізм запобігання циклів LSP.

– 0 – вимкнено,

– 1 – увімкнено.

PVLim – Path Vector Limit – змінна, яка використовується для роботи механізму запобігання циклів.

Max PDU Length – максимальна можлива довжину суміщених повідомлень LDP в байтах. LDP повідомлення групуються в PDU (Protocol Data Units) і передаються в одному пакеті TCP/IP. Сусіди можуть пропонувати різні значення, але обидва повинні вибрати мінімальне. Зауважимо, що навіть одне повідомлення упаковується всередину PDU.

Receiver LDP Identifier – Ідентифікатор простору міток (або Label Space Identifier). Формат поля наступний: LSR_ID: Label_Space_ID. LSR_ID це ідентифікатор LSR. Даний ідентифікатор повинен бути унікальний в рамках домену MPLS і єдиний для кожного LSR-а. Label_Space_ID – це ідентифікатор множини міток. Label Space Identifier вказується в заголовку PDU, ідентифікуючи тим самим сусіда та інтерфейс за яким встановлено сусідство. Наприклад, два LSR-и можуть бути з’єднані двома каналами, і для кожного

каналу повинен бути виділений різний Label Space Identifier, відрізнятися які будуть тільки значенням Label_Space_ID.

LDP-сесія встановлюється при виконанні наступних умов:

– збіг версій протоколу;

– збіг значень А-біту. В мережі на різних з'єднаннях можливе використання різних режимів поширення інформації про мітки, але на одному з'єднанні режим повинен збігатися.

Розбіжність PVLim, відповідно до RFC не повинна призводити до закриття сесії , але може викликати попередження на LSR.

Повідомлення KeepAlive

За кожною сесією LDP LSR повинен закріплювати таймер. Після отримання будь-якого повідомлення LDP, LSR встановлює таймер в 00:00 і знову запускає його. До досягнення таймером значення «KeepAlive Time» сусідній LSR повинен надіслати будь-яке LDP повідомлення. Якщо у сусіда немає інформативних повідомлень для пересилання, то він повинен послати повідомлення KeepAlive. Якщо в призначений термін повідомлення не приходять, то сусід вважається відключеним і сесія з ним повинна встановлюватись заново.

Обмін інформацією про мітки

Розглянемо схему взаємодії LSR, представлену на рис. 9.15.

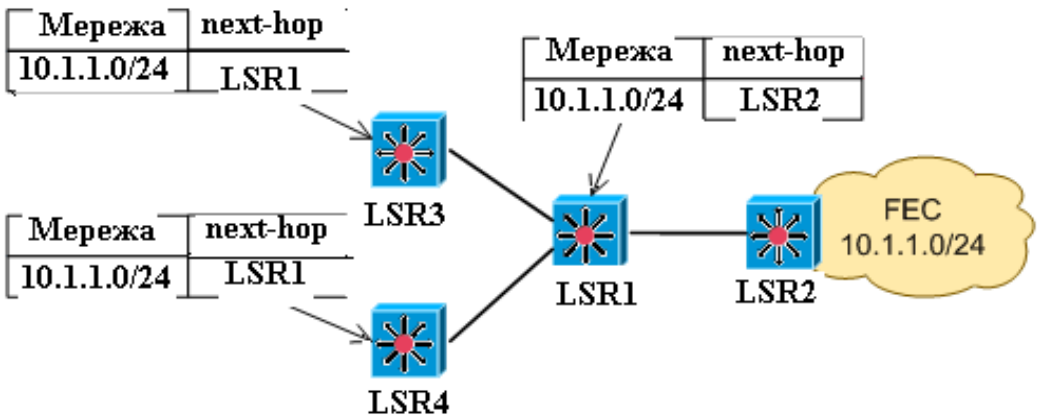

Рисунок 9.15 – Приклад схеми взаємодії LSR

З точки зору LSR1 для FEC 10.1.1.0/24 сусідів LSR1 можна розділити на дві групи:

– Upstream LSR-и – це LSR3, LSR4.

– Downstream LSR – це LSR2.

Downstream LSR називають next – hop – LSR (NH-LSR).

Параметри функціонування LDP

Існує кілька параметрів функціонування LDP [О: 10, 11, Д: 40, 41]:

– режим обміну інформацією про мітки (Label Distribution Mode),

– режим контролю над розповсюдженням міток (Label Distribution Control),

– механізм збереження міток (Label Retention Mode).

Режим обміну інформацією про мітки

Між сусідами можливо використання двох режимів обміну інформацією про мітки:

– Downstream On Demand – із запитом;

– Downstream Unsolicited – без запиту.

При режимі Downstream On Demand LSR повинен запитувати мітку для створення LSP (для FEC) від сусіднього LSR, який є next-hop-ом для цього FEC.

При режимі Downstream Unsolicited LSR для кожного FEC, який знаходиться у нього в таблиці IP-маршрутизації, призначає мітку і розсилає її всім своїм сусідам. Якщо для сусіднього LSR вихідний LSR є next-hop-ом, то мітка записується в таблицю комутації.

Механізм контролю над поширенням міток

Існує кілька механізмів контролю над поширенням міток (Label Distribution Control Mode) [О: 10, 11, Д: 40, 41]:

– Independent Label Distribution Control – незалежний контроль;

– Ordered Label Distribution Control – впорядкований контроль.

При використанні незалежного контролю над поширенням міток LSR може виділяти мітки для FEC своїм сусідам навіть у разі, якщо LSR не має вихідної мітки для себе від наступного LSR.

Якщо використовується упорядкований контроль над поширенням міток, то LSR не виділятиме мітки своїм сусідам, поки сам LSR не отримає вихідну мітку для заданого FEC від NH-LSR-а. При цьому режимі першим відсилає мітку той LSR, до якого безпосередньо приєднаний FEC.

Режим збереження міток

Існують наступні режими збереження міток (Label Retention Mode) [О: 10, 11, Д: 40, 41]:

- Conservative Label Retention Mode (стриманий режим збереження міток);
- Liberal Label Retention Mode (вільний режим збереження міток).

При використанні стриманого режиму збереження міток при знищенні маршруту на FEC-мітка видаляється. Для відновлення LSP необхідно, щоб мітка була заново виділена сусіднім NH-LSR-ом.

Якщо використовується вільний режим збереження міток, то, при знищенні маршруту на FEC, мітка не видаляється, а лише позначається як неактивна. І у випадку, якщо маршрут на FEC відновлюється через той самий NH-LSR, то мітка не запрошується, а використовується стара, статус якої змінюється на активний.

Протокол LDP повинен реагувати на такі події [О: 10, 11, Д: 40, 41]:

– поява нового запису FEC в таблиці маршрутизації;
– зникнення запису FEC з таблиці маршрутизації;
– зміна next-hop для запису FEC.

Можливі комбінації режимів роботи протоколу LDP та приклади функціонування наведені в табл. 9.1 [О: 10, 11, Д: 40, 41].

При зникненні запису FEC з таблиць маршрутизації всі LSR повинні обов’язково відкликати призначені мітки для комутації FEC у своїх сусідів. Робиться це за допомогою надсилання повідомлення Label Withdraw.

Таблиця 9.1 – Режими роботи протоколу LDP

Режим обміну інформацією про мітки	**Downstream Unsolicited**	**Downstream Unsolicited**	**Downstream Unsolicited**	**Downstream On Demand**	**Downstream On Demand**
Механізм контролю над розповсюдженням міток	Independed Control	Ordered Control	Ordered Control	Ordered Control	Independed Control
Режим утримання міток	liberal	liberal	conservative	conservative	conservative
Поява нового запису FEC	1)Відправляємо усім сусідам мітки на усі відомі FEC 2) Очікуємо мітку від NH-LSR 3)Використовуємо отриману мітку для комутації	1) Очікуємо, поки надійде мітка від NH-LSR 2) Відправляємо усім сусідам мітку на FEC 3) Використовуємо отриману мітку для комутації		1) Відправляємо запит NH-LSR на виділення мітки 2) Очікуємо відповідь 3) Використовуємо отриману мітку для комутації	
Зміна next-hop для запису FEC	1) Шукаємо мітку у списку «відкладених» 2) Якщо мітка не знайдена, відправляємо запит на NH-LSR про виділення мітки. Інакше п. 4 3) Очікуємо відповідь 4) Використовуємо отриману мітку для комутації	1) Відправляємо запит NH-LSR на виділення мітки 2) Очікуємо відповідь 3) Використовуємо отриману мітку для комутації			
Отримання запиту на виділення мітки	Виділяємо мітку **не очікуючи відповіді** від свого NH-LSR	Виділяємо мітку **тільки після відповіді від** свого NH-LSR.			Виділяємо мітку **не очікуючи відповіді** від свого NH-LSR

Механізм запобігання циклів

Протокол LDP має у своєму складі механізм запобігання циклів. Призначення цього механізму – не допускати зациклення запитів і маршрутів [О: 10, 11, Д: 40, 41]. Це досягається включенням у всі повідомлення Label Mapping і Label Mapping Request інформації про LSR через які дані запити пройшли. У разі якщо LSR-и функціонують в режимі Ordered Control даний ефект досягається легко. Якщо LSR-и використовують Independed Control , то у цьому випадку LSR-и повинні заново відсилати запити і відповіді на них, тому що інформація про LSR-и, через які пройшли запити, буде оновлюватися.

Механізм запобігання циклів може і не використовуватися, бо відсутність циклів повинен гарантувати протокол IP-маршрутизації, інформацію від якого використовує LDP. Зациклення можуть виникати на нетривалий період, тільки у випадку, якщо протокол IP-маршрутизації повільно сходиться, а LDP працює швидше, ніж протокол IP-маршрутизації.

Типи повідомлень LDP

У табл. 9.2 наведені типи LDP повідомлень [О: 10, 11, Д: 40, 41].

Таблиця 9.2 – Типи LDP повідомлень

Повідомлення	Опис
Hello	Повідомлення використовується для ідентифікації сусідів, встановлення фази N1 сусідських відносин
Init	Повідомлення використовується для встановлення сусідських відносин (фаза N2), обміну та узгодження параметрів LDP-сесії
KeepAlive	Повідомлення використовується для підтримки активного статусу LDP-сесії
Address Message	Повідомлення використовується для оповіщення сусідів про появу нових IP-мереж, які безпосередньо приєднані до LSR
Address Withdraw	Повідомлення використовується для оповіщення сусідів про зникнення IP-мереж, які безпосередньо приєднані до LSR
Label Mapping	Повідомлення містить опис FEC і мітку, призначену LSR-ом, який посилає повідомлення
Label Request	Даним повідомленням LSR запитує у сусідів мітку для комутації для конкретного FEC. Опис FEC включається в запит
Label Release	Підтвердження отримання мітки в повідомленні Label Mapping. Надсилається, якщо мітка запитувалася повідомленням Label Request
Label Abort Request	Скасування запиту на виділення мітки, який був попередньо надісланий в повідомленні Label Request
Label Withdraw	Відкликання призначеної мітки у сусіда. Сусід повинен припинити використовувати відкликану мітку для комутації

Побудова комутованого маршруту

Розглянемо, як система MPLS автоматично створює шлях LSP за допомогою протоколу LDP [О: 10, 11, Д: 40, 41].

Спочатку, за допомогою багатоадресної розсилки повідомлень Hello, комутуючі маршрутизатори визначають своє «сусідство» (суміжності) в рамках протоколу LDP. Крім близькості на канальному рівні, LDP може встановлювати зв’язок між «логічно сусідніми» LSR, що не належать до одного каналу. Це необхідно для реалізації тунельної передачі.

Після того як сусідство встановлено, LDP відкриває транспортне сполучення між учасниками сеансу поверх TCP/IP. Цим з'єднанням передаються запити на встановлення прив'язки і сама інформація про прив'язку (повідомлення Init). Крім того, учасники сеансу періодично перевіряють працездатність один одного, відправляючи тестові повідомлення (повідомлення KeepAlive).

Розглянемо на прикладі, як відбувається заповнення таблиць міток за протоколом LDP (рис. 9.16).

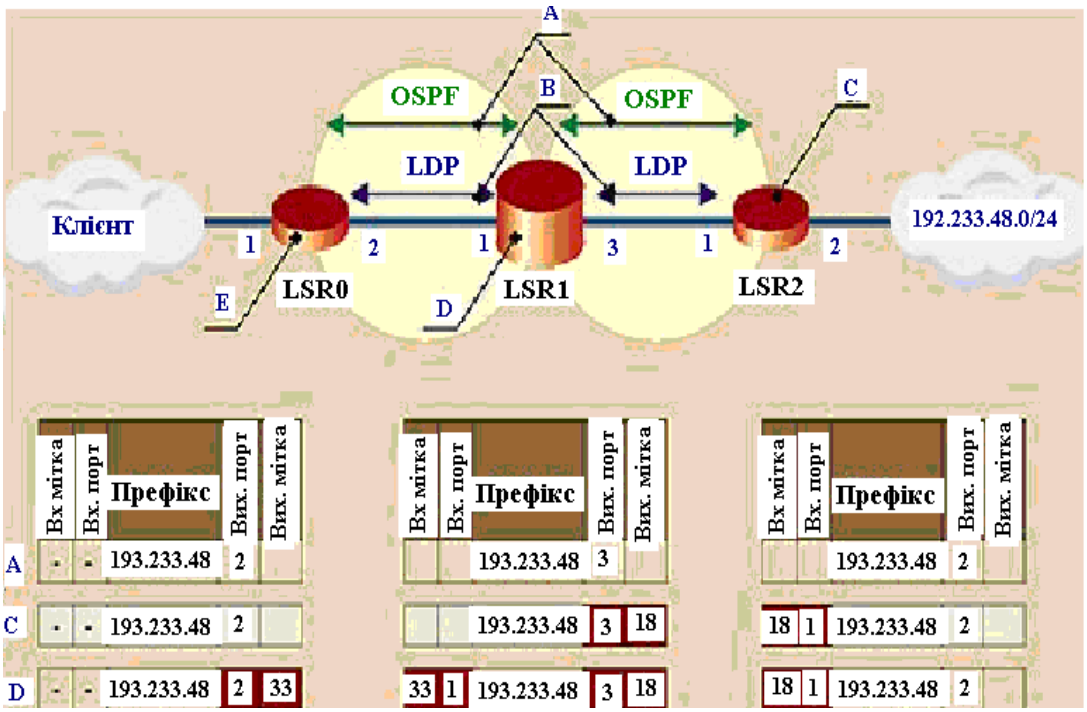

Рисунок 9.16 – Приклад заповнення таблиць міток по протоколу LDP

Припустимо, що обраний упорядкований режим розподілу міток LSP зі спонтанним поширенням відомостей про прив'язку. На стадії A кожен з пристроїв мережі MPLS будує базу топологічної інформації, використовуючи будь-який з сучасних протоколів маршрутизації (на схемі – OSPF). На стадії B маршрутизатори LSR застосовують процедуру знаходження сусідніх пристроїв і встановлюють з ними сеанси LDP. Далі (стадія C) LSR 2 на основі аналізу власних таблиць маршрутизації виявляє, що він є вихідним LSR для шляху, що веде до IP-мережі 193.233.48.0. Тоді LSR 2 асоціює клас FEC з пакетами,

адреса одержувача яких відповідає префіксу даної мережі, і привласнює цього класу випадкове значення мітки – у нашому випадку 18.

Отримавши прив’язку, протокол LDP повідомляє верхній маршрутизатор LSR (LSR 1) про те, що потоку, адресованому мережі з префіксом 193.233.48, присвоєна мітка 18. LSR 1 записує це значення в поле вихідної мітки своєї таблиці. На стадії D пристрій LSR 1, якому відомо значення мітки для потоку, адресованого на префікс 193.233.48, присвоює власне значення мітки даному FEC і повідомляє верхнього сусіда (LSR 0) про цю прив’язку. Тепер LSR 0 записує отриману інформацію в свою таблицю.

Після завершення даного процесу все готово для передачі пакетів з мережі «клієнта» в мережу з адресою 193.233.48.0, тобто по вибраному шляху LSP. Специфікація класу FEC може містити кілька компонентів, кожен з яких визначає набір пакетів, відповідних даному класу.

На сьогоднішній день визначені два компоненти FEC [О: 10, 11, Д: 40, 41]: адреса сайту (адреса вузла) та адресний префікс (префікс мережі). Пакет класифікується як той, що належить до даного класу FEC, якщо адреса одержувача точно збігається з компонентом адреси вузла або має максимальний збіг з адресним префіксом. У нашому прикладі вузол LSR 0 виконує в процесі передачі класифікацію пакетів, що надходять до нього з мережі клієнта, і, якщо адреса одержувача в них збігається з префіксом 193.233.48, присвоївши пакету мітку 33, відправляє його через інтерфейс 2.

9.6. MPLS QoS

Забезпечення гарантованої якості обслуговування

Реалізація функціональності гарантованої якості обслуговування для мереж MPLS забезпечується застосуванням існуючих механізмів QoS для мереж IP у мережах MPLS [О: 10, 11, Д: 38, 39]. Для інтеграції було введено деякі розширення, наприклад використання бітів поля CoS мітки (застаріла назва – біти EXP), що дозволило залишити існуючі фундаментальні алгоритми без змін. У силу тунельної природи технології MPLS механізми QoS

дозволяють передавати інформацію клієнтів з урахуванням класу інформації, встановленого клієнтом, на основі QoS політики оператора.

Використання декількох класів інформаційних потоків дозволяє операторам надавати диференційовані послуги, наприклад, для передачі голосової інформації, чутливої до затримок, може використовуватися один клас, переданий по каналу з мінімальною затримкою. При цьому дані, не чутливі до затримок, можуть передаватися по інших каналах, не заважаючи передачі більш пріоритетної інформації.

Забезпечення функцій QoS – це важливий компонент технології MPLS. В MPLS-мережі QoS-інформація передається в поле CoS заголовка MPLS-мітки. Так само, як й IP QoS, MPLS QoS досягається за допомогою виконання двох головних логічних кроків і використовує такі ж QoS-функції [О: 10, 11, Д: 38, 39].

Основними функціями формування трафіка є його вирівнювання (traffic shaping) та обмеження (traffic policing). Вирівнювання трафіка дозволяє усунути сплески і тим самим зменшити імовірність втрати пакетів даних. Обмеження трафіка полягає у відкиданні пакетів, що не задовольняють заданим параметрам, і здійснюється, наприклад, за допомогою механізму узгодження швидкості доступу (Committed Access Rate, CAR). В основу задач вирівнювання й обмеження трафіка покладені алгоритми дозування трафіка, які мають назви «кошик маркерів» (token bucket) і «діряве відро» (leaky bucket), а також їх різновиди.

Варіант 1. Механізм CAR (Committed Access Rate – узгодження швидкості доступу) обмежує трафік на вхідному маршрутизаторі для всього вхідного в MPLS-мережу IP-трафіка. Він встановлює для трафіка значення IP-пріоритету, виходячи із профілю трафіка й існуючих політик. Значення поля IP-пріоритету пакета копіюється в поле MPLS CoS.

Варіант 2. Механізм CAR обмежує трафік на вхідному маршрутизаторі для всього вхідного в MPLS-мережу IP-трафіка. Він встановлює для трафіка значення поля MPLS CoS виходячи із профілю трафіка й існуючого контракту – SLA (Service Layer Agreement – угода про рівень обслуговування). На відміну від варіанта 1, значення пріоритету в IP-заголовку залишається незмінним.

MPLS використовує функції IP QoS для реалізації різних рівнів обслуговування трафіка, що передається по MPLS-мережі. Єдине розходження полягає в тім, що MPLS QoS базується на CoS-бітах MPLS-мітки, тоді як IP QoS базується на значенні поля пріоритету в заголовку IP-пакета.

Для проведення класифікації трафіка з тим, щоб функції QoS усередині мережі могли реалізувати різні типи обслуговування, можна встановити біти MPLS CoS на границі мережі. Таким чином, для наскрізного надання послуг IP QoS через MPLS-мережу з підтримкою QoS потрібно відобразити або скопіювати значення IP-пріоритету в біти MPLS CoS на границі MPLS-мережі. Значення IP-пріоритету може бути використано й після виходу пакета за межі MPLS-мережі.

9.7. Віртуальні приватні мережі MPLS (MPLS VPN)

VPN третього рівня

VPN третього рівня, або BGP VPN, – це найбільше поширене застосування технологія MPLS [О: 10, 11, Д: 42]. На рис. 9.17 відображена схема мережі MPLS, що забезпечує послуги VPN.

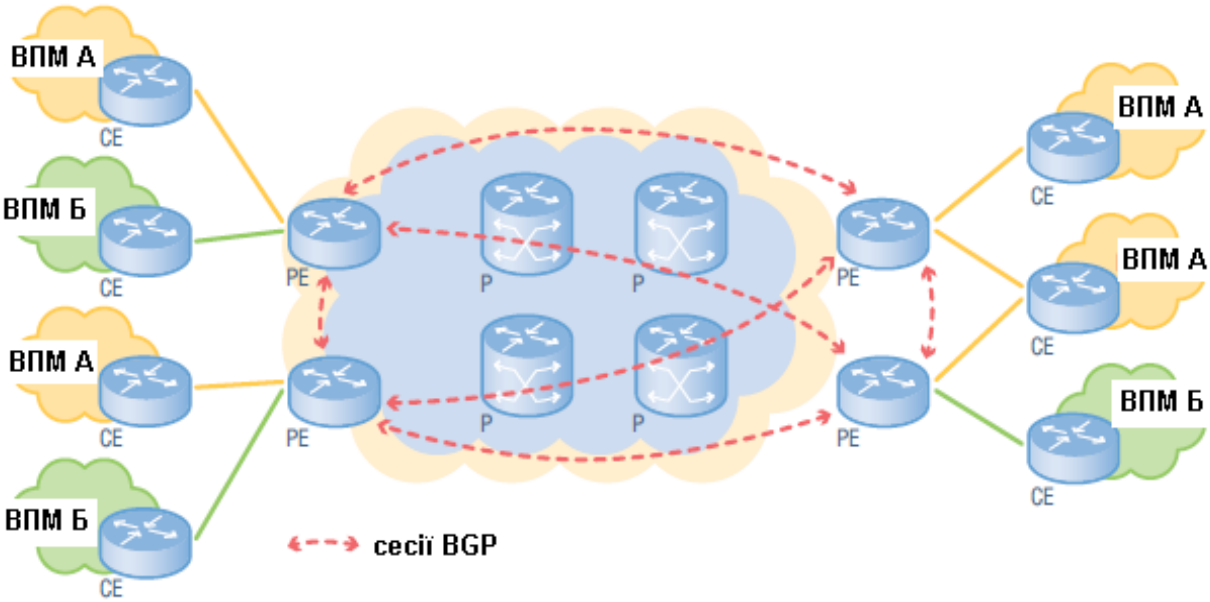

Рисунок 9.17 – Схема мережі MPLS, що забезпечує послуги VPN

Для створення персональної таблиці маршрутизації для кожного клієнта використовуються віртуальні маршрутизатори (Virtual Routing Instance)

і протокол BGP (на PE-маршрутизаторах) для взаємодії прикордонних маршрутизаторів оператора (PE) та передачі міток, пов'язаних з VPN. Оскільки маршрутизатори ядра (P) не мають інформації про VPN, то архітектура, що утворюється, має високу масштабованість.

BGP VPN використовуються тими клієнтами, які потребують передачі інформації на третьому рівні й прагнуть делегувати процеси маршрутизації операторові. Крім того, забезпечується незалежність типів інтерфейсів і протоколів маршрутизації, використовуваних на різних кінцях VPN.

При використанні фільтрації маршрутів легко реалізуються різні топології VPN, у тому числі зіркоподібна й повнозв'язана.

VPN другого рівня

VPN другого рівня дозволяють операторам забезпечувати передачу інформації другого рівня через ядро IP-MPLS [О: 10, 11, Д: 42].

Основні типи послуг:

– транспорт другого рівня через MPLS – прозора передача кадрів другого рівня через ядро IP-MPLS (також відомо як Atom, Any Transport over MPLS);
– послуга віртуального приватного каналу – додає до функціональності Atom передачу сигналізації й автовиявлення CE (Customer Edge) пристроїв;
– послуга віртуальної ЛОМ – надання послуг ЛОМ – через ядро IP-MPLS за допомогою віртуальних комутаторів (Virtual Switch Instance) на PE-маршрутизаторах.

По VPN другого рівня зазвичай передаються кадри на основі Ethernet, ATM, Frame Relay, PPP і HDLC.

Atom і VPN третього рівня засновані на загальній концепції та мають подібні характеристики масштабованості. Для передачі міток VPN використовується сесія LDP. VPN другого і третього рівня доповнюють один одного й можуть співіснувати на одному пристрої.

Якість обслуговування в мережах MPLS VPN

Якість обслуговування – це ключовий компонент служби VPN.

Так само, як і послуги IP QoS, послуги MPLS VPN QoS можуть бути як диференційованими, так і гарантованими. Для забезпечення MPLS VPN QoS можна використати ті ж засоби QoS, що й у протоколі IP [О: 10, 11, Д: 39].

Диференційовані послуги QoS реалізуються за допомогою механізмів CAR (узгодження швидкості доступу), WFQ (зважений алгоритм рівномірного обслуговування черг) й WRED (зважений алгоритм довільного передчасного виявлення), тоді як гарантовані послуги QoS – за допомогою протоколу RSVP.

Диференційовані послуги MPLS VPN QoS

Ця модель QoS має властивості, які, в деяких випадках, схожі на властивості механізму узгодження швидкості передачі інформації (Committed Information Rate – CIR) в Frame Relay-мережі [О: 10, 11, Д: 39].

Кожна VPN-мережа надає функції CAR й CDR (Committed Delivery Rate – погоджена швидкість доставки інформації) на кожному порту, через який вузол одержує доступ до мережі. В Frame Relay-мережі механізм CIR застосовується для вхідного й вихідного трафіка пристрою, підключеного до мережі. Однак в оточенні, не орієнтованому на встановлення з'єднання, яким є IP-мережа, для кожного порту, що підключений до VPN – мережі, існують дві різні швидкості передачі:

– швидкість CAR – для всього вхідного трафіка від мережі в порт доступу до VPN – мережі,
– швидкість CDR – для всього вихідного трафіка від останньої частини VPN – мережі в мережу, підключену через порт доступу.

Трафік, що надходить у порт доступу до MPLS VPN-мережі, **називається погодженим**, якщо швидкість вхідного й вихідного трафіка цього порту нижче обумовленої швидкості CAR й CDR, відповідно. Погоджені пакети доставляються з більшою ймовірністю, чим неузгоджений трафік.

Через те, що в IP-мережі VPN передача даних здійснюється без встановлення з'єднання, пакети можна відправляти між будь-якими двома

сегментами мережі VPN, але при цьому потрібно визначити погоджену швидкість передачі інформації окремо для вхідного й вихідного трафіка.

Для того щоб реалізувати служби CAR й CDR на порту доступу, застосовують функцію обмеження вхідного й вихідного трафіка. Ця функція привласнює більший IP-пріоритет погодженому трафіка й менший IP-пріоритет – неузгодженому.

У VPN – магістралі постачальника послуг диференційовані QoS-функції WFQ й WRED застосовуються для доставки погодженого трафіка з більшою ймовірністю в порівнянні з неузгодженим трафіком.

У табл. 9.3 наведені функції QoS, які застосовуються для трафіка, що передається з одної VPN – хмари в іншу через мережу MPLS постачальника послуг [О: 10, 11, Д: 39]. Постачальник послуг використає механізми CAR й CDR на кожному порту доступу. Постачальник послуг конфігурує мережу таким чином, щоб порти доступу працювали на швидкостях CAR й CDR. Будь-який трафік, що перевищує погоджену швидкість передачі, відкидається з більшою ймовірністю в порівнянні з погодженим трафіком. Погоджені пакети в правильно сконфігурованій мережі MPLS VPN доставляються з високою ймовірністю, порівнянною з надійністю механізму CIR у мережах Frame Relay.

Для забезпечення **гарантованих послуг** QoS потрібне наскрізне використання протоколу RSVP у всіх VPN-хмарах, підключених за допомогою магістралі постачальника послуги, на всьому шляху від джерела до точки призначення [О: 10, 11, Д: 39]. Розміри області й рівень гарантованих послуг QoS залежать від того, яка частина мережі робить явне резервування ресурсів за допомогою повідомлень RSVP PATH. RSVP-резервування відбувається тільки у вузлах VPN-мережі. Постачальник послуг VPN просто передає всі RSVP-пакети так, якби це були звичайні IP-пакети з даними. Це дозволяє клієнту управляти розподілом ресурсів у рамках клієнтської мережі, але ніяк не впливає на обробку клієнтського трафіка в мережі постачальника послуг.

Таблиця 9.3 – Реалізація функцій QoS в MPLS VPN

Крок	Місце застосування	Функції QoS
1	Маршрутизатор на вході в мережу MPLS VPN	Варіант 1. Механізм CAR обмежує трафік, що приходить від прикордонного маршрутизатора клієнта, у прикордонному маршрутизаторі постачальника послуг. Трафіку привласнюється значення IP-пріоритету, виходячи з його профілю й контракту. Значення IP-пріоритету в заголовку IP-пакета копіюється в поле MPLS CoS. Варіант 2. Механізм CAR обмежує трафік, що приходить від прикордонного маршрутизатора клієнта, у прикордонному маршрутизаторі постачальника послуг. Трафіку привласнюється значення MPLS CoS, виходячи з його профілю й контракту.
2	Магістраль постачальника послуг	Диференціація трафіка на основі поля MPLS CoS за допомогою функцій IP QoS WFQ й WRED
3	Маршрутизатор на виході з мережі MPLS VPN	Поле CoS з MPLS мітки копіюється в поле IP-пріоритету в заголовку IP-пакета
4	Маршрутизатор на виході з мережі MPLS VPN	Механізм CAR обмежує вихідний трафік на інтерфейсі прикордонного маршрутизатора постачальника послуг, підключеному до кінцевого маршрутизатора клієнта, на основі функції CDR

На вході в мережу постачальника послуг VPN-трафік з гарантованим обслуговуванням відзначається високим значенням поля MPLS CoS. Цей трафік буде доставлятися по мережі постачальника послуг MPLS VPN з великим ступенем імовірності, завдяки використанню таких функцій IP QoS, як WFQ та WRED. Постачальник послуг MPLS VPN передає всі RSVP-пакети як звичайні IP-пакети з даними.

Таким чином, трафік, для якого були зарезервовані ресурси, одержує більш високий рівень обслуговування без участі постачальника послуг, якому у цьому випадку не потрібно робити резервування ресурсів окремо для кожного клієнта своєї мережі.

Тунелі з гарантованою смугою пропускання (guaranteed bandwidth – GB) використовуються для передачі трафіка, що вимагає резервування ресурсів у магістралі постачальника послуг [О:10, 11, Д: 39]. Основним принципом при створенні GB-тунелю є помітка частини ваги черги гарантованої смуги пропускання як зайнятої. RSVP – резервування відбувається як наскрізне

від джерела до точки призначення через мережу постачальника послуг VPN. Усередині мережі постачальника послуг RSVP-повідомлення передаються по GB-тунелю, як звичайні пакети з даними.

9.8. Керування трафіком у мережах MPLS

Механізм керування трафіком (Traffic Engineering – TE) в технології MPLS надає можливість встановлювати явний шлях, по якому будуть передаватися потоки даних [О: 10, 11, Д: 43]. Стандартно, IP-трафік маршрутизується за допомогою його передачі від однієї точки призначення до іншої і прямує до пункту призначення по шляху, що має найменшу сумарну метрику мережного рівня. Шлях, по якому прямує IP-трафік, може не бути оптимальним, тому що він залежить від інформації про статичну метрику маршруту. При виборі шляху не враховуються вільні мережні ресурси, а також вимоги до обслуговування трафіка.

Механізм TE визначає шлях, що комутується за допомогою міток (Label Switched Path – LSP), для передачі трафіка по явно заданому шляху, який може відрізнятися від звичайного шляху, заснованого на адресі пункту призначення і визначеного стандартними протоколами маршрутизації [О: 10, 11, Д: 43]. Як сигнальний протокол для встановлення шляху LSP використовується протокол резервування ресурсів (Resource Reservation Protocol – RSVP) TE-модифікації (TE-RSVP).

Найбільшою проблемою для постачальника послуг при забезпеченні наскрізної якості обслуговування є нездатність визначити точний шлях, по якому будуть передаватися IP-пакети. Маршрутизатор не може заздалегідь визначити шлях, по якому піде IP-трафік, що є результатом поетапного ухвалення рішення про маршрут. Подібна властивість IP-маршрутизації обумовлена тим, що IP є протоколом без орієнтації на встановлення з’єднання.

При маршрутизації на основі резервування ресурсів (Routing by Resource Reservation – RRR) для забезпечення механізму IP TE використовуються тунелі, побудовані на базі LSP-шляху, для керування трафіком, і мітки – для передачі

даних по явному шляху, що відрізняється від шляху, визначеного стандартною IP-маршрутизацією [О: 10, 11, Д: 43].

На сьогоднішній день рішення, засноване на використанні механізму MPLS TE, обмежено одним доменом маршрутизації (автономною системою). Для протоколів внутрішнього шлюзу (Interior Gateway Protocol – IGP) необхідні розширення, що дозволяють їм підтримувати механізм TE. На даний момент тільки відкритий протокол вибору найкоротшого шляху (Open Shortest Path First – OSPF) і протокол обміну інформацією про маршрути між проміжними системами (Intermediate System-to-Intermediate System – IS-IS) підтримують TE-розширення [О: 10, 11, Д: 44, 45].

9.9. Маршрутизація на основі резервування ресурсів

Такі протоколи маршрутизації, як OSPF та IS-IS, визначають маршрути передачі IP-пакетів на підставі інформації про топологію мережі й метрики маршрутів. При реалізації маршрутизації на основі резервування ресурсів (Routing by Resource Reservation – RRR) додатково враховуються такі параметри, як клас трафіку, вимоги до обслуговування й наявність доступних мережних ресурсів [О: 10, 11, Д: 44, 45].

Механізм RRR TE припускає визначення транку трафіку, вимог до ресурсів, політики тунелю, а також способу обчислення або специфікації явного шляху по якому буде прокладатися TE-тунель. Операційна модель роботи механізму RRR [О: 10, 11, Д: 44, 45] показана на рис. 9.18.

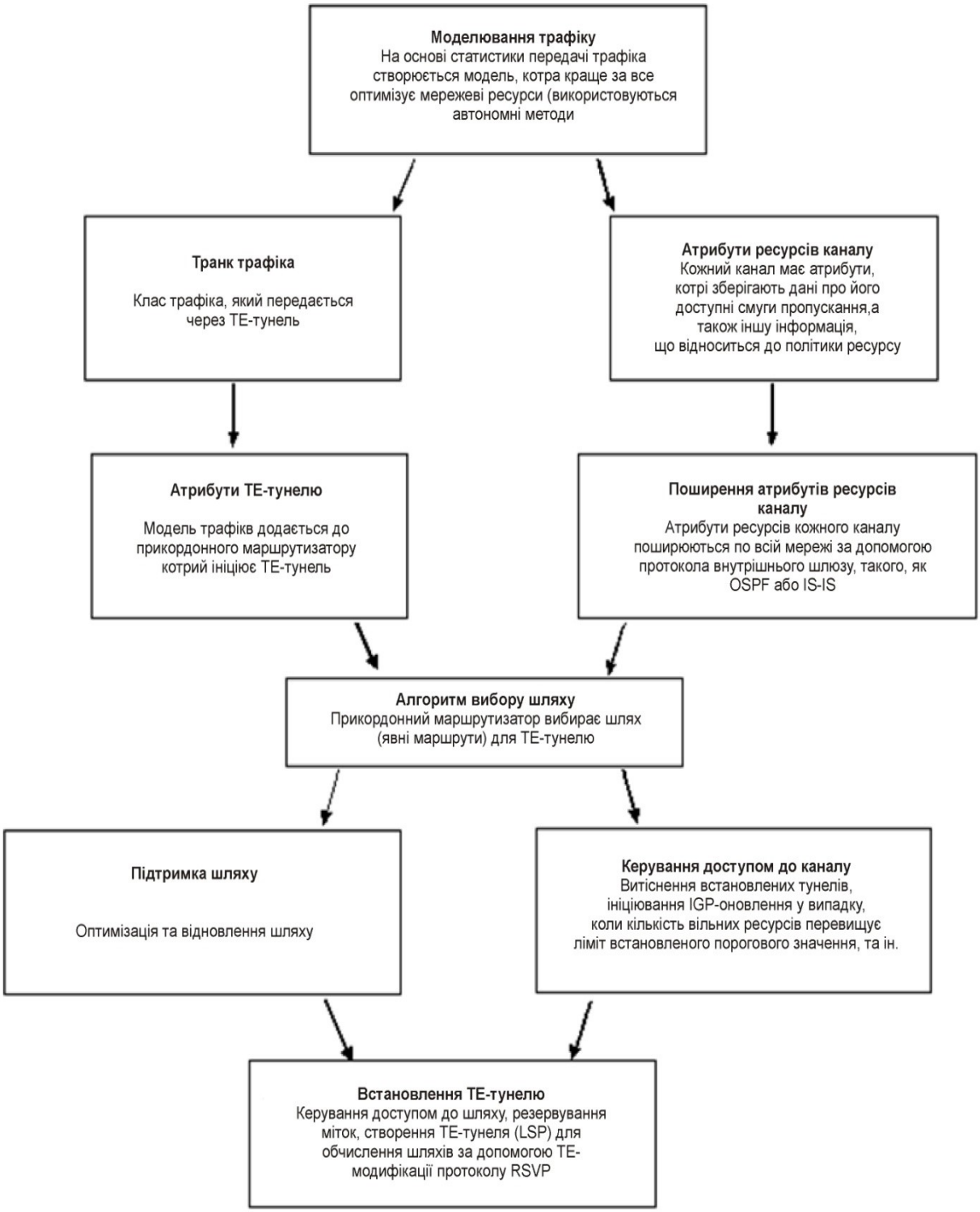

Рисунок 9.18 – Модель маршрутизації на основі резервування ресурсів

Перед впровадженням механізму ТЕ в мережі дуже важливо створити профілі шаблонів потоків трафіка й зібрати статистику в декількох точках мережі. В результаті необхідно отримати модель трафіка з максимальним використанням ресурсів мережі.

ТЕ-транк (TE trunk) визначає клас пакетів, що передаються по ТЕ-тунелю. Це локальна політика для головного маршрутизатора, що ініціює створення ТЕ-тунелю. Клас трафіку визначається по заголовках пакетів протоколу IP. Клас трафіку може ґрунтуватися на одному параметрі, наприклад на IP-адресі точки призначення, або MPLS-полі класу обслуговування (Class of Service – CoS), а також на декількох параметрах. Прикладом може служити клас HTTP-трафіка, що передається від заданого відправника до заданого одержувача.

Відповідно до механізму RRR всі пакети трафіку одного класу передаються через мережу по спеціально зазначеному або динамічно створеному певному шляху. Саме тому класи IP трафіку також називають транком трафіку (traffic trunks). ТЕ-тунель має певні атрибути, що описують вимоги транку трафіку, від яких залежить вибір адміністративних політик. Нижче наведені основні атрибути, які використовуються для формування вимог.

Атрибут смуги пропускання (bandwidth) визначає наскрізну смугу пропускання необхідну для ТЕ-тунелю. Її вибір заснований на вимогах класу трафіка, що передається по ТЕ-тунелю.

Пріоритети встановлення й утримання використовуються при керуванні доступом. Пріоритет встановлення визначає пріоритет ресурсу. Коли ресурси перебувають у стані конфлікту (contention), новий тунель із високим пріоритетом одержання ресурсу може позачергово зайняти вільний ресурс, витісняючи при цьому, всі встановлені тунелі, пріоритет утримання яких менше пріоритету одержання ресурсу нового тунелю. Встановлений ТЕ-тунель із найвищим пріоритетом утримання не може бути витиснутий.

Атрибут спорідненості класу ресурсу надає можливість обмежувати вибір шляху адміністративним включенням у мережу певних каналів або виключенням їх з мережі. Атрибут спорідненості класу ресурсу складається з 32-бітового атрибута спорідненості ресурсу й 32-бітової маски класу ресурсу. Атрибут спорідненості ресурсу (resource affinity) вказує на необхідність включення в процес розрахунку шляхи певного каналу або його виключення

із процесу. Кожен канал має атрибут класу ресурсу, рівний за замовчуванням 0x00000000 (якщо він не визначений явно). Маска класу ресурсу використовується для виділення потрібного набору бітів атрибута класу ресурсу каналу.

Клас ресурсу, маска класу ресурсу й атрибут спорідненості класу ресурсу співвідносяться один з одним у такий спосіб:

Клас ресурсу & маска класу ресурсу == спорідненість класу ресурсу,

де «&» позначає побітову операцію логічного «Та», а «==» – побітова логічна рівність.

Атрибут порядку вибору шляхи (path selection order) визначає порядок, у якому прикордонний маршрутизатор вибирає явні шляхи для TE-тунелів. Явний шлях (explicit path) – це вихідний маршрут, що представляє собою послідовність IP-адрес, задану адміністративним способом або динамічно обчислену на основі найкоротшого шляху з урахуванням різних обмежень. В атрибуті порядку вибору шляху вказується порядок, у якому адміністративно заданий або динамічно отриманий шлях використовується для встановлення TE-тунелю.

В **атрибуті адаптуємості** (adaptability) визначається, чи потрібно повторно оптимізувати існуючий TE-тунель у випадку, якщо з'явився кращий шлях.

В **атрибуті стійкості** (resilience) визначається бажане поводження системи у випадку припинення існування поточного шляху TE-тунелю. Це зазвичай відбувається через збої в мережі або витісненні тунелю.

Важливою вимогою для введення в дію механізму керування трафіком є поширення локальної інформації про ресурси каналів по RRR-мережі. Атрибути ресурсів каналу лавиноподібно поширюються по мережі за допомогою розширень таких протоколів маршрутизації стану каналу, як OSPF й IS-IS.

Існуючі протоколи маршрутизації внутрішнього шлюзу лавиноподібно поширюють інформацію про метрику й топологію мережі по всіх каналах, для того, щоб забезпечити обчислення найкоротшого шляху до точки призначення. В RRR-мережі використовується розширений протокол маршрутизації, який разом з інформацією про метрику поширює по кожному каналу атрибути ресурсів. Лавиноподібне поширення атрибутів ресурсів відбувається незалежно від поширення інформації про метрику й використовується тільки тоді, коли змінюється стан каналу або клас ресурсу каналу, а також коли ширина вільної смуги пропускання перевищила заздалегідь сконфігуроване граничне значення.

ТЕ-шлях повинен враховувати як обмеження каналу, так і вимоги ТЕ-тунелю. **При обчисленні ТЕ-шляху проводяться такі дії**:

Крок 1. Розглядаються наступні атрибути й інформація:

– атрибути ТЕ-тунелю, встановлені головним маршрутизатором, що створює цей тунель;
– атрибути ресурсів каналів всієї мережі, які поширюються за допомогою протоколу внутрішнього шлюзу;
– інформація про топологію мережі, що поширюється за допомогою протоколу внутрішнього шлюзу;
– якщо виконується повторна оптимізація ТЕ-тунелю, то розглядається його поточний шлях.

Крок 2. Видаляються канали з недостатньою пропускною здатністю або невідповідні політиці ресурсів.

Крок 3. Після видалення всіх нерозглянутих каналів, запускається окрема копія алгоритму вибору найкоротшого шляху (Shortest Path First – SPF) для обчислення найкоротшого (з найменшою метрикою) шляху з бази станів каналів протоколу внутрішнього шлюзу.

Копія алгоритму SPF запускається спеціально для обчислення ТЕ-шляху і вона відрізняється від копії алгоритму, що використовується маршрутизатором для побудови своєї таблиці маршрутизації на основі бази станів каналів. Явний шлях для ТЕ-тунелю обчислюється за допомогою алгоритму SPF. Обчислений

явний шлях представляється як послідовність, що складається з IP-адрес маршрутизаторів. Після запиту на встановлення ТЕ-тунелю, останній створюється з використанням явного шляху на основі атрибута порядку вибору.

Такий шлях носить назву ER-LSP (Explicitly Routed Label Switched Path) – LSP з явно заданим маршрутом [О: 10, 11].

9.10. Сигнальний протокол TE-RSVP

Для сигналізації ТЕ-тунелю використовується протокол TE-RSVP. Протокол TE-RSVP – це розширення існуючого протоколу RSVP з підтримкою сигналізації LSP-шляху [О: 10, 11, Д: 46, 47]. Протокол TE-RSVP використовує сигнальні повідомлення протоколу RSVP, додаючи певні розширення для підтримки механізму керування трафіком.

Нижче перераховані деякі важливі розширення протоколу TE-RSVP [О: 10, 11, Д: 46, 47].

Підтримка резервування міток. Для того щоб використовувати RSVP при сигналізації LSP-тунелю, цей протокол повинен підтримувати резервування й встановлення міток. На відміну від звичайних RSVP-потоків, протокол TE-RSVP використовує RSVP з метою резервування міток для потоків, які не вимагають резервування смуги пропущення. Для цього був визначений новий тип об’єкта FlowSpec. Крім того, протокол TE-RSVP резервує також мітки для потоків, які не вимагають резервування смуги пропущення.

Підтримка маршрутизації від джерела. В LSP-тунелях використовується явна маршрутизація від джерела. Явна маршрутизація від джерела реалізована в протоколі RSVP шляхом додавання нового об’єкта – SRO.

Підтримка RSVP-вузлів. На відміну від протоколу RSVP, у якому повідомлення RSVP PATH й RSVP RESV створюються прикладними програмами на кінцевих вузлах, у протоколі TE-RSVP повідомлення RSVP PATH й RSVP RESV ініціюються головними маршрутизаторами. Отже, протокол TE-RSVP вимагає підтримки маршрутизаторами емуляції RSVP-вузлів.

Підтримка ідентифікації тунелю, створеного на базі ER-LSP-шляху. Для передачі ідентифікатора тунелю використовуються нові типи об'єктів Filter_Spec й Sender_Template. Об'єкту Session також дозволяється переносити номер IP-протоколу «null», тому що по LSP-тунелю, найімовірніше, будуть передаватися IP-пакети з багатьма іншими номерами протоколу.

Підтримка нового алгоритму виділення резервування. Додано нове RSVP-повідомлення, RESV TearConfirm. Це повідомлення додається для гарантованого виділення TE-тунелю.

До протоколу TE-RSVP додані дві важливі функціональні можливості, які дозволяють йому створювати LSP-шлях, що явно маршрутизується (Explicitly Routed Label Switched Path – ER-LSP) [О: 10, 11, Д: 46, 47]:

– метод прив'язки міток до RSVP-потоків,
– функція явної маршрутизації RSVP-повідомлень.

У TE-тунелі, побудованому на основі ER-LSP-шляху, єдиним відправником є перший вузол в LSP-шляху, а єдиним одержувачем – останній вузол. Всі проміжні вузли в LSP-шляху роблять звичайну комутацію міток на основі вхідної мітки. Перший вузол в LSP-шляху ініціює створення ER-LSP-шляху. Цей вузол іноді називають головним маршрутизатором. Головний маршрутизатор ініціює встановлення TE-тунелю, відправляючи повідомлення RSVP PATH на IP-адресу точки призначення тунелю з об'єктом SRO (Source Route Object – об'єкт вихідного маршруту), що визначає явний маршрут. В об'єкті SRO перебуває список IP-адрес і покажчиків, що визначають наступну точку призначення в списку.

Всі вузли в мережі передають PATH-повідомлення за адресою наступної точки призначення, зазначеною в об'єкті SRO. Крім цього, вони додають об'єкт SRO у свій блок стану шляхи.

Коли пристрій у точці призначення одержує PATH-повідомлення, він визначає, що йому необхідно встановити ER-LSP-шлях на основі об'єкта запиту мітки (Label Request Object – LRO), що перебуває в PATH-повідомленні, й генерує RSVP-повідомлення запиту на резервування ресурсів для сеансу

(RESV). У повідомленні RSVP RESV, що передається відправникові, пристрій розміщає створену їм мітку й відправляє її як об’єкт LABEL.

Вузол, що одержав RESV-повідомлення, використовує мітку для відправлення всього трафіку цим шляхом. Він також створює нову мітку й відправляє її як об’єкт LABEL у повідомленні RSVP RESV наступному вузлу в напрямку до відправника. Саме цю мітку вузол розраховує одержати разом з усім трафіком, що надходить цим шляхом.

Повідомлення RSVP RESV прямує по шляху, прямо протилежному шляху, по якому йшло повідомлення RSVP PATH. Це пов’язане з тим, що об’єкт SRO у блоці стану шляху (установлений у кожному вузлі PATH-повідомленням) визначає спосіб передачі повідомлення RSVP RESV нагору по напрямку руху потоку трафіка.

Після того як шлях ER-LSP сформований, встановлюється TE-тунель. Якщо трафік доставляється без гарантій, то резервування ресурсів робити необов’язково. Для того щоб розподілити ресурси на ER-LSP-шляху, використовуються звичайні RSVP-об’єкти Tspec й Rspec. Tspec – об’єкт відправника, що знаходиться в PATH-повідомленні, використовується для визначення трафіку, що передається цим шляхом. Точка призначення повідомлення RSVP PATH використовує цю інформацію для створення відповідних об’єктів Tspec та Rspec одержувача, які використовуються при розподілі ресурсів на кожному вузлі ER-LSP-шляху.

Система керування доступом до каналу вирішує, якому з TE-тунелів варто надати ресурси. Вона припускає наявність наступних функцій:

– визначення наявності ресурсів. Система визначає, чи є вільні ресурси;
– знищення тунелю. Система повинна знищити існуючі тунелі, коли нові тунелі з високим пріоритетом утримання ресурсів витісняють існуючі тунелі з низьким пріоритетом утримання ресурсів;
– ведення локального обліку використання ресурсів. Система стежить за використанням ресурсів;
– ініціювання відновлення інформації про маршрути протоколу внутрішнього шлюзу.

Система ініціює процес лавиноподібного поширення відновленої інформації про маршрути протоколу внутрішнього шлюзу, коли зі статистики локального обліку використання ресурсів походить, що кількість доступних ресурсів може опуститися нижче сконфігурованого граничного значення.

Підтримка ТЕ-шляху включає виконання таких функцій, як повторна оптимізація шляху і його відновлення. Ці операції виконуються після встановлення тунелю.

Повторна оптимізація шляху описує бажане поводження маршрутизатора у випадку появи, після встановлення ТЕ-шляху, його потенційно кращого спадкоємця. Під час повторної оптимізації шляху маршрутизатор повинен спробувати знайти яку-небудь можливість для того, щоб заново оптимізувати існуючий ТЕ-шлях. Необхідність застосування функції повторної оптимізації шляху вказується в атрибуті адаптуємості.

Відновлення шляху (path restoration) описує метод відновлення ТЕ-тунелю у випадку, якщо поточний шлях перестав функціонувати, використовуючи для цього атрибут стійкості.

9.11. Розширення протоколу маршрутизації внутрішнього шлюзу

Протоколи маршрутизації внутрішнього шлюзу OSPF й IS-IS були розширені для поширення атрибутів ресурсів каналу разом з інформацією про IP-досяжність мережного рівня [Д: 44, 45]. Для передачі інформації про ресурси каналу протокол IS-IS використовує новий пакет, що складається з поля типу (Type), довжини (Length) і значення (Value), який відомий як пакет TVL, а протокол OSPF – непрозоре повідомлення про стан каналу (Opaque Link State Advertisement – LSA). Слід зазначити, що існуюча функціональність протоколів OSPF й IS-IS залишилась колишньої, тільки тепер ці протоколи відправляють інформацію про поточні обмеження щораз, коли їм необхідно поширити інформацію про IP-досяжність.

Механізм керування доступом до ТЕ-каналу маршрутизатора вимагає повторного поширення інформації про обмеження щоразу, коли він помічає

істотні зміни в інформації про доступні ресурси, ґрунтуючись на сконфігурованих граничних значеннях.

Пакет TLV, що визначає досяжність проміжних систем (Intermediate System – IS), розширений для передачі інформації про ресурси каналу. Розширений кортеж TLV називається пакетом TLV типу 22. Усередині цього пакета використовуються різні підпакети TLV, призначені для передачі атрибутів каналу з метою підтримки механізму керування трафіком.

Оскільки базове LSA-оголошення маршрутизатора протоколу OSPF є нерозширюваним, для підтримки механізму керування трафіком використається непрозоре LSA-оголошення. Існує три типи непрозорого LSA-оголошення, кожне з яких характеризується своєю власною областю лавиноподібного поширення. OSPF-розширення для забезпечення підтримки механізму TE використовують тільки LSA-оголошення типу 10, поширення яких обмежене однією областю.

Новий тип непрозорого LSA-оголошення (тип 10), що забезпечує підтримку механізму TE, називається оголошенням TE LSA. У цьому LSA-оголошенні описуються маршрутизатори й двоточкові канали (так само, як в оголошенні LSA маршрутизатора). Для опису каналів із множинним доступом вистачає існуючого оголошення LSA мережі, а тому для цієї мети не було визначено ніяких додаткових LSA-оголошень. Слід зазначити, що обмеження механізму TE до перемикання між декількома маршрутами може не виправдати себе в тому випадку, якщо інший трафік у мережі буде передаватися без участі механізму TE.

Запитання для самоперевірки та контролю засвоєння знань

1. Який підхід до комутації покладено в основу технології MPLS?
2. За рахунок чого в технології MPLS об’єднана техніка віртуальних каналів з функціональністю стеку TCP/IP?
3. В чому полягає мультипротокольність технології MPLS?
4. Наведіть основні функції технології MPLS?
5. За рахунок чого досягається ефективність роботи технології MPLS?

6. Які функціональні рівні виділяють в технології MPLS?

7. Яку функцію виконує рівень керування?

8. Яку функцію виконує рівень комутації?

9. Яку функцію виконує керуючий компонент?

10. Яку функцію виконує передавальний компонент?

11. Які пристрої виділяють в структурі MPLS-мережі?

12. Які функції виконує LSR?

13. Які функції виконує LER?

14. Яку назву мають пограничні маршрутизатори MPLS-мережі в термінології віртуальних приватних мереж?

15. Яку назву мають магістральні маршрутизатори MPLS-мережі в термінології віртуальних приватних мереж?

16. За допомогою якого пристрою мережі клієнта підключається до мережі провайдера в термінології віртуальних приватних мереж?

17. Що розуміють під класом еквівалентності в технології MPLS?

18. Чому при передачі пакетів через MPLS-мережу немає необхідності аналізувати заголовок 3-го рівня пакету?

19. На підставі чого MPLS-маршрутизатор приймає рішення, на який порт передати пакет?

20. Що розуміють під стеком міток в технології MPLS?

21. Що характерно для архітектури cell-mode MPLS?

22. Що характерно для архітектури frame-mode MPLS?

23. З яких полів складається мітка?

24. Для чого використовується поле класу трафіку?

25. На що вказує прапор дна стеку?

26. Для чого використовується поле TTL?

27. На яку частину маршруту передачі пакету розповсюджується значення мітки?

28. Яка інформація міститься у базі LFIB?

29. За допомогою якого протоколу маршрутизатори обмінюються інформацією про мітки?

30. Яким чином виконується передача пакетів через MPLS-мережу?

31. Для яких цілей використовують стек міток?

31. Яка мітка стеку використовується при комутації пакетів?

32. Для чого використовуються MPLS-тунелі?

33. Що розуміють під прив’язкою мітки?

34. Які методи поширення міток можуть бути використані в технології MPLS?

35. Який протокол використовується в технології MPLS для поширення інформації про мітки?

36. Що характерно для спадного призначення міток?

37. Який маршрутизатор першим присвоює мітку для конкретного FEC при спадному призначенні міток?

38. Яка мітка при спадному призначенні міток присвоюється маршрутизатором локально, а яка надходить від іншого маршрутизатору?

39. Що характерно для спадного призначення міток на вимогу?

40. Що характерно для висхідного призначення міток?

41. Який маршрутизатор першим присвоює мітку для конкретного FEC при висхідному призначенні міток?

42. Яка мітка при висхідному призначенні міток присвоюється маршрутизатором локально, а яка надходить від іншого маршрутизатору?

43. Які фази виділяють при встановленні сусідських відносин по протоколу LDP?

44. Як посилаються повідомлення Hello в протоколі LDP?

45. Що розуміють під транспортною адресою в протоколі LDP?

46. Який сценарій встановлення LDP-сесії?

47. Для чого використовується повідомлення Init в протоколі LDP?

48. Для чого використовується повідомлення KeepAlive в протоколі LDP?

49. Які виділяють режимі функціонування протоколу LDP?

50. Для чого використовується режим обміну інформацією про мітки?

51. Які існують механізми контролю над розповсюдженням міток?

52. Що характерно для механізму незалежного контролю?

53. Що характерно для механізму упорядкованого контролю?

54. Які існують режими збереження міток?

55. Що характерно для стриманого режиму збереження міток?

56. Що характерно для вільного режиму збереження міток?

57. Для чого використовується механізм запобігання циклів в протоколі LDP?

58. Який тип повідомлення LDP використовується для ідентифікації сусідів?

59. Який тип повідомлення LDP використовується для встановлення LDP-сесії?

60. Який тип повідомлення LDP використовується для підтримки активного статусу LDP-сесії?

61. Який тип повідомлення LDP використовується для оповіщення сусідів про появу нових IP-мереж?

62. Який тип повідомлення LDP містить FEC і мітку?

63. Який тип повідомлення LDP використовується для запиту мітки?

64. Які параметри можуть бути використані для визначення FEC?

65. Яке поле мітки використовується для реалізації механізмів якості обслуговування у технології MPLS?

66. Які функції для формування трафіку реалізують у технології MPLS для забезпечення якості обслуговування?

67. Для чого використовують функцію вирівнювання трафіку?

68. Для чого використовують функцію обмеження трафіку?

69. Чим відрізняються два варіанти обмеження трафіку на вході в MPLS-мережу?

70. Який підхід до класифікації трафіку і встановлення бітів поля CoS використовується в технології MPLS?

71. За допомогою яких механізмів реалізують диференційовані послуги MPLS VPN QoS?

72. За допомогою яких механізмів реалізують інтегровані послуги MPLS VPN QoS?

73. Що розуміють під швидкістю CAR?

74. Що розуміють під швидкістю CDR?

75. Який трафік називають погодженим?

76. За допомогою якої функції реалізують служби CAR та CDR на порту?

10. ІНСТРУМЕНТИ КЛАСИФІКАЦІЇ ТА МАРКУВАННЯ ПАКЕТІВ

10.1. Порівняльна характеристика базових архітектурних моделей QoS

У таблиці10.1 наведена коротка порівняльна характеристика базових архітектурних моделей QoS у мережах IP [О: 5, 6], які буди розглянуті у попередніх розділах.

Таблиця 10.1 – Порівняльна характеристика базових архітектурних моделей QoS

	best effort	**DiffServ**	**IntServ**
Наданий рівень якості обслугову-вання	Негарантована доставка	Рівень диференційованих послуг (Soft QoS), не забезпечує гарантій наданих послуг	Рівень інтегрованих послуг (Hard QoS), дає тверді гарантії щодо забезпечення необхідного обсягу ресурсів
Область застосування	Без обмежень	Магістральні мережі	Орієнтується на периферію мережі, LAN, MAN
Функція QoS мережного рівня	Зв’язність вузлів мережі, рекомендується застосовувати механізми запобігання перевантажень (WRED)	Механізми CB-Marking, Class-Based Shaping, CAR, WRED, WFQ, CBWFQ, LLQ. У мережах MPLS – CoS	Протокол RSVP
Функція QoS канального рівня	Обслуговування з невизначеною бітовою швидкістю UBR у мережах ATM, механізм узгодження швидкості передачі інформації (CIR) у мережах Frame Relay	IEEE 802.1p	Диспетчер пропускної здатності підмережі SBM, механізм обслуговування з постійною бітовою швидкістю CBR у мережах ATM, механізм узгодження швидкості передачі інформації (CIR) у мережах Frame Relay
Переваги	Простота, не вимагає реалізації складних QoS-механізмів	Висока масштабованість; можлива організація великої кількості класів обслуговування (при використанні шестибітного поля DSCP – 64 класи)	Забезпечення твердих гарантій щодо наданого рівня якості послуг; реалізація управління доступом з метою перевірки доступних ресурсів; інтеграція з інфраструктурою правил COPS; підтримка сигналізації H.323

Продовження табл. 10.1

	best effort	DiffServ	IntServ
Недоліки	Відсутність усіляких гарантій щодо якості доставки і щодо самого факту доставки, відсутність диференціації трафіка на класи (рівні) обслуговування. Мінімальна гарантована смуга пропускання, залежить від кількості інших класів і їхніх вимог	Відсутність твердих гарантій; вимагає реалізації великої кількості механізмів QoS, можливі складності в налаштуванні	Низька масштабованість, що пов'язано з обробкою кожного потоку окремо, необхідністю зберігати інформацію про стан кожного потоку на кожному вузлі, великими обсягами сигнальної інформації

10.2. Методика застосування моделей QoS до мережного трафіку

Для будь-якого правила QoS потрібно, насамперед, визначити трафік, що має особливі вимоги. Інструмент класифікації позначає пакет або кадр певним значенням. Ці значення міток дозволяють розмежувати різні типи трафіка та застосувати до них різні правила обробки черг (CBWFQ, MDRR тощо). Класифікація й маркування проводиться на основі аналізу наступних параметрів [О: 5, 6, Д: 48, 49]:

- параметри рівня 2 (біти класу послуг CoS 802.1Q/p, значення експериментальних біт MPLS);
- параметри рівня 3 (біти IP Precedence (IPP), кодові точки диференційованих послуг (DSCP Code-Points), IP-адреси джерела й пункту призначення);
- параметри рівня 4 (порти TCP або UDP);
- параметри рівня 7 (підписи прикладних програм).

Тільки після повної ідентифікації трафіка до нього можна застосовувати QoS правила (policies). Рекомендується ідентифікувати й позначати трафік (за допомогою DSCP) якнайближче до джерела. Периферійна частина мережі, де відбувається прийом (або відхилення) класифікованих пакетів, називається «границею довіри» (trust boundary). Якщо мітки проставлені правильно, то проміжним ділянкам мережі не доводиться повторно ідентифікувати трафік.

На цих ділянках просто виконуються певні правила QoS, які були поставлені у відповідність кодам DSCP раніше. Такий підхід спрощує мережне керування й скорочує навантаження на процесори [О: 5, 6, Д: 48, 49].

Класифікація повинна виконуватися на мережній периферії (наприклад, в IP-телефонах або на інших терміналах голосового зв'язку). Однак не рекомендується довіряти маркуванню, зробленому прикладними програмами на персональних комп'ютерах, тому що можливе зловживання з боку користувачів. Класифікація здійснюється за допомогою списків доступу (ACL), DSCP або MPLS EXP [О: 5, 6, Д: 48, 49].

Для маркування трафіку можуть використовуватися наступні механізми [О: 5, 6, Д: 48, 49].

802.1Q/p класи послуг (CoS). Ethernet кадри можуть позначатися за допомогою біт у заголовку другого рівня з використанням 802.1p біт пріоритету в заголовку 802.1Q. Розмір поля 802.1p – 3 біти, у такий спосіб тільки вісім класів сервісу (0-7) доступні для маркування Ethernet кадрів другого рівня.

Байт типу сервісу заголовку IP-пакету – Type of Service (ToS). У зв'язку з тим, що в процесі проходження пакетів від джерела до призначення часто міняється середовище другого рівня, більш універсальний метод маркування полягає у використанні заголовка третього рівня. Другий байт заголовка IPv4 пакета – це байт TOS. Перші три біти байту TOS називаються бітами IPP. IPP, також як й CoS біти 802.1p, дозволяють позначити пакет тільки вісьма значеннями (0-7). Зазвичай IPP біти встановлюються наступним чином [О: 5, 6, Д: 48, 49]:

- IPP значення 6 й 7 резервуються для мережного керуючого трафіка (наприклад, протоколів маршрутизації);
- IPP значення 5 рекомендоване для голосового трафіка;
- IPP значення 4 використовується спільно трафіком відеоконференцій і потокового відео;
- IPP значення 3 призначене для сигналізації викликів;
- IPP значення 1 й 2 можуть використатися для даних прикладних програм;

– IPP значення 0 – маркування за замовчуванням.

Враховуючи кількість біт IPP (3 біти), IPP маркування є досить обмеженим й не дозволяє визначати велику кількість класів сервісу. Це обмеження знімається при застосуванні 6-бітної моделі маркування DSCP (64 значення).

DSCP-значення. DSCP-значення можуть бути виражені в цифровій формі або з використанням спеціальних ключових слів. Визначено три класи DSCP маркування: по можливості (BE – best effort або DSCP 0), гарантована доставка (Assured Forwarding, AFxy) і термінова доставка (Expedited Forwarding – EF). На додаток до цих трьох визначених класів існують коди селектора класів (class selector code points), які зворотно сумісні з IPP (CS1-CS7 ідентичні значенням 1-7 IPP) [О: 5, 6, Д: 48, 49]. Ці DSCP- значення описані в RFC 2547, 2597 й 3246, відповідно. Визначено чотири класи гарантованої доставки, вони починаються з AF і далі дві цифри. Перша цифра визначає AF-клас і приймає значення від 1 до 4. Друга цифра визначає рівень імовірності скидання пакета в межах кожного класу й приймає значення від 1 (мінімальна ймовірність скидання) до 3 (максимальна ймовірність скидання). Значення DSCP можуть бути виражені в десятковому форматі або з використанням ключових слів DSCP. Наприклад, DSCP EF аналогічно DSCP 46, а DSCP AF31 аналогічно DSCP 26.

MPLS EXP. Біти MPLS EXP – це три біти в MPLS – мітці, що містять індикатор QoS. За замовчуванням, під час додавання формування мітки в значення MPLS EXP записується значення поля IPP IP пакета. Розмір поля дозволяє використовувати до восьми QoS маркувань проти 64 для DSCP. Значення EXP біт використовуються для визначення PHB-політик на вузлах MPLS мережі й можуть використовуватися для забезпечення прозорості переносу значень IPP/DSCP у пакетах клієнта у випадку застосування MPLS-методів тунелювання Diffserv (MPLS DiffServ Tunneling Modes) [О: 10, 11, Д: 43].

10.3. Інструменти планування

Інструментами планування (scheduling) називаються засоби, які визначають, як кадр або пакет буде виходити з мережного вузла [О: 5, 6, Д: 48, 49]. У кожному разі, коли пакети входять у пристрій швидше, ніж виходять із нього (тобто у випадку розбіжності швидкостей на вході й виході) виникає «точка перевантаження» або «вузьке місце». У мережних пристроїв є буфери, які дозволяють сформувати чергу з пакетів, що надходять. У разі реалізації пріоритетних механізмів обробки черг, пакети з більшим пріоритетом будуть вилучати з черги і направлятися для передачі частіше, ніж менш пріоритетні. Цей підхід називається «черговістю» (queuing).

Алгоритми черговості починають працювати тільки в моменти переповнення буферу й відключаються після того, як переповнення зникає [О: 5, 6, Д: 48, 49]. Коли переповняється буфер черги, пакети що входять, починають відкидатися або в загальному порядку (tail-drop), або вибірково. Вибіркове відкидання пакетів до повного заповнення буфера називається технологією запобігання переповненню (congestion avoidance). Механізми запобігання переповненню найкраще працюють із прикладними програмами, які використовують протокол TCP, оскільки у випадку відкидання пакетів механізми TCP починають автоматично знижувати швидкість передачі до прийнятного рівня. Механізми запобігання переповненню добре сполучаються з алгоритмами буферизації. Алгоритми буферизації управляють «головою» черги, а механізми боротьби з переповненням управляють її «хвостом» і тим самим побічно допомагають зазначеним алгоритмам справлятися зі своїми завданнями.

Найбільш поширеними інструментами планування є наступні [О: 5, 6, Д: 48, 49].

Class-Based Weighted Fair Queuing (CBWFQ – справедливе зважене керування чергами на основі класів) – це композитний алгоритм зваженої справедливої черговості, що дозволяє визначати класи за вказаними критеріями, таким як списки доступу (ACL), вхідний інтерфейс, протокол і так

далі. У рамках Cisco MQC (Modular QoS Command-Line Interface) він дозволяє виділяти смугу для кожної з 64 можливих черг й обробляти ці черги по алгоритму справедливої черговості. Він, також, підтримує механізм WRED (Weighted Random Early Detect – зважене випадкове раннє виявлення), як механізм, що забезпечує свою політику відкидання пакетів для кожного класу трафіка.

Modified Deficit Round Robin (MDRR) – це композитний механізм, заснований на використанні класів трафіка, який дозволяє організувати буферизацію до восьми класів на платформах Cisco відповідних серій. Він працює подібно CBWFQ і також дозволяє організовувати буферизацію трафіка чутливого до затримок у випадку використання черги строгого пріоритету. MDRR може застосовуватися на напрямку до матриці комутації (на вхідному інтерфейсі) або від матриці (на вихідному інтерфейсі). Біти MPLS EXP й IPP обробляються в MDRR однаково.

Low-Latency Queuing (LLQ – використання черги строгого пріоритету у випадку MDRR) – використовує виділену чергу для обробки трафіка чутливого до затримки, такого як голос або відео. LLQ в Cisco MQC включає функцію полісингу трафіка, що дозволяє обробляти пакети в певній черзі LLQ у момент надходження, що, у свою чергу, дозволяє конфігурувати кілька черг строгого пріоритету (наприклад, одна LLQ черга може бути сконфігурована для голосу, а інша для інтерактивного відео). Програмне забезпечення абстрагується від того факту, що насправді існує тільки одна LLQ черга. Продовжуючи приклад, якщо для пріоритетної черги сконфігуровано 100 Кб/с для голосового трафіка й 400 Кб/с для інтерактивного відео, то програмне забезпечення налаштує пріоритетну чергу на 500 Кб/с. Воно буде обробляти голосовий трафік як трафік строгої пріоритетності до 100 Кб/с. Потім, будь-який додатковий голосовий трафік буде підданий полісингу (скинутий), що підкреслює важливість правильного налаштування CAC (Call Admission Control). Алгоритм буде продовжувати виділяти для інтерактивного відео 400 Кб/с смуги пропускання. Якщо надійде більше 400 Кб/с відео-трафіку, то надлишковий трафік буде

знову ж підданий полісингу. При алгоритмі MDRR зі строгою пріоритетністю пріоритетна черга буде оброблятися доти, поки в цій черзі є пакети. Крім подібної реалізації в MDRR існує буферизація з альтернативною пріоритезацією. У цьому режимі пріоритетна черга обслуговується по черзі з іншими чергами.

WRED – це механізм попередження перевантажень у мережі, що дозволяє здійснити інтелектуальне відкидання пакетів на підставі маркування конкретного пакета в момент виникнення перевантаження. Незважаючи на те, що відкидання здійснюються й випадковим чином, статистично, пакети з меншим пріоритетом відкидаються більш агресивно при досягненні адміністративно заданого порога заповнення черги. WRED дозволяє уникнути проблем з масовим відкиданням пакетів у хвості черги.

10.4. Канальні механізми

До канальних механізмів відносять наступні [О: 5, 6, Д: 48, 49].

Засоби обмеження швидкості (Policing) і вирівнювання трафіка (Shaping). Як засоби полісингу, так і засоби вирівнювання трафіка зазвичай ідентифікують порушення профілю трафіку користувача однаково. Їхня головна відмінність полягає в тім, як вони реагують на ці порушення. Засоби обмеження швидкості (policer) зазвичай скидають трафік. Засіб вирівнювання трафіку зазвичай затримує надлишковий трафік у буфері, використовуючи буфер для зберігання пакетів і вирівнювання потоку у випадку сплесків трафіку.

Фрагментація й чергування пакетів (Link Fragmentation and Interleaving). У випадку застосування низькошвидкісних каналів для передачі великих пакетів через канал потрібно значно більше часу. Цю затримку називають затримкою серіалізації й вона може привести до того, що для голосових пакетів буде перевищене значення порога затримки й/або коливань затримки (jitter). Існує два основних інструменти скорочення затримки на низькошвидкісних каналах: фрагментація й чергування пакетів для багатоканального PPP (Link-Fragmentation and Interleaving for Multilink Point-to-Point Protocol) і Frame Relay фрагментація (FRF.12).

Механізми компресії – технології компресії, такі як Compressed Real-Time Protocol (cRTP), мінімізують вимоги до смуги пропускання й надзвичайно ефективні на низькошвидкісних каналах. Заголовок IP-пакета в 40 байт може скласти практично дві третини всього пакета. Для того, щоб уникнути неефективного використання доступної смуги пропускання каналу застосовується механізм cRTP (він працює не глобально, а на окремому каналі – точка-точка). Застосування cRTP дозволяє скоротити IP/UDP/RTP заголовок з 40 до 2-5 байт.

Налаштування буфера TX Ring-Transmit – TX ring. Це остання FIFO черга, у якій перебувають кадри перед їхнім відправленням на фізичний інтерфейс. Призначення цього буфера полягає в тому, щоб гарантувати наявність готового до передачі кадру на той момент, коли інтерфейс буде готовий прийняти кадр і, таким чином, домогтися 100 відсоткової утилізації каналу. Розмір TX Ring буфера залежить від моделі маршрутизатора, програмного забезпечення, протоколу другого рівня й механізму буферизації, сконфігурованого на інтерфейсі.

10.5. Вимоги до якості обслуговування корпоративних користувачів й «Базові Основи QoS»

При проектуванні мережі оператори повинні враховувати вимоги до якості обслуговування, які висуваються корпоративними клієнтами [О: 5, 6, Д: 48, 49]. Загальна проблема операторів і корпоративних клієнтів полягає в багатстві QoS функціональності Cisco IOS й, як наслідок, великої кількості варіантів реалізації й комбінацій. Для того, щоб дати деякі загальні рекомендації з реалізації якості обслуговування, Cisco розробила документ, який отримав назву «Базові Основи QoS». Метою даного документу є уніфікація рішень на апаратних платформах Cisco. В «Базових Основах QoS» специфіковані маркування й правила обробки до 11 класів сервісу в корпоративних мережах. Важливо відзначити, що «Базові Основи QoS» не диктують кожному корпоративному клієнтові негайно впровадити 11 класів трафіка, а скоріше враховують існуючі й майбутні потреби в підтримці QoS. Навіть якщо корпоративному клієнтові зараз потрібна тільки частина із цих

11 класів, то відповідність рекомендаціям «Базових Основ QoS» дозволить йому у майбутньому плавно мігрувати на розширення кількості підтримуваних класів.

Рекомендації проміжного варіанта «Базових Основ QoS» по маркуванню трафіка представлені в таблиці 10.2 [О: 5, 6, Д: 48, 49].

В «Базових Основах QoS» рекомендується маркувати голосову сигналізацію за допомогою CS3. Однак у даний момент, Cisco IP телефони маркують сигналізацію як AF31. Міграція від AF31 до CS3 вже запланована, але, як тимчасове рішення, рекомендується зарезервувати AF31 та CS3 для сигналізації й використати DSCP 25 для локальних критичних даних прикладних програм. По завершенні міграції варто застосовувати рекомендації «Базових Основ QoS» по маркуванню сигналізації за допомогою CS3 і локальних критичних додатків за допомогою AF31. Ці рекомендації з маркування більше відповідають RFC2597 та RFC2474 [Д: 22, 27].

Таблиця 10.2 – Рекомендації «Базових Основ QoS» по маркуванню трафіка

Прикладна программа	**Класифікація L3**			**Класифікація L2 CoS/MPLS-exp**
	IPP	**PHP**	**DSCP**	
Маршрутна інформація	6	CS6	48	6
Голос	5	EF	46	5
Інтерактивне відео	4	AF41	34	4
Потокове відео	4	CS4	32	4
Дані чутливі до втрат	3	–	25	3
Сигналізація дзвінків	3	AF31/CS3	26/24	3
Транзакційні дані	2	AF21	18	2
Мережне керування	2	CS2	16	2
Об’ємний клас	1	AF11	10	1
Інтернет/Scavenger	1	CS1	8	1
Все інше	0	0	0	0

Друга версія Cisco AutoQoS буде автоматично конфігурувати QoS для голосу, відео й даних. Cisco AutoQoS Enterprise буде визначати й конфігурувати до 10 класів трафіка на підставі моделі відображеної вище (єдиний клас, що не буде конфігуруватися автоматично – це локальні критичні дані, тому що цей клас вимагає поінформованості про бізнес-процеси, що перебуває за межами можливостей AutoQoS). Cisco AutoQoS Enterprise призначений для автоматизації й спрощення проектування мережі відповідно до «Базових Основ QoS» [О:5].

10.6. Виділення смуги пропускання для різних типів даних

Вимоги до якості обслуговування голосових даних

Як оператори, так і їхні замовники повинні забезпечувати необхідну смугу пропускання для програм голосу й відео, щоб гарантувати присутність необхідних ресурсів в мережі [О: 5, 6, Д: 48, 49]. Оператори повинні також забезпечувати й необхідні параметри затримки для VoIP-пакетів у випадку виникнення перевантажень у мережі. Оператори повинні гарантувати відповідну смугу пропускання для VoIP-програм корпоративних клієнтів. Планування повинне бути досить точним і враховувати декілька факторів. Для виділення відсотка загальної смуги пропускання факторами, що лімітують, є параметри затримки й коливань затримки, а також загальна пропускна здатність каналу. Трафік, що попадається в пріоритетних чергах (LLQ або PQ), піддається затримці внаслідок рівня утилізації каналу й затримки серіалізації одного VoIP-пакета. Максимальний відсоток VoIP-трафіка на конкретному каналі між оператором і клієнтом залежить від наступних факторів [О: 5, 6, Д: 48, 49]:

- бюджету затримки на каналі доступу;
- максимальної затримки на порожній LLQ/PQ черзі. Ця затримка повинна виключати і затримку, викликану конфліктами VoIP-пакетів;
- пропускної здатності. Для даної пропускної здатності затримка, викликана конфліктами VoIP-пакетів, буде рости з ростом інтервалів пакетизації й наступним ростом затримки серіалізації VoIP-пакета;
- використовуваного кодеку й інтервалу пакетизації. Для даного кодека й інтервалу пакетизації, затримка, викликана конфліктами VoIP-пакетів, буде рости із відсотковим ростом LLQ/PQ складової трафіку. Затримка буде падати з ростом смуги пропускання каналу. Чим більш швидкісні кодеки застосовуються, тим більше буде затримка, пов’язана з ростом розміру VoIP-пакетів, що, у свою чергу, збільшує затримку серіалізації.

Ґрунтуючись на вище викладеному, приклад планування VoIP у частині відсоткового співвідношення PQ-трафіка в мережі оператора виглядає наступним чином:

- швидкість доступу до мережі 512 Кб/с й оптимальний бюджет по затримці на каналі доступу 15 мсек;
- гірший варіант затримки незавантаженої PQ-черги – 10 мсек;
- 5 мсек – затримки буферизації VoIP;
- G.711 кодек дасть 20 мсек без cRTP.

У цьому випадку можна гарантувати максимум два виклики, що становить приблизно 35 відсотків завантаження PQ-черги. Це дає нам приблизно 180 Кб/с з невеликим запасом. При розрахунку смуги пропускання для VoIP необхідно забезпечити досить додаткової ємності й для трафіку даних.

Вимоги до якості обслуговування для відео

Існує два основних типи відеопрограм: інтерактивне відео (наприклад, відеоконференції) і потокове відео (наприклад, IP/TV, що може використовувати як одно- так і багатоадресне розсилання).

Налаштування для трафіка інтерактивного відео

При налаштуванні інтерактивного відео (відеоконференцій) рекомендується наступне [О: 5, 6, Д: 48, 49]:

- інтерактивний відеотрафік, відповідно до «Базових основ QoS», повинен бути промаркірований AF41;
- втрати повинні бути не більше одного відсотка;
- односпрямована затримка повинна бути не більше 150 мсек;
- коливання затримки повинні бути не більше 30 мсек;
- мінімально гарантована смуга пропускання (LLQ) повинна дорівнювати розміру сесії відеоконференції плюс 20 відсотків (наприклад, сесія відеоконференції в 384 Кб/с вимагає налаштування 460 Кб/с смуги трафіку гарантованого пріоритету).

Відеоконференція включає аудіокодек G.711 і, відповідно, вимоги до втрат, затримці й коливанням затримки аналогічні голосовому трафіку. Однак

трафік відеоконференції має і відмінності від трафіка голосу. Наприклад, трафік відеоконференцій використовує змінні розміри пакетів і змінні швидкості передачі пакетів. Швидкість передачі трафіку відеоконференції – це швидкість семплірування відеопотоку, але не реальна смуга пропускання, що вимагає відеовиклик. IP, UDP й RTP заголовки (40 байт на пакет) повинні бути додатково включені у вимоги до смуги пропускання (також як і заголовки другого рівня). Враховуючи змінні розміри пакетів і, відповідно, швидкості генерації пакетів, досить важко точно підрахувати абсолютне значення накладних витрат. Практика показує, що для розрахунку можна використовувати швидкість передачі трафіку відеоконференції плюс 20 відсотків.

Налаштування для трафіка потокового відео

При налаштуванні потокового відео рекомендується таке [О: 5, 6, Д: 48, 49]:

- потокове відео (одноадресного або багатоадресного розсилання), відповідно до «Базових основ QoS» повинне бути промаркіроване CS4;
- втрати повинні бути менш 2 відсотків;
- затримка повинна бути менш 4-5 секунд (залежно від можливостей буферизації відео прикладних програм);
- не існує значних вимог по коливанню затримки;
- вимоги по гарантіях смуги (CBWFQ) залежать від формату кодування швидкості відеопотоку;
- потокове відео звичайно односпрямоване й, тому, у віддалених філіях маршрутизатори можна не налаштовувати на підтримку потокового відео в напрямку від філії до центра;
- неважливі програми потокового відео, такі як відео для розваги, можуть бути промаркіровані DSCP CS1 і для них потрібен мінімум гарантій смуги пропускання в черзі CBWFQ (використовуючи клас Інтернет/scavenger). Програми потокового відео менш вимогливі до QoS, тому що вони менш чутливі до затримок (може пройти кілька секунд перед початком відеокартинки) і нечутливі до коливань затримки (завдяки буферизації на рівні програм). Однак потокове відео

може містити важливу інформацію, таку як електронне навчання або трансляцію корпоративних нарад й, отже, вимагати гарантій QoS. Неважливий відеозміст (такий як фільми, музичні відеокліпи тощо) може розглядатися як Інтернет-сервіс (сервіс гірше чим «Best Effort»), що означає, що ці потоки працюють поки є смуга пропускання, але будуть витіснятися у випадку виникнення перевантажень у мережі.

Вимоги до якості обслуговування для трафіка даних

Визначаючи вимоги QoS для даних, потрібно брати до уваги такі фактори [О:5, 6, Д: 48].

- доцільно використовувати не більше чотирьох основних класів трафіка, таких як:
  - локально-визначений критичний (для критично важливих програм);
  - транзакційні й інтерактивні програми з високим бізнес-пріоритетом;
  - транзакційний/інтерактивний – програми клієнт-сервер, програми по передачі повідомлень;
  - об’ємні програми – передача великих файлів, синхронізація й реплікація баз даних, електронна пошта (e-mail);
  - «по можливості» – клас за замовчуванням для всього непризначеного трафіку (необхідно виділити як мінімум 25 відсотків смуги для цього класу);
  - опціонний клас – Інтернет/scavenger (ігровий трафік, розваги).
- додатковий опціонний клас містить у собі маршрутизацію й мережне керування.

Локально-визначений критичний клас трафіка даних

Локально-визначений критичний клас трафіка даних – це можливо найбільш невизначений клас трафіка в «Базових основах QoS». Відповідно до моделі «Базових основ QoS» всі класи трафіка розглядаються як критичні для корпоративного клієнта. Термін локально-визначений використається, щоб підкреслити призначення цього класу – тобто мати вищий клас сервісу для

кожного клієнта для обраного набору їх інтерактивних і транзакційних програм, що має для них найбільший бізнес-пріоритет. Наприклад, компанія може настроїти Oracle та SAP, як транзакційний/інтерактивний клас. Однак більша частина їхніх доходів залежить від SAP і, тоді вони можуть захотіти дати цьому транзакційному прикладному продукту ще більш високий рівень пріоритету, призначивши його у виділений клас (такий як локально-визначений критичний клас). Враховуючи, що критерій призначення такого класу не технічний (а визначається бізнес вимогами й цілями організації), рішення про те, які додатки повинні потрапити в цей спеціальний клас можуть бути вирішені тільки організаційно. У цей клас рекомендується призначати як можна меншу кількість програм.

Транзакційний/інтерактивний клас даних

Транзакційний/інтерактивний клас – це комбінація двох схожих типів програм: транзакційних програм клієнт-сервер і програм інтерактивних повідомлень. Вимоги за часом відгуку відрізняють транзакційні від звичайних клієнт-серверних програм. У випадку транзакційних клієнт-серверних програм (таких як SAP) користувач змушений очікувати завершення операції. E-mail – це не транзакційна програма, тому що більшість операцій відбувається у фоновому режимі й користувач, звичайно, не помічає затримок у кілька сотень мілісекунд.

Об’ємний клас даних (Bulk)

Об’ємний клас даних призначений для неінтерактивних програм, не критичних до втрат пакетів, що зазвичай працюють у фоновому режимі. Такі програми включають: FTP, e-mail, backup, синхронізація й реплікація баз даних, поширення відео-змісту або інших типів програм, у яких користувач не змушений чекати результатів виконання операцій. Перевага виділення смуги пропускання для об’ємного класу трафіка (замість накладання на них обмежень) полягає в тім, що програма може динамічно використовувати вільну смугу пропускання й, таким чином, підвищувати свою продуктивність у періоди низького навантаження, що, у свою чергу, знижує ймовірність впливу на них перевантажень.

Клас «по можливості» (Best Effort)

Клас «по можливості» – це клас за замовчуванням для всього трафіка даних. Тільки в тому випадку, якщо додаток був відібраний для особливої обробки, він буде вилучений із класу за замовчуванням. Враховуючи, що багато корпоративних клієнтів використають сотні, якщо не тисячі, програм даних у своїх мережах (більшість із яких і залишиться в цьому класі), потрібно виділити адекватну смугу пропускання для класу за замовчуванням. У противному випадку, програми, які потрапили в цей клас, будуть подавлені. Рекомендується виділяти, принаймні, 25 відсотків смуги пропускання для підтримки класу трафіка «по можливості».

Інтернет/Scavenger клас

Інтернет/Scavenger клас призначений для надання певним програмам диференційованих послуг або послуг по залишковому принципу. Ці програми, як правило, не мають прямого відношення до основного бізнесу компанії й зазвичай орієнтовані на розваги. Вони включають: програми однорангового медіа-обміну (KaZaa, Morpheus, Groekster, Napster, iMesh тощо), ігрові програми (Doom, Quake, Unreal Tournament, і т.д.), і розважальні відеопрограми. Це типовий клас, визначений для корпоративного клієнта, але звичайно перемаркірований на клас «Best Effort» на границі мережі оператора. Призначення мінімальних гарантій по смузі пропускання цьому класу означає, що в моменти перевантажень він практично не одержує ніяких ресурсів, але може використати вільну смугу в спокійні періоди.

Класи маршрутизації й мережного керування

Для корпоративних мереж доцільно у явному вигляді гарантувати мінімальну смугу пропущення для протоколів маршрутизації й програм мережного керування (SNMP, NTP, Syslog або NFS). Важливо відзначити, що протоколи внутрішньої маршрутизації, такі як RIP й EIGRP, зазвичай не вимагають виділення спеціальної смуги пропускання. Це перевага використання внутрішнього механізму Cisco – PAK_PRIORITY. В OSPF тільки

пакети hello маркуються за допомогою PAK_PRIORITY. В BGP (хоча він і позначається IPP6/CS6) також можуть знадобитися спеціальні налаштування.

Запитання для самоперевірки та контролю засвоєння знань

1. Які параметри можуть бути використані для класифікації й маркування трафіку?
2. В якому місці мережі рекомендується ідентифікувати й позначати трафік?
3. Які механізми можуть бути задіяні для маркування трафіка?
4. Що розуміють під інструментами планування?
5. У чому полягає головна відмінність механізмів обмеження швидкості (Policing) і вирівнювання трафіка (Shaping)?
6. Яку задачу вирішують механізми фрагментації й чергування пакетів?
7. Яка задача вирішується за допомогою механізмів компресії?
8. Наведіть основні вимоги до якості обслуговування голосових даних.
9. Наведіть основні вимоги до якості обслуговування відеоданих.
10. Наведіть рекомендовані параметри при налаштуванні потокового відео.
11. Наведіть основні вимоги до якості обслуговування трафіка звичайних комп'ютерних даних.
12. Наведіть основні вимоги до якості обслуговування локально-визначеного критичного класу трафіка даних.
13. Наведіть основні вимоги до якості обслуговування транзакційно/ інтерактивного класу трафіка даних.
14. Наведіть основні вимоги до якості обслуговування класу «по можливості» (Best Effort).

БІБЛІОГРАФІЧНИЙ СПИСОК

Основна література

1. Технології та засоби керування в інформаційних мережах: Курс лекцій для студентів спеціальності «Виробництво електронних засобів» / Уклад.: П. В. Кучернюк. – К.: НТУУ «КПІ», 2011. – 122 с.

2. Гребешков А.Ю. Стандарты и технологии управления сетями связи // М.: Эко-Трендз, 2003. – 288 с.

3. Гребешков А.Ю. Управление сетями электросвязи по стандарту TMN // Учеб. пособие. – М.: Радио и связь, 2004. – 155 с.

4. Дуглас Мауро, Кевин Шмидт. Основы SNMP // СПБ.: Символ-Плюс, 2012. – 516 с.

5. Вегешна Ш. Качество обслуживания в сетях IP // М.: Изд-во «Вильямс», 2003. – 368 с.

6. Кучерявый Е. А. Управление трафиком и качество обслуживания в сети Интернет // СПБ.: Изд-во «Наука и Техника», 2004. – 336 с.

7. Олифер В. Г., Олифер Н. А. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. – 4-е изд. // СПб.: Питер, 2010. – 944 с.

8. Комп'ютерні мережі та засоби телекомунікацій: Засоби телекомунікацій: Текст лекцій для студентів спеціальності «Радіоелектронні апарати та засоби»/ Уклад.: П.В. Кучернюк. – К.: НТУУ «КПІ», 2013. – 127 с.

9. Кучернюк П. В. Комп'ютерні мережі: Навчальний посібник для студентів спеціальності «Радіоелектронні апарати та засоби». – К.: НТУУ «КПІ», 2015. – 238 с.

10. Вивек Олвейн. Структура и реализация современной технологии MPLS // М.: Изд-во «Вильямс», 2004. – 480 с.

11. Технология и протоколы MPLS: Научно-практическое пособие / Гольдштейн А.Б., Гольдштейн Б.С. – СПб.: БХВ-Петербург, 2014. – 304 с.

Додаткова література

1. ISO/IEC 7498-4:1989 [ISO/IES 7498-4:1989] Information processing systems – Open systems interconnection – Basic Reference Model – Part 4: Management FrameWork [Електронний ресурс]. – Режим доступу: https://www.iso.org/obp/ui/#iso:std:iso-iec:7498:-4:ed-1:v1:en

2. Кучернюк П. В., Стеченко Н. В. Архитектура интеллектуальной системы управления информационными сетями // Электроника и связь. – 2009. – № 1. – С. 58-61.

3. Principles for a telecommunications management network: ITU-T. Recommendation. M.3010. – ITU, 1996. – P. 75.

4. TMN management functions: ITU-T. Recommendation. M.3400. – ITU, 2001. – P. 93.

5. Семейство стандартов SNMP [Електронний ресурс]. – Режим доступу: https://ru.wikibooks.org/wiki/Семейство_стандартов_SNMP

6. Семенов Ю.А. Протокол управления SNMP [Електронний ресурс]. – Режим доступу: http://book.itep.ru/4/44/snm_4413.htm

7. Structure and identification of management information for TCP/IP-based internets [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc1155

8. Management Information Base for network management of TCP/IP-based internets [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc1156

9. Simple Network Management Protocol (SNMP) [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc1157

10. Management Information Base for Network Management of TCP/IP-based internets: MIB-II [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc1213

11. A Convention for Defining Traps for use with the SNMP [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc1215

12. Remote Network Monitoring Management Information Base [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc1757

13. Introduction to version 2 of the Internet-standard Network Management FrameWork [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc1441
14. Structure of Management Information for version 2 of the Simple Network Management Protocol (SNMPv2) [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc1442
15. An Architecture for Describing Simple Network Management Protocol (SNMP) Management FrameWorks [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc3411
16. Management Information Base (MIB) for the Simple Network Management Protocol (SNMP) [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc3418
17. Технології та засоби керування в інформаційних мережах: Методичні вказівки до виконання розрахунково-графічних (розрахункових) робіт для студентів спеціальності «Радіоелектронні апарати та засоби» / Уклад.: П. В. Кучернюк. – Київ : НТУУ «КПІ», 2015 р. – 51 с.
18. Технології та засоби керування в інформаційних мережах: Методичні вказівки до виконання лабораторних робіт для студентів спеціальності «Радіоелектронні апарати та засоби» / Уклад.: П. В. Кучернюк. – Київ : НТУУ «КПІ», 2015 р. – 86 с.
19. Алгоритмы управления очередями [Електронний ресурс]. – Режим доступу: https://www.osp.ru/lan/2007/12/4659316
20. Specification of the Controlled-Load Network Element Service [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2211
21. Specification of Guaranteed Quality of Service [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2212
22. Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2474
23. An Architecture for Differentiated Service [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2475

24. Resource ReSerVation Protocol (RSVP) [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2205
25. Resource ReSerVation Protocol (RSVP). Version 1 Applicability Statement. Some Guidelines on Deployment [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2208
26. SBM (Subnet Bandwidth Manager):A Protocol for RSVP-based Admission Control over IEEE 802-style networks [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2814
27. Assured Forwarding PHB Group [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2597
28. An Expedited Forwarding PHB [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2598
29. An Expedited Forwarding PHB (Per-Hop Behavior) PHB [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc3246
30. Aggregation of RSVP for IPv4 and IPv6 Reservations Service [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc3175
31.Integrated Services in the Internet Architecture: an Overview[Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc1633
32. The use of RSVP with IETF integrated services [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2210
33. Specification of the controlled-load network element service [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2211
34. Specification of guaranteed quality of service [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2212
35. Стив Мак-Квери, Келли Мак-Грю, Стефан Фой. Передача голосовых данных по сетям Cisco Frame Relay, ATM и IP. – М.: Изд-во «Вильямс», 2002. – 512 с.
36. Frame Relay Fragmentation for Voice [Електронний ресурс]. – Режим доступу: https://www.cisco.com/c/en/us/support/docs/voice/voice-over-frame-relay-vofr/9232-fr-frag.html

37. VoIP over Frame Relay with Quality of Service (Fragmentation, Traffic Shaping, LLQ / IP RTP Priority) [Електронний ресурс]. – Режим доступу: https://www.cisco.com/c/en/us/support/docs/voice/voice-quality/12156-voip-ov-fr-qos.html
38. Multiprotocol Label Switching Architecture [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc3031
39. Семёнов Ю.А.Телекоммуникационные технологии [Електронний ресурс]. – Режим доступу:http://citforum.ck.ua/nets/semenov/
40. LDP Specification [Електронний ресурс]. – Режим доступу: https://www.ietf.org/rfc/rfc3036.txt
41. Описание протокола Label Distribution Protocol (LDP) [Електронний ресурс]. – Режим доступу: http://www.opennet.ru/docs/RUS/mpls/ldpdescription.html
42. BGP/MPLS VPNs [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc2547
43. Requirements for Traffic Engineering Over MPLS [Електронний ресурс] – Режим доступу: https://tools.ietf.org/html/rfc2702
44. IS-IS Extensions for Traffic Engineering [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc5305
45. Traffic Engineering Extensions to OSPF Version 3 [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc5329
46. RSVP-TE: Extensions to RSVP for LSP tunnels [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc3209
47. Inter-Domain MPLS and GMPLS traffic engineering resource reservation protocol-traffic engineering (RSVP-TE) extensions [Електронний ресурс]. – Режим доступу: https://tools.ietf.org/html/rfc5151
48. Захватов М. Качество обслуживания в операторских сетях [Електронний ресурс]. – Режим доступу: https://www.opennet.ru/docs/RUS/qos_oper/#mozTocId308174
49. Гольдштейн Б.С. IP-телефония//СПБ.: БХВ-Петербург, 2006. – 336 с.

ДОДАТОК А
Перелік скорочень

AAL	ATM Adaptation Layer – рівень адаптації ATM
ABR	area border router – прикордонний маршрутизатор області
ABR	Available Bit Rate – доступна швидкість передачі
AC	Application Context
ACL	Access Control List – список доступу
ACSE	Association Control Service Element
ADSL	Asymmetric Digital Subscriber Line
AF	Assured forwarding – гарантована передача
ANSI	American national standards institute
API	Інтерфейс прикладного програмування
ARP	Address Resolution Protocol
ASN.1	Abstract Syntax Notation One
ATM	асинхронний режим передачі
Atom	Any Transport over MPLS – будь-який транспорт поверх MPLS
Bc	узгоджений розмір сплеску
BE	Best effort – негарантована доставка
Be	розширений розмір сплеску
BECN	Backward Explicit Congestion Notification – зворотне явне повідомлення про перевантаження
BGP	Border Gateway Protocol – протокол суміжного шлюзу
CAR	Committed Access Rate – узгодження швидкості доступу
CBC	Cipher Block Chaining
CBQ	Class Based Queuing – обслуговування на основі класу
CBR	Constant Bit Rate – постійна бітова швидкість
CBWFQ	Class-Based Weighted Fair Queuing – зважений механізм рівномірного обслуговування черг на основі класу трафіку
CDR	Committed Delivery Rate – погоджена швидкість доставки

CE	Customer Edge – границя клієнта
CEF	Cisco Express Forwarding – швидка комутація Cisco
CFI	Canonical Format Identifier – ідентифікатора канонічного формату
CIR	Committed Information Rate – узгоджена швидкість передачі
CLP	Cell Loss Priority – пріоритет відкидання комірки
CMIP	Common Management Information Protocol
CMIS	Common Management Information Service
CMISE	Common Management Information Service Element
COPS	Common Open Policy Service – спільна відкрита служба політик
CoS	Class of Service – клас послуги
CPU	Central processing unit
CQ	Custom Queuing – черга, що настроюється
CS	Classs Selector – селектор класу
DE	Discard Eligible – придатний для відкидання
DES	Data Encryption Standard
DLCI	Data-Link Connection Identifier – ідентифікатор підключення канального рівня
DMI	Definition of Management Information
DN	Distinguished Name
DPI	Distributed Protocol Interface
DRR	Deficit Round Robin – механізм кругового обслуговування з дефіцитом
DSBM	Designated SBM – призначений SBM
DSCP	Differentiated Services Code Point – код диференційованого обслуговування
DTE	Data terminal equipment – термінальне обладнання даних
EF	Expedited forwarding – негайна передача
EGP	Exterior Gateway Protocol
EPD	Early Packet Discard – раннє відкидання пакетів

ER-LSP	Explicitly Routed Label Switched Path – LSP-шлях, що явно маршрутизується
ETSI	European Telecommunications Standards Institute
FCS	Frame Check Sequence – контрольна послідовність кадру
FDN	Full Distinguished Name
FEC	Forwarding Equivalency Class – клас еквівалентності пересилки
FECN	Forward Explicit Congestion Notification – пряме явне повідомлення про перевантаження
FF	Fixed Filter – резервування з фіксованим фільтром
FIFO	First-in, First-out – «першим прийшов, першим обслужений»
FQ	Fair Queuing – справедливе обслуговування
FRF	Frame Relay Fragmentation – фрагментація Frame Relay
FRF	Frame Relay Forum – форум Frame Relay
FRTS	Frame Relay Traffic Shaping – вирівнювання трафіку Frame Relay
FTP	File Transfer Protocol – протокол передачі файлів
GB	Guaranteed bandwidth – гарантована смуга пропускання
GDMO	Guidelines for the Definition of Managed Objects
GMI	Generic Management Information
GPS	Generalized Processor Sharing – узагальнений поділ процесорного часу
GTS	Generic Traffic Shaping
GUI	Graphical user interface
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
ICMP	Internet Control Message Protocol
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force – інженерна спеціалізована група Інтернет
IGP	Interior Gateway Protocol – протокол внутрішнього шлюзу

ILMI	Interim Local Management Interface – проміжний інтерфейс локального управління
IP	Internet Protocol – міжмережний протокол
IPP	IP Precedence – біт IP Precedence
IPX/SPX	Internetwork packet exchange/Sequenced packet exchange
IS-IS	Intermediate System-to-Intermediate System – протокол обміну даними між проміжними системами
ISO	International Organization for Standardization
ITU	International Telecommunication Union
ITU-T	International Telecommunication Union – Telecommunication sector
LAN	Local area network
LDN	Local Distinguished Name
LDP	Label Distribution Protocol – протокол розповсюдження міток
LER	Label Edge Router – прикордонний маршрутизатор міток
LFIB	Label Forwarding Information Base – інформаційна база передачі міток
LIB	Label Information Base – інформаційна база міток
LLQ	Low Latency Queuing – обслуговування з малою затримкою
LMI	Local Management Interface – інтерфейс локального керування
LRO	Label Request Object – об’єкт запиту мітки
LSA	Link State Advertisement – повідомлення про стан каналу
LSP	Label Switching Path – помічений комутований шлях
LSR	Label Switching Router – маршрутизатор комутації міток
MBS	Maximum burst size- максимальний розмір сплеску
MCR	Minimum Cell Rate – мінімальна швидкість передачі комірок
MD5	Message Digest 5
MF	Mediation Function
MIB	Management Information Base
MINCIR	Minimal CIR – мінімальна CIR

MPLS	Multiprotocol Label Switching – мультипротокольна комутація міток
MQC	Cisco Modular QoS command-line interface
MRTG	Multi Router Traffic Grapher
MTU	Maximum transmission unit – максимальний розмір одиниці передачі інформації
NE	Мережний елемент
NEF	Network Element Function
NFS	Network File System – мережна файлова система
NMS	Network management systems
NNTP	Network News Transfer Protocol
nRT-VBR	Non-Real Time Variable Bit Rate – змінна швидкість передачі не в режимі реального часу
OSF	Operations System Function
OSI	Open System Interconnection – взаємодія відкритих систем
OSPF	Open Shortest Path First – відкритий протокол пошуку найкоротшого шляху
PCR	Peak Cell Rate – пікова швидкість передачі комірки
PDP	Policy Decision Point – точка визначення політики
PDU	Protocol data units
PE	Provider Edge – границя провайдеру
PEP	Policy Enforcement Point – точка вживання політики
PHD	Per-hop behavior – покрокова поведінка
PNG	Portable Network Graphics
POP3	Post Office Protocol Version 3
PPD	Partial Packet Discard – часткове відкидання пакетів
PQ	Priority Queuing – пріоритетне обслуговування
PRTG	Paessler Router Traffic Grapher
PVC	Permanent Virtual Channel – постійне віртуальне з’єднання
QAF	Q-Adapter Function

QoS	Quality of Service – якість обслуговування
RAM	Random Access Memory
RDN	Relative Distinguished Name
RED	Random Early Detect – алгоритм довільного передчасного виявлення
RFC	Request for Comments – запит коментарів
RM	Resource Management – управління ресурсами
RMON	Remote Network MONitoring
RN	Round Number – номер циклу
ROSE	Remote Operations Service Element
RPC	Remote Procedure Call
RR	Round Robin – кругове (циклічне) обслуговування
RRR	Routing by Resource Reservation – маршрутизації на основі резервування ресурсів
RSVP	Resource Reservation Protocol – gротокол резервування ресурсів
RTP	Real-time Transport Protocol – транспортний протокол реального часу
RTSE	Reliable Transfer Service Element
RT-VBR	Real-Time Variable Bit Rate – змінна швидкість передачі в режимі реального часу
SBM	Subnet Bandwidth Manager – диспетчер смуги пропускання підмережі
SCR	Sustainable Cell Rate – стабільна швидкість передачі комірки
SDH	Synchronous Digital Hierarchy
SE	Shared Explicit – загальне явне резервування
SHA	Secure Hash Algorithm
SLA	Service Level Agreement – угода про рівень обслуговування
SMAE	Systems Management Application Entities
SMI	Structure of Management Information
SNMP	Simple Network Management Protocol

SNS	Simple Notification Service
SONET	Синхронна оптична мережа
SPF	Shortest Path First – алгоритм переважного вибору найкоротшого шляху
SRO	Source Route Object – об’єкт вихідного маршруту
SSH	Secure Shell
SSL	Secure Sockets Layer
STM	Synchronous Transfer Mode
SVC	Switched Virtual Circuit – комутований віртуальний канал
TCB	Traffic Conditioning Block
TCI	Tagged Control Information – маркірована контрольна інформація
TCP	Transmission Control Protocol – протокол управління передачею
TCP/IP	Transmission Control Protocol/Internet Protocol
TE	Traffic Engineering – керування трафіком
TMN	Telecommunication Management Network
ToS	Type of Service – тип послуги
TPID	Tagged Protocol Identifier – ідентифікатор маркірованого протоколу
TS	Traffic Shaping – вирівнювання трафіку
TTL	Time To Live – час життя
UBR	Unspecified Bit Rate – невизначена бітова швидкість
UDP	User Datagram Protocol – протокол датаграм користувача
USM	User-based Security Model
VBR	Variable Bit Rate – змінна швидкість передачі
VC	Virtual Channel –віртуальний канал
VLAN	Virtual Local Area Network – віртуальна локальна мережа
VoATM	Voice over ATM – передача голосу по протоколу ATM
VoFR	Voice over Frame Relay – передача голосу по протоколу Frame Relay
VoIP	Voice over IP – передача голосу по протоколу IP

VP	Virtual Path – віртуальний шлях
VPN	Virtual Private Network – віртуальна приватна мережа
WAN	Wide Area Network
WF	Wildcard Filter – резервування з груповим фільтром
WFQ	Weighted fair queuing – зважений алгоритм рівномірного обслуговування черг
WLAN	Wireless Local Area Network
WMI	Windows Management Instrumentation
WRED	Weighted Random Early Detect – зважений алгоритм довільного передчасного виявлення
WRR	Weighted Round Robin – зважений механізм кругового обслуговування
WSF	WorkStation Function
X/Open	X/Open Company, Ltd.
XML	Extensible Markup Language
БД	База даних
ЕОП	Електронні обчислювальні пристрої
ОС	Операційна система
ОУ	Об’єкт управління
СУ	Система управління
СУБД	Система управління базами даних
ФБ	Функціональний блок