ЛЕКЦІЯ 6

Моделі надання послуг

та якість обслуговування (QoS) у глобальних мережах

Слайд 2. Пригадуємо лекцію 5 та план лекції 6

Спочатку коротко пригадаємо ключове з минулої лекції. Тема була — «Послідовні канали зв'язку у WAN та протоколи HDLC і PPP». Ми розглянули виділені канали типу T1 (1,544 Мбіт/с) і E1 (2,048 Мбіт/с), поняття DTE та DCE у Cisco-термінології, стандарти інтерфейсів V.35, X.21, HSSI. Далі — протокол HDLC, стандартизований у ISO/IEC 13239, з його бітово-орієнтованою структурою і трьома типами кадрів I, S, U; зокрема різницю між ISO HDLC та Cisco HDLC. Другу половину минулої лекції ми присвятили протоколу PPP, стандарту RFC 1661: його стек протоколів (LCP + NCP + автентифікація), 6-фазну машину станів (Dead → Establish → Auth → Network → Open → Terminate), а також два протоколи автентифікації: PAP (передає пароль у відкритому вигляді) і CHAP (безпечний challenge-response за MD5).

Сьогодні перевертаємо сторінку — переходимо до значно ширшого погляду на глобальні мережі. Не «як з'єднати два маршрутизатори», а «як провайдери продають мережеві послуги та як забезпечити їх якість». План лекції: почнемо з традиційних моделей послуг — виділені канали, MPLS VPN, керовані сервіси. Далі — хмарні моделі: IaaS, PaaS, SaaS та концепція XaaS. Потім розглянемо SLA — угоди про рівень обслуговування, які юридично фіксують гарантії. І друга велика частина лекції — усе про QoS. Розберемо 4 базові метрики, дві принципові моделі (IntServ і DiffServ), стандартні класи PHB (EF, AF, CS), механізми управління чергами (FIFO, PQ, WFQ, CBWFQ, LLQ), запобігання перевантаженням (WRED) та формування трафіку (shaping і policing).

Слайд 3. Традиційні моделі мережевих послуг

Розпочнемо з трьох класичних моделей мережевих послуг, які й досі становлять фундамент операторського бізнесу. Ліворуч на слайді — три картки: Leased Lines, MPLS VPN, Managed Services.

Перша модель — Leased Lines, оренда виділених каналів. Ми вже детально розглядали це на попередній лекції. Оператор надає клієнту постійно виділений канал фіксованої пропускної здатності між двома точками. Гарантії — смуга 24/7, мінімальна затримка. Реалізується як T1 (1,544 Мбіт/с), E1 (2,048 Мбіт/с), T3, або на вищому рівні — OC-3, OC-12 (155 Мбіт/с і вище). Застосування — критично важливі з'єднання: банки, дата-центри, з'єднання головного офісу з філіями, коли важлива не тільки швидкість, а й приватність та передбачуваність.

Друга модель — MPLS VPN. MPLS — це технологія багатопротокольної комутації міток (Multi-Protocol Label Switching), стандартизована в RFC 3031. MPLS VPN дає ізоляцію трафіку клієнтів на єдиній MPLS-інфраструктурі оператора. Ключові переваги — гарантована QoS, масштабування без зміни архітектури клієнтської мережі. Тривалий час MPLS VPN був основною технологією корпоративних WAN — його продавали великим корпораціям для з'єднання множини філій. Реалізації бувають L3VPN (RFC 4364) — на третьому рівні, оператор бачить IP-адреси клієнта і виконує маршрутизацію; та L2VPN — на другому рівні, з варіантами VPWS, VPLS, EVPN, коли оператор просто транспортує клієнтські Ethernet-фрейми.

Третя модель — Managed Network Services, керовані мережеві послуги. Це коли оператор бере на себе не просто надання каналу, а й ПОВНЕ управління обладнанням клієнта: моніторинг мережі 24/7, реагування на інциденти, планове обслуговування, зміну конфігурацій. Рівень відповідальності визначається угодою SLA. Це особливо актуально для клієнтів, які не мають власного мережевого персоналу — вони фактично аутсорсять всю мережеву інфраструктуру.

Слайд 4. Хмарні моделі надання послуг

Тепер перейдемо до сучасних хмарних моделей. Хмарні обчислення (Cloud Computing) — це надання обчислювальних ресурсів як послуги через Інтернет з оплатою за фактичне використання. NIST — Національний інститут стандартів і технологій США — визначає п'ять основних характеристик хмарних послуг: самообслуговування за запитом (клієнт може отримати ресурси без взаємодії з оператором), широкий мережевий доступ, об'єднання ресурсів (multi-tenancy), швидка еластичність (можна миттєво нарощувати або зменшувати ресурси), вимірюване обслуговування (pay-as-you-go).

Діаграма вгорі слайда показує, як розподіляється відповідальність між провайдером і клієнтом у чотирьох сценаріях: On-Premise (традиційний варіант, коли клієнт володіє і керує всім), IaaS, PaaS і SaaS. Помаранчевий колір — це те, що клієнт керує сам; синій — за що відповідає провайдер. Чим вище рівень абстракції, тим більше синього кольору.

Перша модель — IaaS, Infrastructure as a Service. Провайдер надає базові обчислювальні ресурси: віртуальні сервери, сховища даних, віртуальні мережі. Клієнт контролює операційну систему, додатки, дані та мережеві налаштування. Головні гравці — Amazon EC2, Microsoft Azure, Google Compute Engine. Типовий користувач — IT-адміністратори, які розгортають серверну інфраструктуру.

Друга — PaaS, Platform as a Service. Провайдер надає готову платформу для розробки та розгортання застосунків. Клієнт не думає про операційну систему,

віртуалізацію, мережі — він просто пише код і завантажує його на платформу. Приклади — Google App Engine, Heroku, AWS Elastic Beanstalk. Типові користувачі — розробники ПЗ.

І найвищий рівень — SaaS, Software as a Service. Провайдер надає готові програмні додатки безпосередньо через Інтернет. Клієнт відповідає ТІЛЬКИ за свої дані. Все інше — обладнання, ОС, програмне забезпечення, оновлення — на провайдері. Класичні приклади: Microsoft 365 (Word, Excel, Outlook через хмару), Google Workspace, Salesforce (CRM), Zoom (відеоконференції). Типовий користувач — бізнес-користувач, який навіть не хоче знати, що таке сервер.

Компроміс тут очевидний: чим вище рівень (від IaaS до SaaS), тим більше зручності, але тим менше контролю. IaaS — максимум гнучкості, максимум складності. SaaS — мінімум складності, мінімум гнучкості.

Слайд 5. XaaS та угоди SLA

Три класичні моделі — IaaS, PaaS, SaaS — стали настільки успішними, що виникло метапоняття XaaS — Everything as a Service, «все як послуга». Ідея — надавати БУДЬ-ЯКІ IT-ресурси у формі хмарних сервісів. За останнє десятиліття з'явилися десятки нових аббревіатур. Ліворуч на слайді — чотири найважливіші.

NaaS — Network as a Service. Мережеву інфраструктуру як хмарний сервіс. Клієнт отримує віртуальні мережеві ресурси — VPN, брандмауер, балансувальник навантаження — без придбання фізичного обладнання. Це особливо зручно для стартапів, які починають малими.

SECaaS — Security as a Service. Кібербезпека через хмару. З'явилася на хвилі DDoS-атак: очевидно, що клієнт сам не може захиститися від атаки на кілька терабіт за секунду, потрібна хмарна інфраструктура з великою пропускною здатністю. Сучасні SECaaS-провайдери — Cloudflare, Akamai. Надають захист від DDoS, WAF (Web Application Firewall), моніторинг вразливостей.

UCaaS — Unified Communications as a Service. Інтегровані комунікаційні сервіси: VoIP, відеоконференції, обмін повідомленнями, спільна робота з документами — все в одному пакеті. Приклади: Microsoft Teams, Zoom, Cisco Webex. Пандемія 2020 року різко прискорила їх поширення.

SASE — Secure Access Service Edge. Термін від Gartner 2019 року. Об'єднує SD-WAN та SECaaS в єдиний хмарний сервіс. Ідея — не мати окремих продуктів для мережі і безпеки, а мати єдину хмарну платформу, яка одночасно й транспортує трафік, і захищає його. Це — сучасний тренд для великих корпорацій з розподіленою робочою силою.

Тепер про SLA — Service Level Agreement, угоду про рівень обслуговування. Це формальний, юридично зобов'язальний документ, який визначає параметри якості послуг та відповідальність сторін. SLA — ключовий елемент будь-якої моделі надання послуг: без SLA обіцянки провайдера — просто маркетинг. З SLA — юридичне зобов'язання.

Типова SLA включає 4 групи параметрів, показані праворуч. Перше — доступність. Наприклад, 99,95% — це означає, що допустимий час простою не більше 4 годин 22 хвилин на рік. Для критичних систем вимагають 99,999% (5 хвилин на рік) або навіть 99,9999% (32 секунди на рік). Друге — продуктивність: пропускна здатність, затримка, джитер, втрати пакетів. Це саме те, про що ми говоритимемо в другій частині лекції — метрики QoS. Третє — параметри підтримки: час реагування на звернення (наприклад, 15 хвилин для критичних інцидентів), час відновлення (MTTR — Mean Time To Repair). І четверте — механізм компенсації. Це так звані service credits — якщо провайдер не витримав SLA, клієнт отримує знижку від 5% до 100% від місячної плати залежно від рівня порушення.

Слайд 6. Що таке QoS і навіщо він потрібен

Тепер переходимо до другої великої теми лекції — якості обслуговування, QoS. Quality of Service — це сукупність технологій та механізмів, що забезпечують диференційоване обслуговування різних типів мережевого трафіку відповідно до їхніх вимог.

Що ж таке проблема? Погляньте на верхній банер слайда — це принципова проблема, яку вирішує QoS. Без механізмів QoS мережа обробляє BCI пакети однаково, за принципом «найкращих зусиль» — best effort. Кожен пакет — просто послідовність байтів. Маршрутизатор не знає і не хоче знати, що в цьому пакеті — голос, відео, чи резервне копіювання. Він просто отримує пакет, дивиться в таблицю маршрутизації, і надсилає далі. Якщо в цей момент мережа перевантажена, і всі черги повні — пакет може бути відкинутий. І байдуже, чи це був пакет VoIP-виклику важливого топ-менеджера, чи 100-й пакет Torrent-скачування популярного фільму.

Це працює нормально, поки мережа не перевантажена. Але як тільки трафіку стає більше, ніж може пропустити канал — починаються проблеми. І першими страждають найчутливіші сервіси: голос, відео, інтерактивні застосунки.

Тому потрібне рішення: диференціювати обробку різних типів трафіку. Пріоритизувати чутливе, знижувати пріоритет байдужого. Саме це й робить QoS.

Для оцінки якості каналу використовуються чотири основні метрики, показані внизу слайда. Bandwidth — пропускна здатність, максимальний обсяг

даних за одиницю часу (біт/с, кбіт/с, Мбіт/с). Delay — затримка, час передачі пакета від джерела до одержувача (у мілісекундах). Jitter — джитер, варіація затримки між послідовними пакетами (теж у мс). І Packet Loss — втрати пакетів, відсоток пакетів, які не досягли одержувача.

Різні типи трафіку по-різному чутливі до цих метрик. Голос дуже чутливий до затримки і джитера, але майже не потребує смуги (кодек G.729 — усього 8 кбіт/с). Файловий обмін — навпаки: смуги хоче багато, але терпить затримку. Тому QoS має вміти диференціювати.

Слайд 7. Рекомендовані параметри QoS для різних типів трафіку

Це, мабуть, найважливіший слайд лекції з точки зору практичних цифр. У таблиці — рекомендовані параметри QoS для типових видів трафіку. Це Таблиця 1.3 з навчального посібника. Джерела — рекомендації ITU-T G.114 для голосу і RFC 5127 для класифікації трафіку.

Пройдемо по рядках. VoIP — голосовий трафік. Затримка — не більше 150 мілісекунд. Це рекомендація ITU-T G.114, вироблена на основі психоакустичних досліджень: більше 150 мс — і люди починають перебивати одне одного, розмова стає незручною. Джитер — не більше 30 мс. Втрати пакетів — не більше 1%. І смуга — від 30 до 128 кбіт/с залежно від кодека.

Відеоконференція. Затримка трохи більша — до 200 мс (відео менш чутливе, ніж голос). Джитер той самий — 30 мс. Втрати — 1%. Але смуга — від 1 до 8 Мбіт/с — на два порядки більша, ніж для голосу.

Критичні дані — банківські транзакції, керуючі системи, telemetry SCADA. Затримка — до 300 мс. Джитер — не критичний (транспорт TCP компенсує). Втрати — не більше 0,1% — найжорсткіший показник у таблиці.

Веб-трафік — звичайний браузинг. Затримка до 500 мс — понад це користувачі починають скаржитись. Втрати до 5% (TCP компенсує повторною передачею). Смуга — залежить від потреб.

І передавання файлів — найтерплячіший вид трафіку. Затримка некритична, втрати до 5% — все терпить. Хоче максимум смуги, коли вона доступна.

Тепер розберемо ліву картку внизу — 4 складові затримки. Це важливо, бо не всі складові можна змінити. Перша — серіалізація: час на «виштовхування» усіх бітів пакета в канал. Пакет 1500 байт на 10 Мбіт/с — це 1,2 мілісекунди. На гігабіті — 12 мікросекунд. Тобто на повільних каналах серіалізація суттєва. Друга — поширення: швидкість світла в оптоволокні приблизно 5 мікросекунд на кілометр. Київ-Львів — 550 км × 5 мкс = близько 2,75 мс тільки на поширення. Третя — обробка: час на маршрутизацію в кожному вузлі (сучасні маршрутизатори — кілька мікросекунд). І четверта — черга: очікування пакета у чергах маршрутизатора. І ось саме тут — «QoS тут!» — саме на цю складову

впливають механізми QoS. Три перші складові — фізика й реалізація, їх майже не змінити. Але правильна пріоритизація в чергах може зменшити або збільшити цю складову в десятки разів.

Права картка — реальні цифри для VoIP-кодеків. G.711 — 64 кбіт/с — це PCM (Pulse Code Modulation), еталонна якість, стандарт для PSTN (публічної телефонії). G.729 — 8 кбіт/с — стиснення в 8 разів, ціна — трошки нижча якість, але цілком прийнятна. Opus — гнучкий сучасний кодек, від 6 до 510 кбіт/с, використовується в WebRTC, Zoom, Discord, WhatsApp. І HD-відеоконференція — від 4 до 8 Мбіт/с.

Слайд 8. Дві моделі QoS: IntServ та DiffServ

Тепер про архітектурні підходи до реалізації QoS. Історично склалися дві принципово різні моделі: IntServ і DiffServ. Обидві показані на рисунку 1.8 з посібника.

Перша модель — IntServ, Integrated Services, стандартизована у RFC 1633. Це потік-орієнтований підхід. Ключова ідея: перед тим, як відправити чутливий трафік (наприклад, VoIP-виклик), пристрій-джерело надсилає спеціальне повідомлення протоколом RSVP — Resource Reservation Protocol. Це повідомлення проходить через КОЖНИЙ маршрутизатор на шляху від джерела до одержувача, і в кожному з них резервує ресурси для конкретно ЦЬОГО потоку. Кожен маршрутизатор має пам'ятати про кожен резервований потік.

IntServ визначає два класи обслуговування. Guaranteed Service — жорстка гарантія затримки: провайдер гарантує, що затримка не перевищить певної межі. Controlled Load — м'якіша гарантія: якість, порівнянна з нерозвантаженою мережею.

Плюс IntServ — дуже сильні гарантії, до жорсткої затримки на конкретний потік. Але є критичний недолік, показаний у нижній картці: не масштабується. У маршрутизаторі має зберігатися стан для КОЖНОГО потоку. У корпоративній мережі з десятками потоків — прийнятно. У операторській магістралі з мільйонами одночасних потоків — неможливо. Тому IntServ використовується локально, в невеликих мережах для критичних сервісів; в операторських мережах — не застосовується.

Друга модель — DiffServ, Differentiated Services, стандартизована в RFC 2474 та 2475. Це клас-орієнтований підхід — принципово інша філософія. Замість того, щоб пам'ятати про кожен потік, DiffServ ділить весь трафік на обмежену кількість КЛАСІВ (до 64, за 6-бітним полем DSCP), і кожен маршрутизатор обробляє пакети за їх класом — а не за потоком.

Ключова концепція DiffServ — PHB, Per-Hop Behavior. Це стандартизована поведінка обробки пакета на кожному вузлі. Про це — на наступному слайді.

Головна перевага DiffServ, показана в правій картці — масштабується. Стан у маршрутизаторі — не за потоком, а тільки за класом. Тому в ядрі — усього кілька класів, а не мільйони потоків. Саме через це DiffServ став де-факто стандартом у сучасних операторських мережах.

Слайд 9. DiffServ у деталях: DSCP та PHB

Розберемо DiffServ детальніше. Ліворуч на слайді — структура поля DSCP. DSCP — Differentiated Services Code Point — це 6 бітів у полі ToS (Type of Service) або DS (Differentiated Services) байті IP-заголовка. Історичний контекст: у оригінальному RFC 791 (1981) це поле називалося ToS і мало іншу структуру — IP Precedence. У 1998 в RFC 2474 його перевикористали для DSCP: 6 бітів на DSCP плюс 2 біти на ECN — Explicit Congestion Notification.

6 бітів дають нам 2 в 6-му степені = 64 можливі значення класів. У реальності використовується значно менше — приблизно 8-10 стандартних класів у типовій корпоративній мережі.

Маркування виконується на КОРДОНІ мережі — так званий trust boundary, «межа довіри». Це може бути перший маршрутизатор або комутатор, до якого підключений пристрій. Далі кожен маршрутизатор всередині мережі просто дивиться на DSCP і обробляє пакет за відповідним PHB. Це і є принцип DiffServ: класифікація один раз, застосування правил — багато разів вздовж шляху.

Праворуч — чотири класи PHB, визначені стандартом. Default PHB — просто best effort, за замовчуванням для немаркованого трафіку. Expedited Forwarding, EF — DSCP значення 101110 (у десятковій — 46). Це найвищий пріоритет, спеціально для голосу. Маршрутизатори мають обробляти EF-пакети мінімально швидко, з мінімальною чергою.

Assured Forwarding, AF — це ціла родина класів. Позначаються як AFxy, де x — це номер класу (1, 2, 3, 4), а y — рівень пріоритету відкидання (1, 2, 3). У сумі 4 × 3 = 12 значень AF. Ідея — 4 класи важливості, і в кожному 3 рівні того, наскільки легко пакет може бути відкинутий при перевантаженні. Наприклад: AF41 — четвертий клас (найважливіший), перший рівень відкидання (найменш ймовірно бути відкинутим).

І Class Selector, CS — CS0 до CS7 — це 8 значень для зворотної сумісності зі старим стандартом IP Precedence.

Внизу — таблиця з типовим відображенням DSCP → застосунок у корпоративній мережі. VoIP → EF (46). Відеоконференція → AF41 (34). Multimedia streaming → AF31 (26). Транзакційні дані → AF21 (18). Web-трафік і файли — просто Default, значення 0. Це — типовий шаблон, який ви побачите в реальних мережах.

Слайд 10. Механізми управління перевантаженнями

Тепер про конкретні алгоритми, за якими маршрутизатор упорядковує пакети у чергах. Це називається Congestion Management. На слайді — п'ять типових алгоритмів, від найпростішого до найпотужнішого.

FIFO — First In, First Out, «перший прийшов — перший вийшов». Найпростіший підхід: жодних пріоритетів. Хто перший потрапив у чергу, той перший з неї вийде. Плюс — простий, дешевий у реалізації. Мінус — критичний трафік може «застрягти» за великими некритичними файлами. Уявіть чергу на пошту: якщо перед вами людина з мішком посилок — доведеться чекати.

PQ — Priority Queuing. Абсолютний пріоритет. Створюються кілька черг (зазвичай 4: high, medium, normal, low). Черга вищого пріоритету обслуговується завжди першою. Плюс — гарантований пріоритет для важливого трафіку. Мінус — starvation, «голодування». Якщо високопріоритетна черга ніколи не порожня, то низькопріоритетний трафік МОЖЕ НІКОЛИ НЕ ПРОЙТИ. Це серйозна проблема — тому чистий PQ у виробничих мережах уникають.

WFQ — Weighted Fair Queuing, зважене справедливе обслуговування. Ідея — справедливий розподіл смуги між потоками з урахуванням ваг (пріоритетів). Кожен потік отримує гарантовану частку смуги. Плюс — справедливість, без starvation. Мінус — не гарантує жорсткої затримки для голосу, бо навіть високопріоритетний потік може чекати, поки обслуговуються інші.

CBWFQ — Class-Based Weighted Fair Queuing. Розвиток WFQ: замість потоків — класи трафіку. Адміністратор визначає класи (наприклад, «клас голосу», «клас відео», «клас критичних даних») і задає мінімальну гарантовану смугу для кожного. Плюс — контрольовані класи, гарантована смуга. Мінус — все ще не ідеально для голосу, бо голос може чекати обслуговування інших класів.

І, нарешті, LLQ — Low Latency Queuing. Це де-факто стандарт для сучасних мереж з VoIP. Ідея — поєднати переваги PQ і CBWFQ. LLQ додає до CBWFQ ще одну спеціальну ПРІОРИТЕТНУ чергу спеціально для голосу. Голос обслуговується як в PQ — миттєво, з абсолютним пріоритетом. Решта трафіку — за схемою CBWFQ. Але — критично важливо — на пріоритетну чергу застосовується policing, обмеження смуги. Це запобігає starvation, характерному для чистого PQ: якщо голос перевищує задану смугу, надлишок відкидається, а не блокує решту трафіку. Так LLQ дає гарантовану мінімальну затримку для голосу, при цьому інші класи теж отримують свої частки.

Слайд 11. WRED, Shaping, Policing

Крім управління чергами, QoS має ще кілька важливих механізмів. Розглянемо три. На картинці зверху — рисунок 1.9 з посібника, який показує повний ланцюжок обробки пакета в маршрутизаторі: 1) класифікація на вході, 2) маркування, 3) policing на вході, 4) маршрутизація, 5) обслуговування черг (queuing), 6) shaping на виході.

Перший механізм — WRED, Weighted Random Early Detection. Це механізм запобігання перевантаженням. Проблема: коли черга маршрутизатора переповнюється, він просто відкидає нові пакети, які не входять — це називається Tail Drop, «відкидання хвоста». Проблема з Tail Drop: TCP-потоки реагують на втрати сповільненням, але якщо ВСІ TCP-потоки одночасно втратять пакет — вони одночасно сповільняться, і мережа стане недовикористаною. Ця проблема називається TCP global synchronization.

WRED вирішує це так: не чекає, поки черга переповниться, а починає випадково відкидати пакети ЗАЗДАЛЕГІДЬ, коли черга досягає певного заповнення (наприклад, 70%). Причому — з урахуванням пріоритету: низькопріоритетні пакети відкидаються частіше, високопріоритетні — рідше. TCP на це реагує окремими сповільненнями окремих потоків — і мережа працює плавніше. Це і є принцип: краще керовано відкинути кілька пакетів раніше, ніж хаотично багато пакетів пізніше.

Другий механізм — Traffic Shaping, формування трафіку. Використовується для згладжування піків трафіку. Реалізується за алгоритмом Token Bucket — «відро з жетонами». Ідея: пакети можуть проходити, поки в «відрі» є жетони. Жетони поповнюються з постійною швидкістю. Якщо пакетів приходить більше — вони буферизуються, а не відкидаються. Отже, shaping — «делікатний»: він не втрачає пакети, а просто затримує зайві. Ключове: shaping виконується на ВИХІДНИХ інтерфейсах.

Третій механізм — Traffic Policing, обмеження трафіку. Схожий на shaping, але з принциповою різницею: policing не буферизує зайві пакети, а ВІДКИДАЄ їх (або перемарковує з високого пріоритету на низький). Тобто policing — «жорсткий». Використовується для контролю дотримання клієнтом контракту SLA: якщо клієнт купив 10 Мбіт/с — policing на вході в мережу оператора буде обмежувати його саме цією швидкістю. Все, що вище — відкидається або перемарковується як low priority. Policing виконується на ВХІДНИХ інтерфейсах.

Наочна аналогія: shaping — це швидкісний обмежувач у ліфті (плавно виводить всіх). Policing — це бетонний блок на дорозі (хто не вписався у ширину — відлетів).

Слайд 12. QoS у MPLS та SD-WAN

Розглянемо, як QoS реалізовано у двох основних сучасних операторських технологіях — MPLS і SD-WAN.

Ліворуч — QoS у MPLS. У заголовку MPLS-пакета є 3-бітне поле TC (Traffic Class, раніше називалося EXP — Experimental). Ці 3 біти дають 2 в 3-му = 8 класів обслуговування — аналогічно DSCP (тільки менше). Типове мапування: EF з DSCP → TC = 5, AF41 → TC = 4, і так далі.

MPLS Traffic Engineering, MPLS-TE — це потужна технологія, яка дозволяє явно визначати шляхи проходження трафіку через мережу. Тобто адміністратор може сказати: «весь голосовий трафік іде через маршрут А, а веб-трафік — через маршрут Б». Для сигналізації використовується протокол RSVP-TE (розширення RSVP для Traffic Engineering).

І особливо потужна комбінація — DiffServ-Aware MPLS-TE. Це поєднання DiffServ (класифікація трафіку за DSCP) і MPLS-TE (явні шляхи). На LSP-туннелях резервується пропускна здатність окремо для кожного класу: голос отримує пріоритетне обслуговування; бізнес-критичні дані — гарантовану смугу; best-effort — ресурси, що залишилися. Це — найпотужніший QoS-підхід у сучасних мережах.

Праворуч — QoS у SD-WAN. Це принципово інший підхід. SD-WAN використовує Application-Aware Routing — ідентифікацію КОНКРЕТНИХ ЗАСТОСУНКІВ за допомогою Deep Packet Inspection. Тобто не просто «UDP на порту 5060», а «саме Skype», «саме Zoom», «саме Salesforce». Це дозволяє застосовувати індивідуальні політики маршрутизації для кожного застосунку.

Ключова унікальна риса SD-WAN — динамічний вибір каналу. Контролер SD-WAN БЕЗПЕРЕРВНО моніторить стан кожного WAN-каналу — вимірює затримку, джитер, втрати — і динамічно обирає оптимальний канал для кожного застосунку. При погіршенні якості одного каналу (наприклад, Інтернет-канал деградує через шторм) — голосовий трафік АВТОМАТИЧНО переспрямовується на MPLS-канал. Це називається failover за якістю.

SD-WAN також підтримує WAN-оптимізацію: FEC — Forward Error Correction — для зменшення втрат (пере відправка з надлишковими даними для відновлення втрачених пакетів); дуплікацію пакетів для критичних застосунків (той самий пакет надсилається двома каналами — приймає перший, що приходить); оптимізацію TCP для каналів з високою затримкою; стиснення та дедуплікацію даних.

Слайд 13. Практичний приклад: VoIP через WAN

Тепер зведемо все, що ми обговорили, у практичний приклад. Розглянемо шлях VoIP-виклику через WAN, і які QoS-механізми застосовуються на кожній стадії.

У верхньому банері — вимоги для VoIP-виклику з кодеком G.711. Затримка < 150 мс, джитер < 30 мс, втрати < 1%, пропускна здатність приблизно 87 кбіт/с (це кодек 64 кбіт/с плюс IP/UDP/RTP-заголовки).

Нижче — ланцюжок з 6 стадій, показує, що відбувається на кожному етапі шляху пакета.

Стадія 1 — на IP-телефоні. Кінцевий пристрій (VoIP-телефон, софтфон) МАРКУЄ пакет DSCP = EF (46). Це «запит»: «прошу поставитися до мене з найвищим пріоритетом».

Стадія 2 — на комутаторі доступу. Тут — ключова концепція trust boundary, «межа довіри». Питання: чи довіряти маркуванню, яке проставив кінцевий пристрій? Якщо це офіційний корпоративний IP-телефон — довіряти. Якщо це невідомий комп'ютер — не довіряти, перемаркувати як звичайний трафік. Це запобігає ситуації, коли зловмисник поставить EF на своєму торренті і отримає пріоритет над реальним голосом. Комутатор також працює з 802.1p CoS (Class of Service) — 3-бітним пріоритетом у VLAN-тегу.

Стадія 3 — на маршрутизаторі CE (Customer Edge). Класифікація пакета за DSCP → приміщення у пріоритетну чергу LLQ. Голос отримує гарантований пріоритет над іншим трафіком.

Стадія 4 — на вході в WAN. Тут провайдер виконує policing — перевірку відповідності клієнтського трафіку контракту SLA. Якщо клієнт купив, наприклад, 200 кбіт/с гарантованого EF-трафіку, а надсилає 500 кбіт/с — надлишок буде відкинутий або перемаркований на low priority.

Стадія 5 — у ядрі оператора. Тут пакет вже інкапсульований у MPLS. DSCP значення EF мапується на MPLS TC = 5 (стандартне мапування). Далі — DiffServ-Aware MPLS-TE забезпечує пріоритетне обслуговування вздовж всього шляху через ядро.

Стадія 6 — на виході, на іншому кінці мережі. Тут виконується shaping відповідно до контракту клієнта на другому боці — знову перевірка й приведення трафіку у відповідність.

І тільки після цього пакет доходить до IP-телефона одержувача. Ось так QoS з абстракції перетворюється на конкретний ланцюжок правил і механізмів.

Слайд 14. Порівняння традиційних моделей послуг

Зведемо основні моделі мережевих послуг у порівняльну таблицю: Leased Line, MPLS VPN, Internet VPN, SD-WAN. Пройдемо по критеріях.

Гарантія смуги. Leased Line — повна, це головна перевага виділеного каналу. MPLS VPN — SLA-контракт. Internet VPN — немає (best effort — залежить від навантаження публічного Інтернету). SD-WAN — адаптивно, тобто SD-WAN моніторить якість і адаптується.

Затримка. Leased Line — мінімальна. MPLS VPN — низька, стабільна. Internet VPN — змінна. SD-WAN — оптимізується активно.

QoS. Leased Line — не застосовно, канал виділений повністю під клієнта. MPLS VPN — DiffServ-Aware TE, потужний QoS. Internet VPN — обмежена (провайдер не гарантує). SD-WAN — Application-aware, найгнучкіший.

Вартість. Leased Line — найвища. MPLS VPN — висока. Internet VPN — низька. SD-WAN — середня.

Швидкість розгортання. Leased Line — повільна (місяці, треба проектувати й прокладати фізичний канал). MPLS VPN — середня (тижні, треба узгоджувати з провайдером). Internet VPN — миттєва (просто конфігурація на існуючому Інтернет-каналі). SD-WAN — швидка (дні).

Типовий сценарій. Leased Line — критично важливі з'єднання (банки, оборона). MPLS VPN — корпоративні WAN. Internet VPN — з'єднання філій, які вже мають Інтернет-канал. SD-WAN — гібридні мережі, коли клієнт хоче використовувати одночасно Інтернет, MPLS і мобільний зв'язок.

Головні тренди сучасності, показані у нижньому банері. Перший — MPLS поступово поступається місцем SD-WAN. Причина — вартість: SD-WAN на дешевому Інтернеті часто дає якість, порівнянну з MPLS, при значно нижчій вартості. Другий тренд — виділені канали залишаються, але тільки для критично важливих з'єднань: банкінг, оборона, дата-центри. Третій тренд — гібридні мережі стають нормою: MPLS + Інтернет + LTE/5G одночасно, а SD-WAN керує ними всіма як одним ресурсом.

Слайд 15. Підсумок лекції

Підбиваємо підсумки. Шість ключових ідей сьогоднішньої лекції.

Перше: три традиційні моделі мережевих послуг — Leased Lines (виділені канали), MPLS VPN (ізоляція трафіку на MPLS-інфраструктурі), Managed Services (керовані мережеві послуги) — фундамент операторського бізнесу.

Друге: хмарні моделі IaaS, PaaS, SaaS — визначаються п'ятьма характеристиками NIST. IaaS — максимум контролю (для IT-адмінів), SaaS — мінімум складності (для бізнес-користувачів).

Третє: XaaS — Everything as a Service — метапоняття, під яке підпадають NaaS, SECaaS, UCaaS, SASE. SLA формалізує гарантії доступності (від 99,9% до 99,999%), продуктивності, підтримки і компенсації.

Четверте: чотири базові метрики QoS — bandwidth, delay, jitter, packet loss. Для голосу критичні цифри: затримка < 150 мс, джитер < 30 мс, втрати < 1%. Це — те, що треба запам'ятати назавжди.

П'яте: дві принципові моделі QoS. IntServ (RFC 1633) — потік-орієнтований підхід з RSVP-резервуванням; не масштабується. DiffServ (RFC 2474/2475) —

клас-орієнтований, з DSCP-маркуванням і PHB-обробкою; де-факто стандарт у сучасних мережах.

Шосте: конкретні механізми QoS — LLQ (пріоритетна черга для голосу + CBWFQ для решти), WRED (запобігання перевантаженням через раннє випадкове відкидання), shaping (згладжування піків на виході) і policing (відкидання надлишку на вході).