ЛЕКЦІЯ 4

Архітектура міських мереж Metro Ethernet

та подвійне тегування (QinQ, PBB)

Слайд 2. Пригадуємо лекцію 3 та план лекції 4

План до 4 лекції такий: почнемо з трирівневої архітектури Metro Ethernet — доступ, агрегація, ядро. Потім розглянемо, які транспортні технології використовуються в Metro Ethernet — від найпростішого Native Ethernet до сучасного EoMPLS. Далі перейдемо до центральної теми лекції — проблеми масштабованості VLAN і її вирішення. Розглянемо QinQ, або подвійне тегування 802.1ad: як побудований фрейм, які режими обробки існують, у чому переваги. Потім — наступний рівень масштабування, PBB (802.1ah) з інкапсуляцією MAC-in-MAC. І у другій половині лекції — механізми захисту кільцевих топологій: стандарт ERPS з ITU-T G.8032, який замінив STP у операторських мережах, і технологія Resilient Packet Ring (RPR, IEEE 802.17).

Слайд 3. Трирівнева архітектура Metro Ethernet

Розглянемо трирівневу архітектуру Metro Ethernet. Це де-факто стандарт побудови сучасних міських мереж операторського класу. Ліворуч на слайді — типова схема: три рівні один над одним, кожен виконує свою специфічну функцію.

Найнижчий рівень — рівень доступу (access). Це саме те місце, де до мережі оператора підключається обладнання клієнта. Термінологія: клієнтське обладнання називається CE — Customer Edge; воно підключається до пристрою оператора PE — Provider Edge — через інтерфейс UNI. На цьому рівні комутатори виконують кілька важливих функцій: класифікують клієнтський трафік, додають зовнішні S-VLAN теги (це саме та технологія QinQ, яку ми детально розглянемо далі), застосовують bandwidth profiles відповідно до умов SLA і виконують базову QoS-обробку. Типова пропускна здатність портів рівня доступу — від 10 мегабіт до 10 гігабіт на секунду. Часто на цьому рівні використовується кільцева топологія — вона зручна для міст, бо волоконно-оптичні кабелі природно прокладаються навколо кварталів. Захист таких кілець забезпечує технологія ERPS, яку ми окремо розглянемо.

Середній рівень — рівень агрегації (aggregation). Тут об'єднується трафік від багатьох комутаторів рівня доступу і транспортується до ядра. На цьому рівні виконуються операції складнішого рівня: агрегація EVC, міжвузловий OAM, детальніша QoS-обробка. Саме тут переважно застосовується EtherChannel для збільшення пропускної здатності — на діаграмі ви бачите пунктирну лінію з підписом «EtherChannel» між двома зеленими комутаторами. За потреби на цьому рівні використовується selective QinQ для того, щоб трафік різних сервісів клієнта міг оброблятися по-різному.

Верхній рівень — ядро (core). Це високошвидкісний магістральний транспорт між вузлами агрегації. Тут використовуються найшвидші канали — 10, 40 або 100 Gigabit Ethernet, часто з повнозв'язною топологією для максимальної відмовостійкості. Обробка трафіку на цьому рівні мінімальна — просто швидка передача. Ядро оперує лише кількома S-VLAN або, у випадку PBB, магістральними ідентифікаторами B-VID.

Ключова властивість цієї ієрархії — вона забезпечує масштабованість. Зростання кількості клієнтів на рівні доступу не впливає на складність обробки на рівні ядра, бо ядро оперує лише обмеженою кількістю S-VLAN. Кожен рівень оптимізовано під свою специфічну функцію: рівень доступу — для гнучкої класифікації і QoS, агрегація — для ефективного об'єднання трафіку, ядро — для високошвидкісного транспорту з мінімальною затримкою.

Слайд 4. Транспортні технології Metro Ethernet

Тепер розглянемо, які саме транспортні технології використовуються в Metro Ethernet. На слайді три картки, що показують еволюцію підходів — від найпростішого до найсучаснішого.

Перший підхід — Native Ethernet, «нативний Ethernet». Це найпростіший варіант: Ethernet-фрейми передаються безпосередньо через мережу L2-комутаторів другого рівня, для ізоляції трафіку використовуються VLAN, а для захисту від петель — протоколи STP, RSTP або MSTP. Плюси очевидні: простота і низька вартість — не потрібне складне обладнання. Мінуси — обмежена масштабованість (ті самі 4094 VLAN) і повільна конвергенція STP. Сфера застосування — невеликі мережі або рівень доступу в більших мережах.

Другий підхід — Ethernet over SONET/SDH, скорочено EoS. Це перехідне рішення, яке було особливо актуальним у 2000-х роках, коли оператори мали розгорнуту SDH-інфраструктуру і хотіли поступово переходити на Ethernet без викидання дорогого обладнання. Ідея: Ethernet-фрейми транспортуються через SDH-мережу, як пасажири в SDH-каналах. Плюси — використання наявного TDM-обладнання і його надійність. Мінуси — складна конфігурація, дороге обладнання, TDM-специфіка. Технологічно EoS реалізовано в так званих MSPP-платформах — Multi-Service Provisioning Platforms. Класичний приклад — Cisco ONS 15454 з різними типами карт: G-Series для транспорту Ethernet точка-точка через SONET-канали (від STS-1 до повного Gigabit Ethernet), ML-Series з підтримкою RPR-топології, і так далі.

Третій підхід — сучасний стандарт — Ethernet over MPLS, EoMPLS. Це найбільш масштабоване рішення. Ідея: Ethernet-фрейми інкапсулюються у MPLS-пакети, які й транспортуються через ядро мережі. Це дає одразу кілька переваг: повна ізоляція клієнтських мереж через MPLS VPN — і L2VPN, і L3VPN; гарантії QoS через MPLS Traffic Engineering; швидке відновлення

через MPLS Fast Reroute за менш ніж 50 мілісекунд. Мінуси — складна інженерія і необхідність MPLS-обладнання, що дорого. Три конкретні технології, які тут використовуються: VPWS — Virtual Private Wire Service — реалізує E-Line через pseudowire між двома PE-маршрутизаторами; VPLS — Virtual Private LAN Service (RFC 4761 та 4762) — реалізує E-LAN; і найсучасніший варіант — EVPN, Ethernet VPN (RFC 7432), який поступово витісняє VPLS завдяки використанню BGP для контрольованого вивчення MAC-адрес і підтримки active-active multihoming.

Слайд 5. Проблема 4094 VLAN

Спочатку — про проблему, яку вирішує технологія QinQ.

Пригадаємо стандарт IEEE 802.1Q. Він додає до Ethernet-фрейму спеціальний тег — так званий VLAN-тег, довжиною 4 байти. Одне з полів цього тегу — VID, VLAN Identifier. Воно має довжину 12 бітів. Що означає 12 бітів? 2 у степені 12 = 4096 можливих значень. З них два зарезервовані стандартом — 0 і 4095, — тому реально доступно 4094 унікальних VLAN. Це число ви бачите ліворуч на слайді великим червоним шрифтом.

Для звичайної корпоративної мережі 4094 VLAN — це багато. Навіть найбільша корпорація рідко використовує більше кількох сотень VLAN. Але для оператора зв'язку ситуація зовсім інша. Розглянемо реальний сценарій, показаний праворуч.

Уявіть, оператор має 10 000 клієнтів — а це, погодьтесь, зовсім не унікальна ситуація для середнього регіонального оператора. Кожен клієнт має свою власну VLAN-структуру: наприклад, у одного клієнта VLAN 10 для голосу, VLAN 20 для відео, VLAN 30 для даних. У іншого клієнта — теж свої номери VLAN, які можуть перетинатися з номерами першого. Виникають ДВІ фундаментальні проблеми.

Проблема перша: якщо оператор буде виділяти по одному VLAN на клієнта, він фізично не може обслуговувати більше 4094 клієнтів в одному Ethernet-домені. Це просто неприйнятно для великих мереж. Проблема друга — навіть більш каверзна: у клієнтів VLAN-номери ПЕРЕТИНАЮТЬСЯ. Клієнт А використовує VLAN 10, і клієнт Б — теж VLAN 10, але вони абсолютно незалежні. Оператор має якимось чином ізолювати їхній трафік. Як це зробити?

Відповідь показана внизу слайда: технологія QinQ, або подвійне тегування, дає ієрархію VLAN — до 4094 множимо на 4094, приблизно 16,8 мільйона унікальних комбінацій. Це вже цілком достатньо навіть для найбільших операторів. Розберемо, як саме це реалізовано.

Слайд 6. QinQ — структура фрейму 802.1ad

Технологія QinQ формально стандартизована в IEEE 802.1ad. Її ще називають Provider Bridging, VLAN stacking, Double Tagging. А сама назва «QinQ» буквально означає «802.1Q всередині 802.1Q» — і точно описує принцип роботи.

Погляньте на діаграму вгорі слайда — це реальна структура фрейму з подвійним тегуванням. Порівняйте зі звичайним 802.1Q-фреймом: там після MAC-адрес йшов ОДИН тег 802.1Q. У QinQ між MAC-адресами й полем EtherType/Length стоять ДВА теги. Зовнішній називається S-tag — Service VLAN tag, «сервісний VLAN-тег» — це те, що додає оператор. Внутрішній — C-tag — Customer VLAN tag, оригінальний тег клієнта.

Дуже важлива деталь — різниця у полі EtherType. У звичайному 802.1Q тегу EtherType дорівнює 0x8100. У QinQ зовнішній S-tag має особливий EtherType 0x88A8 — це значення визначено стандартом IEEE 802.1ad. А внутрішній C-tag зберігає стандартний EtherType 0x8100. Це розрізнення принципово важливе: воно дозволяє мережевому обладнанню однозначно ідентифікувати фрейм із подвійним тегуванням і правильно його обробляти. Комутатор бачить EtherType 0x88A8 і розуміє: «це QinQ-фрейм, треба обробити S-tag».

Внутрішня структура кожного тегу однакова — 4 байти: 16 бітів TPID (Tag Protocol Identifier — те, що ми щойно обговорили), 3 біти PCP (Priority Code Point, пріоритет за схемою 802.1p — до 8 класів пріоритету), 1 біт DEI (Drop Eligible Indicator — індикатор можливості відкидання при перевантаженні), і 12 бітів VID (VLAN Identifier — той самий, який має 4094 значень).

Тепер про наслідки. Погляньте на чотири картки внизу слайда. Перша: S-tag має EtherType 0x88A8 і несе VID оператора; додається на UNI оператором. Друга: C-tag має EtherType 0x8100 і несе оригінальний VID клієнта. Третя: простір ідентифікаторів зростає до 4094 × 4094 ≈ 16,8 мільйона комбінацій. І четверта — важливий практичний нюанс: розмір фрейму зростає на 8 байтів (два теги по 4 байти замість одного 4-байтного тегу). Стандартний Ethernet-фрейм — 1518 байтів; з одним тегом — 1522; з двома — до 1526 байтів. Це перевищує стандартний MTU і потребує підтримки на всіх інтерфейсах оператора так званих baby giant frames або jumbo frames. Це важливо пам'ятати при налаштуванні: якщо MTU по всьому шляху не збільшено, QinQ-фрейми будуть відкидатися або фрагментуватися.

Слайд 7. Режими обробки QinQ на UNI

QinQ може працювати у двох основних режимах, які відрізняються тим, як призначається S-VLAN ID. Розглянемо обидва — вони мають різне застосування.

Перший — Port-based QinQ, також відомий як basic QinQ. Це простіший варіант. Принцип: усі фрейми, які надходять на порт UNI, отримують однаковий S-VLAN ID, незалежно від того, який C-VLAN у них був. Конфігурація примітивна: адміністратор просто прив'язує певний S-VLAN ID до порту, і все — будь-який фрейм, що входить через цей порт, автоматично отримує зовнішній S-tag з налаштованим ID. Сценарій застосування: один фізичний порт UNI виділений для одного клієнта, і всі VLAN клієнта повинні транспортуватися через мережу оператора під одним S-VLAN. Приклад: клієнт А отримує порт з S-VLAN 100; тоді всі його C-VLAN — 10, 20, 30 і так далі — ідуть під одним S-VLAN 100. Такий режим ідеально підходить для сценарію, коли оператор надає клієнту виділений порт і не втручається в його внутрішню VLAN-структуру.

Другий режим — Selective QinQ, або VLAN-based QinQ. Це гнучкіший варіант. Принцип: S-VLAN ID призначається не за портом, а на основі значення C-VLAN ID вхідного фрейму. Конфігурація складніша — треба задавати правила відображення: «C-VLAN 10 → S-VLAN 100», «C-VLAN 20 → S-VLAN 200» і так далі. Сценарій застосування: один фізичний порт UNI розподіляється на кілька сервісів; трафік клієнта розділяється на різні S-VLAN за типом послуги. Класичний приклад показаний внизу картки: C-VLAN 10-20 клієнта відображаються на S-VLAN 100 (це послуга «Інтернет»), а C-VLAN 30-40 — на S-VLAN 200 (голосовий сервіс VoIP). Це дозволяє оператору застосовувати різні bandwidth profiles і QoS-правила для різних типів трафіку одного клієнта, при цьому все відбувається через один фізичний порт.

На практиці Port-based QinQ використовується частіше — він простіший і надійніший. Selective QinQ — це більш просунуте рішення для складніших мультисервісних сценаріїв.

Слайд 8. Порівняння: звичайний VLAN vs QinQ

Тепер зведемо все, що ми дізналися про QinQ, у порівняльну таблицю зі звичайним 802.1Q. Це Таблиця 5.1 з навчального посібника. Пройдемо по рядках коротко.

Кількість тегів: у звичайному 802.1Q — один C-tag. У QinQ — два, S-tag плюс C-tag. EtherType: у 802.1Q — 0x8100. У QinQ — зовнішній 0x88A8, внутрішній 0x8100. Простір VLAN ID: 4094 проти 16,8 мільйона комбінацій. Призначення: сегментація локальної мережі проти ізоляції клієнтів оператора. Прозорість: у 802.1Q клієнт бачить VLAN оператора (якщо вони спільно налаштовуються); у QinQ клієнт повністю ізольований від VLAN-структури оператора. Збільшення розміру фрейму: +4 байти для одного тегу, +8 байтів для двох. Масштабованість: обмежена (4094) проти високої (мільйони). І сфера застосування: корпоративні LAN проти операторських MAN і WAN.

Головна ідея, яку я хочу підкреслити, показана у нижньому банері: QinQ — це не якийсь принципово новий протокол. Це той самий VLAN-механізм, який ви добре знаєте з 802.1Q, але з подвоєною ієрархією тегування. Це дало нам зростання масштабу приблизно у 4000 разів, а сама концепція залишилася знайомою і зрозумілою.

Слайд 9. PBB (802.1ah) — MAC-in-MAC

QinQ вирішує проблему кількості VLAN, але у великих мережах виникає інша проблема, показана в лівій картці внизу слайда. У великій мережі MAC-таблиці магістральних комутаторів стають надзвичайно великими, бо вони мають вивчати MAC-адреси УСІХ клієнтських пристроїв з усіх VLAN. Уявіть: 10 000 клієнтів, у кожного в середньому по 50 пристроїв — це вже пів мільйона MAC-адрес, і магістральний комутатор має все це зберігати в пам'яті. Це створює обмеження масштабованості QinQ у великих мережах.

Рішення цієї проблеми — технологія Provider Backbone Bridging, стандартизована в IEEE 802.1ah, також відома як MAC-in-MAC. Ідея показана в центральній картці: інкапсуляція. Весь клієнтський Ethernet-фрейм, включаючи його MAC-адреси, інкапсулюється в новий фрейм з магістральними MAC-адресами — вони називаються B-MAC, Backbone MAC.

Що це дає? Магістральні комутатори тепер бачать лише B-MAC граничних пристроїв оператора — вони називаються BEB, Backbone Edge Bridge — а не тисячі клієнтських MAC. Це кардинально зменшує розмір MAC-таблиць у магістралі. Тепер магістраль оператора «не знає» про клієнтські MAC — вона знає тільки про свої власні прикордонні пристрої.

Права картка показує нові ідентифікатори, які вводить PBB. B-VID — Backbone VLAN Identifier, 12 бітів. Це ідентифікатор магістральних VLAN — VLAN, які використовуються тільки в ядрі оператора. І другий, значно цікавіший — I-SID, Service Instance Identifier. Він має 24 біти! Це дає нам 2 у степені 24 ≈ 16 мільйонів сервісних інстанцій. Порівняно з 4094 у звичайному 802.1Q — це в 4000 разів більше; а порівняно з 16 мільйонами комбінацій QinQ — приблизно стільки ж, але вже як окрема ієрархія на рівні магістралі, не як комбінація тегів.

Верхня картинка показує ієрархію: 802.1Q — один тег; 802.1ad (QinQ) — два теги; 802.1ah (PBB) — повна інкапсуляція фрейму у фрейм з B-DA, B-SA, B-TAG (з B-VID), I-TAG (з I-SID), і всередині — оригінальний клієнтський фрейм цілком.

На практиці PBB використовується в дуже великих операторських мережах і в архітектурах великих дата-центрів. Для середніх операторів достатньо QinQ. Для найбільших — PBB або (частіше) MPLS EVPN.

Слайд 10. Еволюція масштабованості VLAN stacking

Цей слайд узагальнює, куди прийшла мережева інженерія у відповідь на зростання масштабу мереж. Три покоління VLAN stacking у порівнянні.

Ліворуч — IEEE 802.1Q, «дідусь» усіх VLAN. Ємність — 4094 VLAN. Один тег C-VLAN. Стандарт корпоративних LAN. Масштаб — корпоративні мережі.

У центрі — IEEE 802.1ad, QinQ з подвійним тегуванням. Ємність — приблизно 16,8 мільйона комбінацій S-tag + C-tag. Забезпечує ізоляцію клієнтів на межі UNI. Масштаб — оператори середнього масштабу.

Праворуч — IEEE 802.1ah, PBB з інкапсуляцією MAC-in-MAC. Ємність — приблизно 16 мільйонів сервісних інстанцій через I-SID. Клієнтські фрейми інкапсулюються у магістральні. Масштаб — великі оператори та дата-центри.

Кожен наступний крок цієї еволюції додавав новий рівень абстракції. Спочатку — простий VLAN. Потім — ієрархія двох тегів для ізоляції клієнтів. Далі — повна інкапсуляція для приховування клієнтських MAC. На практиці всі три технології сьогодні співіснують у мережах операторів — на рівні доступу часто використовується просто 802.1Q, на кордоні між клієнтом і оператором додається QinQ, а у магістралі найбільших мереж може застосовуватися PBB або сучасніша альтернатива — MPLS з технологіями VPWS, VPLS, EVPN, які ми вже коротко згадували на слайді 4.

Слайд 11. ERPS (G.8032) — кільцевий захист

Тепер переходимо до другої великої теми лекції — захист топологій від відмов. Оскільки для міських мереж природною топологією є кільце (через існуючу топологію волоконно-оптичних кабелів), нам потрібен спеціальний механізм захисту саме для кілець.

Класичний STP — Spanning Tree Protocol, IEEE 802.1D — для цього не підходить. Погляньте на велику цифру ліворуч: час перемикання ERPS менше 50 мілісекунд. Для порівняння: класичний STP при відмові каналу перебудовує топологію 30-50 секунд! Це різниця в тисячу разів. Для оператора зв'язку така повільна конвергенція неприйнятна — за 30 секунд клієнти телефонують у службу підтримки, підписуються скарги, порушується SLA.

Тому Міжнародний союз електрозв'язку — ITU-T — розробив спеціальний стандарт для Ethernet-кілець операторського класу — G.8032. Технологія називається Ethernet Ring Protection Switching, скорочено ERPS. Її час перемикання — менше 50 мілісекунд, що відповідає стандартам «carrier grade» і збігається з часом переключення класичних SDH/SONET APS.

Як працює ERPS? Три основні принципи, показані у правій панелі. Перше: у кільці визначається спеціальний канал, який називається RPL — Ring Protection Link. У нормальному режимі роботи цей канал НАВМИСНО

заблокований. Це запобігає утворенню петлі — адже кільце фізично замкнене, і якби всі канали були активні, кадри б безкінечно циркулювали в кільці. Тобто у нормальному режимі кільце фактично працює як лінійна топологія з розривом у точці RPL.

Друге: для координації всіх вузлів кільця між собою використовується протокол R-APS — Ring Automatic Protection Switching. Це спеціальні керуючі кадри, які постійно надсилаються по кільцю і повідомляють вузли про стан каналів. Якщо десь відбувається відмова — R-APS-повідомлення миттєво розлітаються по кільцю.

Третє: при виявленні відмови ERPS деблокує RPL і одночасно ізолює канал, який відмовив. Трафік переспрямовується через нову топологію — і все це відбувається протягом менш ніж 50 мілісекунд. Клієнти навіть не встигають помітити.

У нижній панелі виділено чотири факти: стандарт — ITU-T G.8032; тип топології — кільцева (одиночна або кілька взаємопов'язаних кілець); клас — carrier grade; порівняння — приблизно в 1000 разів швидший за STP. І ключове повідомлення внизу: 50 мілісекунд — це фактично стандарт carrier grade. Без досягнення цього значення Ethernet-мережа не може вважатися повноцінним замінником SDH/SONET у транспортних мережах операторів зв'язку.

Слайд 12. RPR — Resilient Packet Ring

Ще одна технологія, спеціально розроблена для кільцевих топологій — Resilient Packet Ring, RPR. Її стандартизовано в IEEE 802.17. Історично RPR була важливою альтернативою SDH-кілець для передачі даних, хоча сьогодні поступово поступається місцем іншим технологіям.

Головна особливість RPR, показана в лівій картці, — механізм Spatial Reuse, «просторове перевикористання смуги». Що це означає? У класичних SDH-кільцях (наприклад, UPSR — Unidirectional Path-Switched Ring) трафік від вузла А до вузла Б проходить через ВСЕ кільце — навіть якщо Б знаходиться на сусідньому вузлі. Це неефективно: сегменти кільця «за» точкою призначення все одно займаються цим трафіком. У RPR — інакше: фрейми, адресовані вузлу в кільці, ВИДАЛЯЮТЬСЯ з кільця цим вузлом. Тобто якщо А надіслав фрейм до Б, і Б — сусідній вузол, то далі сегменти кільця вільні і можуть нести зовсім інший трафік одночасно. Це дозволяє мати значно кращу ефективність використання смуги, ніж у SDH UPSR.

Друга ключова особливість — швидке відновлення менше 50 мілісекунд, що показано у правій картці. RPR використовує два механізми захисту. Перший — wrapping, буквально «обмотування». При відмові каналу у точці відмови трафік автоматично перенаправляється на зворотне кільце (RPR — це два кільця, які працюють у протилежних напрямках). Другий механізм —

steering, «керування напрямом». Тут на вхідному вузлі вибирається оптимальний напрям — за годинниковою стрілкою чи проти — щоб оминути точку відмови. Wrapping — швидший (менше 50 мс), steering — оптимальніший з точки зору використання смуги.

Практичний контекст, показаний у нижньому банері. Кільцеві топології типові для міських мереж завдяки існуючій топології волоконно-оптичних кабелів у місті. Класична реалізація RPR — Cisco: ML-Series карти платформи ONS 15454 підтримують RPR-топологію. Ці карти дозволяють будувати стійкі Ethernet-кільця поверх існуючої SONET-інфраструктури. Але сьогодні RPR поступається місцем іншим технологіям: ERPS (простіший і дешевший) і MPLS/EVPN (більш масштабовані і універсальні). У новому проєкті ви навряд чи побачите RPR; але в легасі-мережах, побудованих у 2000-х, вона ще зустрічається, тому знати про неї варто.

Слайд 13. Порівняння протоколів захисту від петель

Ця таблиця систематизує всі механізми захисту від петель, з якими можна зустрітися в мережах різного масштабу. Давайте пройдемо швидко по рядках.

STP — Spanning Tree Protocol, IEEE 802.1D. Час перемикання 30-50 секунд. Топологія — будь-яка. Сфера — класичні LAN, вважається застарілим.

RSTP — Rapid STP, IEEE 802.1w. Швидша версія STP: кілька секунд замість десятків секунд. Топологія — будь-яка. Сфера — сучасні корпоративні LAN.

MSTP — Multiple STP, IEEE 802.1s. Дозволяє мати окремі інстанції STP для різних груп VLAN. Час — кілька секунд. Топологія — з множинними VLAN. Сфера — оператори з QinQ, коли треба керувати STP окремо для різних клієнтів.

ERPS — те, що ми щойно детально розглядали. ITU-T G.8032. Час — менш ніж 50 мс. Топологія — кільцева. Сфера — Carrier Ethernet, де-факто стандарт. Виділено помаранчевим у таблиці, бо це і є найважливіше.

RPR — IEEE 802.17. Час — менш ніж 50 мс. Топологія — кільце. Сфера — Metro Ethernet поверх SDH, легасі-технологія.

І, нарешті, EtherChannel — IEEE 802.1AX. Час failover — мілісекунди. Топологія — point-to-point (з дублюванням каналів). Сфера — агрегація каналів. Про це ми детально говорили минулого разу.

Практичні рекомендації, показані у нижньому банері. Для кільцевих Metro Ethernet мереж — ERPS з G.8032. Для мереж з QinQ і множинними VLAN — MSTP. Для точка-точка з дублюванням каналів — EtherChannel з LACP. І критично важливе: STP у сучасних операторських мережах НЕ використовується через занадто повільну конвергенцію.

Слайд 14. Типовий стек операторської Metro Ethernet

Передостанній змістовий слайд — цілісна картина. Тут показано, як усі технології, які ми розглянули за дві останні лекції (третю і четверту), поєднуються в реальній операторській мережі Metro Ethernet.

На рівні доступу — тобто там, де клієнти підключаються — використовуються: клієнтські VLAN за 802.1Q; оператор додає S-tag за 802.1ad QinQ; для захисту кільцевої топології застосовується ERPS. Тобто на UNI ми зустрічаємо всі три технології одночасно.

На рівні агрегації — де об'єднується трафік від багатьох комутаторів доступу — використовується EtherChannel (з протоколом LACP за 802.1AX) для збільшення пропускної здатності між комутаторами. Може застосовуватися Selective QinQ для більш детального управління клієнтським трафіком. І MSTP — для управління STP-топологією окремо для різних клієнтів (різних S-VLAN).

У ядрі, у core — використовується EoMPLS у різних варіантах: VPWS для E-Line, VPLS або EVPN для E-LAN. Для великих мереж — PBB з інкапсуляцією MAC-in-MAC. І FRR — Fast Reroute — забезпечує швидке відновлення в MPLS-мережах, аналогічно ERPS для Ethernet-кілець.

Головна ідея, показана в нижньому банері: Metro Ethernet — це не одна технологія, а інтеграція цілого стеку: QinQ, EtherChannel, ERPS, MPLS та RPR — все це разом працює як єдина операторська архітектура.

Слайд 15. Підсумок лекції

Підбиваємо підсумки. Шість ключових ідей 4 лекції. Перше: трирівнева архітектура Metro Ethernet — Доступ, Агрегація, Ядро — це ієрархічна модель, що забезпечує масштабованість операторської мережі. Друге: три транспортні технології — Native Ethernet (простий, для невеликих мереж), Ethernet over SDH (перехідне рішення), Ethernet over MPLS (сучасний варіант для великих мереж).

Третє: проблема 4094 VLAN — 12 бітів VID катастрофічно мало для оператора з тисячами клієнтів. Четверте: QinQ, стандарт 802.1ad, вирішує цю проблему подвійним тегуванням — S-tag (0x88A8) плюс C-tag (0x8100), дає 16,8 мільйона комбінацій; існує в двох режимах — port-based і selective.

П'яте: PBB — IEEE 802.1ah, MAC-in-MAC інкапсуляція, вводить нові ідентифікатори B-VID і I-SID (24 біти → 16 мільйонів сервісів), вирішує проблему великих MAC-таблиць у ядрі. Шосте: захист кільцевих топологій — ERPS (стандарт ITU-T G.8032, менше 50 мс, де-факто стандарт для Carrier Ethernet) і RPR (IEEE 802.17, з механізмом spatial reuse, легасі-технологія).