ЛЕКЦІЯ 5

Послідовні канали зв'язку у WAN

та протоколи канального рівня: HDLC і PPP

Слайд 2. Пригадуємо лекцію 4 та план лекції 5

Спочатку коротко пригадаємо, що ми розглянули на минулій лекції. Тема була — «Архітектура міських мереж Metro Ethernet та подвійне тегування». Ми розібрали трирівневу архітектуру Metro Ethernet: доступ, агрегація, ядро. Три транспортні технології: Native Ethernet, EoS (через SDH-інфраструктуру), EoMPLS (найсучасніший варіант). Розглянули проблему обмеження 4094 VLAN у стандарті 802.1Q і як її вирішує QinQ через подвійне тегування — стандарт 802.1ad, який дає до 16,8 мільйона комбінацій S-tag + C-tag. Далі — PBB, стандарт 802.1ah з інкапсуляцією MAC-in-MAC та 24-бітним I-SID для 16 мільйонів сервісних інстанцій. І завершили механізмами захисту кільцевих топологій: ERPS з ITU-T G.8032, що дає час перемикання менш ніж 50 мілісекунд, і Resilient Packet Ring зі spatial reuse.

На сьогоднішній лекції. Ми відходимо від Ethernet і повертаємося до класичних послідовних каналів WAN. План лекції: почнемо з концепції виділених каналів — leased lines — і навіщо вони потрібні. Потім розглянемо фізичну сторону — терміни DTE, DCE, CSU/DSU і стандарти інтерфейсів. Далі перший протокол канального рівня — HDLC: його структуру, типи кадрів, а також розберемо принципову відмінність Cisco HDLC від стандартного ISO HDLC. Потім перейдемо до PPP: чому він з'явився, з яких компонентів складається, як працює його машина станів з 6 фаз, і в чому різниця між двома протоколами автентифікації — PAP і CHAP. Завершимо практичною частиною: покажу приклади реального налаштування у Cisco IOS та команди для моніторингу.

Слайд 3. Виділені канали зв'язку у WAN

Почнімо з питання: що таке виділений канал? У англомовній літературі це називається leased line — «орендована лінія». Ідея дуже проста: оператор надає клієнту у постійне користування канал з фіксованою пропускною здатністю між двома конкретними фізичними точками. І цей канал — не спільний, він виділений саме цьому клієнту, 24 години на добу, 7 днів на тиждень.

У цьому чотири ключові властивості, показані ліворуч на слайді. Перше — смуга гарантована 24/7 незалежно від часу доби і незалежно від того, хто ще щось передає в мережі оператора. Порівняйте зі звичайним Інтернетом: там ваша реальна швидкість залежить від того, скільки сусідів одночасно з вами дивиться Netflix. У leased line — це виключено. Друге — затримка стабільна, з низьким джитером. Знову ж порівняно з пакетними мережами — там затримка може «плавати» від мілісекунд до сотень мілісекунд. Третє — приватність: ваш

трафік НЕ змішується з чужим на рівні комутації. І четверте — фіксована плата: клієнт платить постійно, незалежно від того, чи передав він хоч байт того місяця.

Класичні швидкості виділених каналів, показані внизу лівої панелі. T1 — 1,544 мегабіта на секунду — це американський стандарт, стандартизований ITU-T G.703. E1 — 2,048 мегабіта на секунду — європейський стандарт, використовується в Україні. Далі — T3 та E3 (десятки мегабіт), OC-3 і вище (SONET, від 155 мегабіт). Це історичні, але й досі актуальні швидкості для критичних з'єднань.

Де застосовуються виділені канали. Класичний сценарій — з'єднання головного офісу з філіями. Другий — WAN-канали між маршрутизаторами провайдерів, магістральні лінки. Третій — резервування дата-центрів для реплікації даних 24/7 з гарантованою смугою. І четвертий — критичні системи: банки, платіжні системи (POS-термінали, обмін між банкоматами і центром), урядові мережі — там, де важлива не лише швидкість, а й приватність та передбачуваність.

Слайд 4. DTE, DCE, CSU/DSU — фізичне з'єднання

Тепер про фізичну сторону. На цьому слайді дві схеми — реальна (вгорі) та лабораторна (внизу). Розберемо термінологію Cisco, яку ви зустрічатимете на лабораторній.

DTE — Data Terminal Equipment, буквально «термінальне обладнання даних». Це роутер або комп'ютер — той, хто є споживачем послуги передачі даних. DCE — Data Circuit-terminating Equipment або Data Communications Equipment — «обладнання завершення каналу». Це пристрій, який забезпечує безпосереднє з'єднання з каналом оператора: модем, CSU/DSU (Channel Service Unit / Data Service Unit) для T1/E1, оптичний трансивер тощо. Ключова функція DCE — забезпечувати тактову синхронізацію, тобто задавати частоту передачі бітів (clock rate). Це критично, бо без синхронізації приймач не знає, коли починається і закінчується кожен біт.

У реальній схемі, ланцюжок такий: маршрутизатор R_1 (DTE) підключений до свого CSU/DSU (DCE), звідти через канал оператора (T1/E1) до CSU/DSU (DCE) на іншому боці, далі до маршрутизатора R_2 (DTE). Тактову синхронізацію задає оператор — це його канал.

У прикладі, показано все спрощено. Немає жодних CSU/DSU і жодного оператора — просто два маршрутизатори з'єднані спеціальним кабелем, який називається null-modem cable. Один з них призначається DCE, інший — DTE. Той, хто DCE, задає тактову частоту командою clock rate. Той, хто DTE, «приймає» цю частоту від DCE. Ця конфігурація дозволяє в лабораторії відтворити роботу серіального каналу без реального оператора.

Стандарти інтерфейсів, показані у нижньому банері. V.35 — найпоширеніший стандарт для швидкостей до 2 Мбіт/с, характерний товстий кабель з масивним роз'ємом. X.21 — стандарт ITU-T, синхронний послідовний інтерфейс. EIA/TIA-232 — це той самий RS-232, з яким ви знайомі ще з COM-портів комп'ютера. EIA/TIA-449 і EIA-530 (TIA-530-A) — розвиток RS-232 для вищих швидкостей. І EIA-612/613, більш відомий як HSSI — High-Speed Serial Interface — для швидкостей до 52 мегабіт на секунду, коли вже V.35 не справляється. Вибір кабелю залежить від типу інтерфейсу пристрою і його ролі (DCE чи DTE).

Слайд 5. HDLC — High-Level Data Link Control

Тепер переходимо до першого протоколу канального рівня, який ми розглянемо — HDLC, High-Level Data Link Control. Це фундаментальний бітово-орієнтований протокол, розроблений ISO ще в 1979 році. Формальний стандарт — ISO/IEC 13239, раніше було ISO 3309.

Погляньте на діаграму — це структура кадру HDLC. Кадр починається і закінчується спеціальним байтом Flag зі значенням 0x7E у шістнадцятковій системі, або 01111110 у двійковій. Це фіксована послідовність з шести одиниць, обрамлених нулями. Далі йде поле Address розміром 1 байт — адреса приймача. Поле Control 1-2 байти — воно визначає ТИП кадру та містить порядковий номер. Data — корисне навантаження, розмір змінний. FCS — Frame Check Sequence — контрольна сума CRC, або 16-бітна, або 32-бітна. І знову Flag на завершення.

Одна з класичних технік HDLC — bit stuffing. Проблема: послідовність 01111110 може випадково з'явитися всередині payload. Якби приймач її побачив — він вважав би, що кадр закінчився передчасно. Рішення: після кожних п'яти одиниць підряд у payload передавач ВСТАВЛЯЄ додатковий нуль. Приймач знає про це і при отриманні п'яти одиниць плюс нуля — просто видаляє цей нуль. Так гарантується, що комбінація 01111110 з'явиться лише як Flag.

Розберемо чотири факти внизу слайда. Перший — Flag 0x7E — маркер меж кадру, забезпечується bit stuffing. Другий — три типи кадрів. I-frames (Information) — кадри для передачі даних, мають порядкову нумерацію. S-frames (Supervisory) — керуючі кадри для підтверджень (Receive Ready RR, Receive Not Ready RNR, Reject REJ). U-frames (Unnumbered) — керуючі кадри для встановлення і розриву з'єднання. Третій факт — HDLC бітово-орієнтований, тобто оперує довільними послідовностями бітів, а не байтами. Це принципова відмінність від старіших протоколів SLIP чи від початкового PPP-байтового кадрування. Четвертий факт — HDLC є основою для практично всіх сучасних послідовних протоколів. PPP запозичив у нього формат кадру.

LAPB (Link Access Procedure, Balanced) — це HDLC-варіант, який використовується у X.25. LAPD — у ISDN. LAPF — у Frame Relay. Всі вони — родичі HDLC.

Слайд 6. Cisco HDLC vs стандартний HDLC

Тепер важливий нюанс, який часто плутає студентів. Коли говорять просто «HDLC», може матися на увазі одне з двох: стандартний ISO HDLC — сумісний з усіма — або Cisco HDLC — пропрієтарна реалізація Cisco. Розберемо різницю.

Ліворуч на слайді — ISO HDLC. Розробник — Міжнародна організація стандартизації ISO, приблизно 1979 рік. Стандарт — ISO/IEC 13239. Особливості: чітка структура кадру за стандартом, підтримка тільки одного протоколу верхнього рівня, без ідентифікатора протоколу в заголовку. І тут виникає серйозна проблема, показана внизу лівої картки. Уявіть: у вас на послідовному каналі одночасно потрібно передавати і IPv4, і IPv6, і, скажімо, IPX. У стандартному HDLC це неможливо. Кадр приходить приймачу — але приймач не знає, що всередині: IPv4? IPv6? IPX? Немає поля, яке б це вказувало. Тому стандартний HDLC — це моно-протокол.

Праворуч — Cisco HDLC. Розробник — Cisco Systems. Особливості: додано 2-байтне поле Type (за аналогією з EtherType у Ethernet-фреймі). Це поле вказує, який протокол верхнього рівня знаходиться в payload. Тому Cisco HDLC підтримує кілька протоколів верхнього рівня одночасно — вирішує ту саму проблему, яку QinQ вирішує для VLAN.

Обмеження Cisco HDLC теж очевидне: він працює тільки між пристроями Cisco. Якщо на іншому кінці — Juniper, Huawei чи Mikrotik — Cisco HDLC не встановиться. Для мультивендорних середовищ вибір один — стандартний PPP.

Практичний факт, показаний у нижньому банері: у Cisco IOS команда encapsulation hdlc — це протокол за замовчуванням на будь-якому послідовному інтерфейсі. Тобто якщо ви не вказали encapsulation ppp, то автоматично буде HDLC (точніше — Cisco HDLC). Ця деталь важлива: у виводі show interfaces ви побачите «Encapsulation HDLC», якщо ви не змінили налаштування.

Слайд 7. Чому HDLC вже мало: перехід до PPP

HDLC — простий і швидкий, але для сучасних мереж його можливостей недостатньо. Проаналізуємо його обмеження і зрозуміємо, чому знадобився PPP.

П'ять ключових обмежень HDLC. Перше — немає автентифікації. Якщо хтось фізично підключиться до каналу (наприклад, зловмисник з ноутбуком у

серверній), він одразу зможе передавати трафік і читати чужий. HDLC не перевіряє, хто на іншому боці.

Друге — стандартний HDLC підтримує тільки один протокол верхнього рівня. Ми про це вже говорили. Cisco це вирішило додавши поле Type, але за рахунок несумісності. Проблема мультипротокольності залишається для чистого ISO HDLC.

Третє — Cisco HDLC пропрієтарний, не сумісний з іншими виробниками. Це створює прив'язку до одного вендора.

Четверте — немає механізму узгодження параметрів каналу. Якщо ви хочете змінити MTU з 1500 на 4470, або увімкнути стиснення, або поміняти CRC з 16 на 32 біти — усе це треба конфігурувати вручну на обох кінцях, і якщо конфігурації відрізняються — канал працює неправильно.

І п'яте — немає перевірки якості каналу. У HDLC відсутні механізми моніторингу типу LQM (Link Quality Monitoring).

Тепер що вирішує PPP. PPP — це Point-to-Point Protocol, стандартизований у RFC 1661 у 1994 році (є ще пізніші RFC 1662 про кадрування, RFC 1994 про CHAP і так далі). Розроблений у IETF як універсальний протокол для будь-яких послідовних каналів. Що дає PPP: перше — вбудована автентифікація за протоколами PAP і CHAP (про них далі). Друге — справжня мультипротокольність через NCP: окремі control-протоколи для кожного мережевого протоколу — IPCP для IPv4, IPv6CP для IPv6, IPXCP для IPX і так далі. Третє — відкритий стандарт IETF, підтримується практично всіма виробниками, тому працює у будь-якій мультивендорній мережі. Четверте — автоматичне узгодження параметрів через LCP: MTU, стиснення, чи потрібна автентифікація, magic number для виявлення петель. І п'яте — моніторинг якості через LQR-пакети (Link Quality Report).

Слайд 8. PPP — стек протоколів (RFC 1661)

PPP — це не один протокол, а сімейство протоколів. Розглянемо цю багатошарову архітектуру на схемі ліворуч. Знизу і догори.

На самому низу — фізичний рівень. Це може бути Serial-канал, ISDN, DSL (наприклад, у PPPoE — PPP over Ethernet, характерному для домашнього Інтернету), SONET. PPP працює поверх будь-якого фізичного каналу типу «точка-точка».

Наступний рівень — HDLC-подібне кадрування. Це формально описано у RFC 1662. Ключове слово тут — «HDLC-подібне». PPP запозичив у HDLC саму структуру кадру: ті самі Flag 0x7E на кордонах, той самий bit stuffing (у синхронних лініях) або byte stuffing (у асинхронних, з escape-послідовностями). Це показує, як HDLC став фундаментом для PPP.

Далі — LCP, Link Control Protocol. Це серце PPP. LCP відповідає за керування самим каналом: встановлення, налаштування, підтримку, розрив. LCP узгоджує параметри — MTU (за замовчуванням 1500 байтів, можна змінити), стиснення (Stac, Predictor, LZS, MPPC), чи потрібна автентифікація і яка саме, magic number для виявлення петель у каналі.

Над LCP — блок автентифікації, опціональний. Три основні протоколи: PAP (Password Authentication Protocol, RFC 1334), CHAP (Challenge Handshake Authentication Protocol, RFC 1994), EAP (Extensible Authentication Protocol, RFC 2284, — це універсальна рамка для сучасних методів автентифікації, використовується у Wi-Fi 802.1X).

Далі — NCP, Network Control Protocols. Це критично важливий шар — саме він забезпечує мультипротокольність. Для кожного мережевого протоколу — свій окремий NCP. IPCP (RFC 1332) — для IPv4: узгоджує IP-адреси на обох кінцях каналу, а також опції на кшталт стиснення Ван-Джейкобсона для TCP-заголовків. IPv6CP (RFC 5072) — для IPv6. IPXCP — для IPX (застарілий, але існує). Це і є секрет: коли PPP-канал встановлюється, LCP спрацьовує один раз, а потім по одному NCP запускається для кожного мережевого протоколу, який планується використовувати.

І, нарешті — самі мережеві протоколи: IPv4, IPv6, IPX, AppleTalk. Це те, що PPP переносить, будучи протоколом канального рівня.

Слайд 9. Машина станів PPP — 6 фаз

Тепер розберемо, як саме встановлюється PPP-з'єднання. Це — машина станів з 6 фаз, показана на слайді. Розберемо кожну.

Фаза 1 — Dead. Це початковий стан. Канал фізично неактивний або ще не готовий. PPP чекає сигналу несучої (carrier detect) або відповідного сигналу від фізичного рівня, що канал готовий до роботи.

Фаза 2 — Establish. Активується LCP. Обидві сторони починають узгодження параметрів через LCP Configure-Request повідомлення. Тут вирішується: який MTU буде використовуватися, чи потрібна автентифікація і яка (PAP, CHAP, EAP), яке стиснення, magic number для виявлення петлі. Якщо обидві сторони згодні з усіма параметрами — переходимо далі. Якщо ні — канал не встановиться. Успішне узгодження LCP видно у виводі show interfaces як «LCP Open».

Фаза 3 — Authenticate. Це фаза автентифікації. ВАЖЛИВО — вона опціональна: виконується тільки якщо в LCP-узгодженні було погоджено, що автентифікація потрібна. Якщо ні — одразу переходимо до фази Network. Якщо так — виконується PAP або CHAP (розберемо далі детально). Успішна автентифікація — переходимо далі; неуспішна — канал розривається.

Фаза 4 — Network. Тут запускаються NCP — по одному на кожен мережевий протокол. Для IPv4 запускається IPCP: сторони узгоджують IP-адреси, стиснення TCP-заголовків. Для IPv6 — IPv6CP. Для CDP (Cisco Discovery Protocol) — CDPCP. Кожен NCP завершується у стан «NCP Open». У виводі show interfaces це видно як «Open: IPCP, CDPCP, CCP...».

Фаза 5 — Open. Це робочий стан. Канал працює, дані передаються. Тут PPP просто транспортує пакети мережевого рівня — IPv4, IPv6, тощо.

Фаза 6 — Terminate. Розрив з'єднання. Одна зі сторін надсилає LCP Terminate-Request; інша відповідає Terminate-Ack. Після цього — знову Dead.

У нижньому банері зведено ключові моменти. У фазі Establish обмін LCP Configure-Request/Ack. У фазі Authenticate PAP надсилає ім'я+пароль у відкритому вигляді, а CHAP — challenge, потім хеш-відповідь, потім результат. У фазі Network — IPCP узгоджує IP-адреси. І у виводі show interfaces після успішного встановлення ви побачите обидва рядки: «LCP Open» і «Open: IPCP, ...».

Слайд 10. Автентифікація: PAP vs CHAP

Тепер про два стандартні протоколи автентифікації у PPP. Це PAP і CHAP. Різниця між ними — принципова, з точки зору безпеки.

Ліворуч — PAP, Password Authentication Protocol, стандарт RFC 1334. Позначено як «НЕБЕЗПЕЧНО» — і зараз побачите, чому. Механізм PAP дуже простий, всього два повідомлення. Крок 1: клієнт (той, хто автентифікується) надсилає серверу своє ім'я та пароль. Ключове слово — пароль передається у відкритому вигляді! Не хешується, не шифрується — прямо як є. Крок 2: сервер відповідає — Authenticate-Ack (успішно) або Authenticate-Nak (відмова).

Мінуси PAP очевидні: пароль йде відкритим текстом. Будь-хто, хто прослуховує канал (з sniffer'ом типу Wireshark), одразу отримає пароль. Автентифікація виконується тільки при встановленні каналу і не повторюється — тобто якщо зловмисник перехопить сесію після автентифікації, він може використовувати її як завгодно. Вразливий до replay-атак — можна записати автентифікаційний обмін і відтворити його.

Тому PAP рекомендується використовувати лише для сумісності зі старим обладнанням, яке не підтримує CHAP. У виробничих мережах — PAP уникати.

Праворуч — CHAP, Challenge Handshake Authentication Protocol, стандарт RFC 1994. Позначено як «БЕЗПЕЧНО». Механізм CHAP — три повідомлення, це знаменитий алгоритм challenge-response. Крок 1: сервер надсилає клієнту випадкове число, яке називається challenge. Крок 2: клієнт бере challenge, з'єднує його зі своїм паролем, обчислює MD5-хеш від цієї суми, і надсилає лише хеш серверу. Крок 3: сервер локально обчислює той самий MD5-хеш (він

знає і challenge, і пароль клієнта — пароль зберігається у нього в конфігурації) і порівнює зі своїм. Якщо збігається — Success, ні — Failure.

Головна властивість: пароль НІКОЛИ не передається каналом. Тільки хеш. Навіть якщо зловмисник перехопить хеш — з нього неможливо отримати сам пароль (MD5 — одностороння функція). Плюс — challenge завжди різний (випадкове число), тому replay-атака також неможлива: якщо зловмисник відтворить старий хеш, сервер надішле новий challenge, і старий хеш вже не підійде.

Крім того, CHAP підтримує повторну автентифікацію через випадкові інтервали під час сесії — це запобігає перехопленню вже автентифікованого каналу.

Практична рекомендація: у виробничих мережах — завжди CHAP. У Cisco IOS команда 'ppp authentication chap pap' означає — спочатку пропонується CHAP, якщо інша сторона не підтримує — тоді PAP.

Слайд 11. Налаштування HDLC у Cisco IOS

Тепер практика. Розглянемо, як налаштовується найпростіший випадок — HDLC на послідовному каналі. Ліворуч — 5 обов'язкових кроків, праворуч — реальний приклад з двох маршрутизаторів R_1 і R_2, з методичних рекомендацій Лабораторної роботи №3.

Крок 1 — обрати послідовний інтерфейс. Команда interface Serial 1/0. Формат ідентифікатора може відрізнятися залежно від типу маршрутизатора: іноді просто номер (1), іноді module/number (1/0), іноді slot/module/number (0/1/0).

Крок 2 — для DCE-інтерфейсу задати тактову частоту через команду clock rate. Значення — у бітах за секунду. У прикладі використано 64000 — 64 кбіт/с. Дозволені значення обмежені: 1200, 2400, 4800, 9600, 19200, 38400, 56000, 64000, 72000, 125000, 148000, 500000, 800000, 1000000, 1300000, 2000000, 4000000. Наголошую: на DTE-стороні команда clock rate НЕ виконується — вона або ігнорується, або повертає помилку.

Крок 3 — обрати протокол інкапсуляції канального рівня. Команда encapsulation. Для HDLC — encapsulation hdlc. Але, як я вже говорив, HDLC — за замовчуванням на послідовних інтерфейсах Cisco. Тобто цю команду можна навіть не виконувати — тоді все одно буде HDLC.

Крок 4 — присвоїти IP-адресу. Команда ip address з двома параметрами: адреса та маска. У прикладі — 196.1.1.1 з маскою 255.255.255.252, тобто /30. Маска /30 — стандарт для послідовних каналів point-to-point: 4 адреси (196.1.1.0 — network, 196.1.1.3 — broadcast, 196.1.1.1 і 196.1.1.2 — корисні для двох сторін).

Крок 5 — активувати інтерфейс командою no shutdown. За замовчуванням інтерфейси у Cisco IOS у стані shutdown — треба їх увімкнути.

Праворуч — повний реальний приклад. Заходимо в enable mode, потім у configure terminal. Далі — interface Serial 1/0. Опис (description) — LINK_TO_R_2, це просто мнемонічний коментар, дуже допомагає при діагностиці. Потім clock rate 64000 — R_1 тут DCE. IP-адреса. no shutdown. І аналогічний конфіг на R_2, але без clock rate (R_2 — DTE).

Це базовий HDLC-конфіг. За хвилину після цих команд канал підніметься, ping почне ходити.

Слайд 12. Налаштування PPP + CHAP у Cisco IOS

Тепер більш складний і, водночас, більш реалістичний приклад — PPP з CHAP-автентифікацією. Обидва конфіги — R_1 і R_2 — на слайді. Розглянемо ліву колонку (R_1), потім праву (R_2), а потім розберемо ключові моменти.

На R_1 ми маємо: команда username R_2 password mypass у режимі глобального конфігурування. Це — критично важлива команда. І тут — найпоширеніша помилка початківців. Зверніть увагу: у команді username ми вказуємо ім'я НЕ своє власне, а СУСІДА. Тобто на R_1 ми пишемо username R_2 — це означає: «я очікую, що на іншому боці представиться пристрій з ім'ям R_2, і пароль в нього має бути mypass». Ця логіка на перший погляд плутана, але вона правильна: коли інший бік автентифікується, він надішле мені своє ім'я, і я маю знати, який пароль йому відповідає.

Далі — interface Serial 1/0, як і раніше. clock rate 64000 — оскільки R_1 у нас DCE. Потім ключова команда — encapsulation ppp — замість HDLC, ми обираємо PPP. І далі ppp authentication chap — ця команда каже: «на цьому каналі використовувати CHAP для автентифікації». IP-адреса й no shutdown — все стандартно.

На R_2 — дзеркальна конфігурація. username R_1 password mypass — знову-таки, ім'я сусіда (R_1), пароль mypass. Далі interface Serial 1/0 (без clock rate — R_2 у нас DTE). encapsulation ppp. ppp authentication chap. IP-адреса. no shutdown.

Ключове правило, яке треба запам'ятати:

Перше: у команді username вказуємо ім'я СУСІДА, не своє. На R_1 → username R_2, на R_2 → username R_1.

Друге: пароль (mypass в нашому прикладі) МАЄ ЗБІГАТИСЯ на обох сторонах. Якщо на R_1 у команді username R_2 pass1, а на R_2 у команді username R_1 pass2 — CHAP не пройде.

Тепер як це працює на практиці. Після виконання цих команд канал переходить у Establish, LCP узгоджує параметри, включно з фактом «потрібна CHAP». Далі R_1 надсилає challenge (випадкове число) в бік R_2. R_2

обчислює MD5-хеш(challenge + mypass) і надсилає лише хеш. R_1 локально обчислює той самий MD5(challenge + mypass) і порівнює. Збігається — успіх. Дзеркально в іншому напрямку: R_2 надсилає свій challenge R_1, і той теж проходить перевірку. CHAP — двосторонній.

Після цього — фаза Network, IPCP узгоджує IP-адреси, і канал переходить в Open. У виводі show interfaces ви побачите: «Encapsulation PPP, LCP Open, Open: IPCP, ...».

Слайд 13. Моніторинг і діагностика

Останній практичний слайд — команди для моніторингу і діагностики послідовних каналів. У таблиці — вісім найважливіших команд.

show controllers serial 1/0 — виводить інформацію про фізичний контролер: тип обладнання, тип кабелю, роль (DCE чи DTE), тактову частоту (якщо ви DCE). Ця команда допомагає при найнижчому рівні діагностики — «чи є взагалі фізичний канал».

show interfaces serial 1/0 — головна команда для перевірки роботи каналу. Показує все: чи піднявся інтерфейс, який протокол інкапсуляції, чи піднявся LCP, які NCP активні, статистику пакетів і помилок, MTU, bandwidth, delay.

show interfaces без параметрів — короткий огляд усіх інтерфейсів пристрою.

show compress — показує стан алгоритмів стиснення, якщо вони налаштовані.

show ppp multilink — показує інформацію про багатоканальні PPP-з'єднання (MLPPP, тема нашої майбутньої 7-ї лекції).

Далі — debug-команди. Вони використовуються при активному пошуку проблем. debug serial interface — трасування подій на інтерфейсі. debug ppp negotiation — покрокове відстеження процесу узгодження LCP і NCP; критично корисно, коли канал не підіймається. debug ppp authentication — показує процес PAP чи CHAP: які challenge надсилаються, які відповіді, чи проходить перевірка.

УВАГА: debug-команди генерують дуже багато виводу і можуть перевантажити процесор маршрутизатора у виробничій мережі. Використовуйте їх обережно, вимикайте одразу після діагностики командою undebug all або no debug all.

У нижньому банері — приклад того, що ви маєте побачити при працюючому PPP-каналі у виводі show interfaces: «Serial1/0 is up, line protocol is up» — обидва рівні (фізичний і канальний) працюють; «Encapsulation PPP» — інкапсуляція встановлена; «LCP Open» — LCP успішно узгоджено; «Open: IPCP, CCP, CDPCP» — активні NCP: IPCP для IPv4, CCP для стиснення, CDPCP для CDP.

Слайд 14. Порівняння: HDLC vs PPP

Зведемо все, що ми обговорили, у порівняльну таблицю HDLC vs PPP. Пройдемо швидко по рядках.

Стандарт: HDLC (Cisco) — пропрієтарний Cisco. PPP — відкритий стандарт IETF (RFC 1661).

Сумісність: HDLC — тільки Cisco ↔ Cisco. PPP — усі виробники.

Автентифікація: HDLC — немає взагалі. PPP — PAP, CHAP, EAP.

Мультипротокольність: у Cisco HDLC — так, через поле Type. У PPP — так, через окремі NCP.

Узгодження параметрів: HDLC — немає, треба вручну. PPP — LCP автоматично узгоджує MTU, стиснення, автентифікацію.

Стиснення: HDLC підтримує лише Stac (за документацією Cisco IOS). PPP — Stac, Predictor, LZS, MPPC — набагато ширший вибір.

Моніторинг якості каналу: HDLC — немає. PPP — LQM (Link Quality Monitoring) з LQR-пакетами.

Складність налаштування: HDLC — мінімальна, оскільки за замовчуванням. PPP — потрібне явне налаштування (encapsulation ppp, автентифікація, тощо).

Використання: HDLC — прості канали Cisco-Cisco, для швидкого тестування. PPP — усі інші випадки, тобто практично завжди у виробничих мережах.

У нижньому банері — практичне правило: усе, що не є Cisco-Cisco з'єднанням — PPP. У сумніві — PPP. Він відкритий, гнучкий, безпечніший. HDLC — це радше історична спадщина, зручна для дефолтної роботи, але серйозно поступається PPP за можливостями.

Слайд 15. Підсумок лекції

Підбиваємо підсумки. Шість ключових ідей сьогоднішньої лекції. Перше: виділені канали WAN — leased lines — це канали з гарантованою смугою 24/7, класичні швидкості T1 (1,544 Мбіт/с) і E1 (2,048 Мбіт/с), для критичних з'єднань. Друге: у фізичному з'єднанні беруть участь DTE (маршрутизатор), DCE (CSU/DSU або модем, задає тактову частоту), і мережа оператора. У лабораторії — DCE-сторона задає clock rate.

Третє: HDLC — базовий протокол канального рівня, стандарт ISO/IEC 13239, бітово-орієнтований, Flag 0x7E на кордонах, три типи кадрів (I, S, U). У Cisco — HDLC за замовчуванням. Четверте: PPP — RFC 1661 — це сімейство протоколів: LCP для керування каналом, NCP для кожного мережевого протоколу (IPCP, IPv6CP тощо), опціональна автентифікація, і HDLC-подібне кадрування знизу.

П'яте: два протоколи автентифікації — PAP (передає пароль у відкритому вигляді, небезпечно) і CHAP (challenge-response з MD5, RFC 1994, безпечно). Практично — завжди CHAP. Шосте: PPP-з'єднання проходить 6 фаз машини станів: Dead → Establish (LCP) → Authenticate (опціонально) → Network (NCP) → Open (робочий стан) → Terminate.