

Spanning Tree Protocol (STP)

Spanning Tree Protocol (STP) - (протокол кістякового (зв'язуючого) дерева) – мережевий протокол, що працює на другому рівні моделі OSI. Заснований на однойменному алгоритмі, розробником якого є «Мама Інтернету» – Радья Перлман.

Основним завданням STP є приведення мережі Ethernet з множинними зв'язками до деревоподібної топології (кістякове дерево), що виключає передачу пакетів по колу. Відбувається це шляхом автоматичного блокування надлишкових в цей час зв'язків для повної зв'язності портів. Протокол описаний в стандарті IEEE 802.1D.



Принцип дії

1. У мережі вибирається один кореневий міст ([англ. Root Bridge](#)).
2. Далі кожен, відмінний від кореневого, міст прораховує найкоротший шлях до кореневого порту. Відповідний порт називається кореневим портом ([англ. Root Port](#)). У будь-якого не кореневого комутатора може бути тільки один кореневий порт.
3. Після цього для кожного сегмента мережі прораховується найкоротший шлях до кореневого порту. Міст, через який проходить цей шлях, стає призначеним для цієї мережі ([англ. Designated Bridge](#)). Безпосередньо підключений до мережі порт моста — призначеним портом.
4. Далі на всіх мостах блокуються всі порти, які не є кореневими та призначеними. У підсумку виходить деревоподібна структура (математичний [граф](#)) з вершиною у вигляді кореневого комутатора.

Основні поняття

- *Pathcost*- вартість лінка в STP;
- *Bpduguard*- BPDU-фільтр;
- *Rootguard*- root-фільтр;
- *Bridge ID*= Bridge priority + MAC;
- *Bridge priority*= vlan xxx + 32 768 (default cost);
- *Cost*- «вартість портів». За замовчуванням дорівнює 32 768 (2 в 15 степені);

- *Hello BPD*U= root ID + bridge ID + cost;
- *Root port*(кореневий порт) — це порт, який має найкоротшу відстань до будь-якого порту кореневого комутатора.
- *Designated port*(призначений порт) — це порт, який має найкоротшу відстань від призначеного комутатора до кореневого комутатора.

Швидкість передачі та вартість шляху

Таблиця знизу показує вартість інтерфейсу в залежності від швидкості передачі.

Швидкість передачі	Вартість (802.1D-1998)	Вартість (802.1t-2001)
4 Mbit/s	250	5000000
10 Mbit/s	100	2000000
16 Mbit/s	62	1250000
100 Mbit/s	19	200000
1 Gbit/s	4	20000
2 Gbit/s	3	10000
10 Gbit/s	2	2000

Важливі правила

1. Кореневим (root-овим) комутатором призначається комутатор з найнижчим BID (Bridge ID)
2. Можливі випадки, коли пріоритет у двох і більше комутаторів буде однаковий, тоді вибір кореневого комутатора (root-a) буде відбуватися на підставі MAC-адреси комутатора, де кореневим (root) комутатором стане комутатор з найменшою MAC-адресою.
3. Комутатори, за замовчуванням, не вимірюють стан мережі, а мають заздалегідь прописані налаштування.
4. Кожен порт має свою вартість (cost) з'єднання, встановлену або на заводі-виробнику (за замовчуванням), або вручну.

Алгоритм дії STP (Spanning Tree Protocol)

- Після включення комутаторів в мережу, за замовчуванням кожен (!) Комутатор вважає себе кореневим (root).
- Потім комутатор починає посилати по всіх портах конфігураційні Hello BPDU пакети раз на 2 секунди.
- Виходячи з даних Hello BPDU пакетів, той чи інший комутатор набуває статусу root, тобто кореня.
- Після цього всі порти крім root port і designated port блокуються.
- Відбувається посилка Hello-пакетів раз на 2 секунди, з метою перешкодження появи петель в мережі.

Порти

- Root Port
- Designated Port
- Non-designated Port
- Disabled Port

Rapid Spanning Tree Protocol (RSTP)

Rapid STP (RSTP) характеризується значними вдосконаленнями STP, серед яких зменшення часу збіжності і вища стійкість. Описаний в стандарті IEEE 802.1w (згодом включено до 802.1D-2004).

Per-VLAN Spanning Tree Protocol (PVSTP)

Per-VLAN STP (PVSTP) відповідно до назви розширює функціонал STP для використання VLAN. У рамках даного протоколу в кожному VLAN працює окремий екземпляр STP. Є пропріетарним розширенням Cisco, згодом став з незначними обмеженнями підтримуватися іншими виробниками (Juniper, Extreme Networks). Споконвічно протокол PVST працював тільки через ISL-транки, потім було розроблено розширення PVST +, яке дозволяло працювати через набагато поширеніші 802.1Q-транки. Існують реалізації, об'єднуючі властивості PVST + і RSTP, оскільки ці розширення зачіпають незалежні частини протоколу, в результаті виходить (в термінології Cisco) rapid-pvst. PVST не сумісно з MSTP і при одночасній роботі пристроїв Cisco з цими протоколами викликає проблеми в мережі, зокрема, відключення downlink'ового порту root'ового MSTP-пристрою. Для блокування PVST на більшості мережевих пристроях інших виробників доводиться створювати MAC фільтр, оскільки в їх BPDU фільтрах пакети PVST невідомі і можуть проходити через ці пристрої навіть при відключених STP.

Multiple Spanning Tree Protocol (MSTP)

Multiple STP (MSTP) є найсучаснішою стандартною реалізацією STP, що враховує всі переваги і недоліки попередніх рішень. Описана в стандарті IEEE 802.1s (згодом включено до 802.1Q-2003). На відміну від PVST +, в якому число примірників сполучного дерева (spanning tree) дорівнює кількості віртуальних мереж, MSTP передбачає конфігурування необхідної кількості примірників незалежно від числа віртуальних мереж (VLAN) на комутаторі. В один примірник MST можуть входити декілька віртуальних мереж. Проте, всі комутатори, які беруть участь у MST, повинні мати однаково сконфігуровані групи VLAN (MST instances), що обмежує гнучкість при зміні конфігурації мережі.

Shortest Path Bridging

Shortest Path Bridging (SPB) або IEEE 802.1aq долає обмеження блокування.

Історія

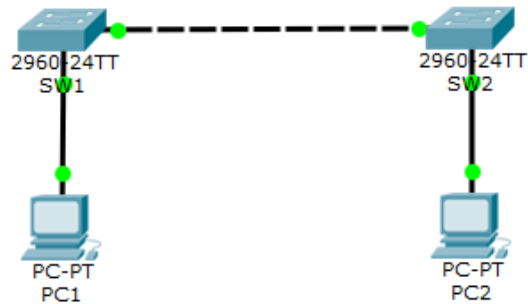
Радья Перлман що винайшла цей алгоритм, також написала короткий вірш що пояснює принципи його роботи:

I think that I shall never see
A graph more lovely than a tree.
A tree whose crucial property
Is loop-free connectivity.
A tree that must be sure to span
So packets can reach every LAN.
First, the root must be selected.
By ID, it is elected.
Least-cost paths from root are traced.
In the tree, these paths are placed.
A mesh is made by folks like me, Then bridges find a spanning tree.

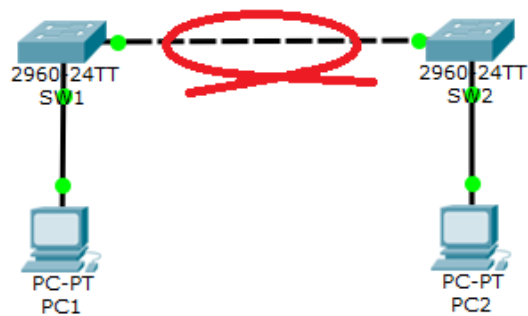
Особливості роботи протоколу зв'язуючого дерева STP

Дізнаємося, як це дерево будуватися, як можна ним управляти, що таке петлі, як з ними боротися.

Починати треба з теорії. Але сенс розбирати протокол, коли не стикався з проблемою, яку цей протокол може вирішити. Тому почнемо з практики та проблем. Далі потрібно розібратися з проблемою і показати, що робити далі. Зберемо звичайну схему.



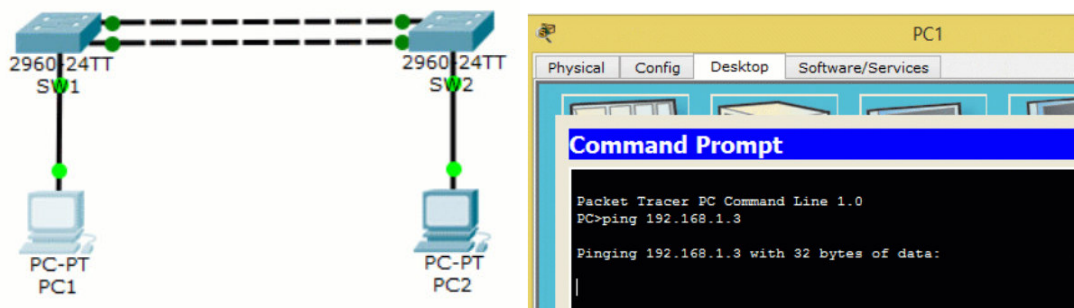
Є 2 комп'ютери та 2 комутатори, підключені один до одного. Адреса PC1-192.168.1.2, а PC2-192.168.1.3. Комп'ютери спілкуються один з одним, щось надсилають один одному. Але ми помічаємо вразливе місце.



Якщо відбудеться обрив кабелю, учасники залишаться без зв'язку. І найперша думка, яка спадає на думку – це встромити ще один кабель. Але перша думка не завжди вірна. На картинках це не показати, тож я покажу це у вигляді анімації.

Комутаційна петля

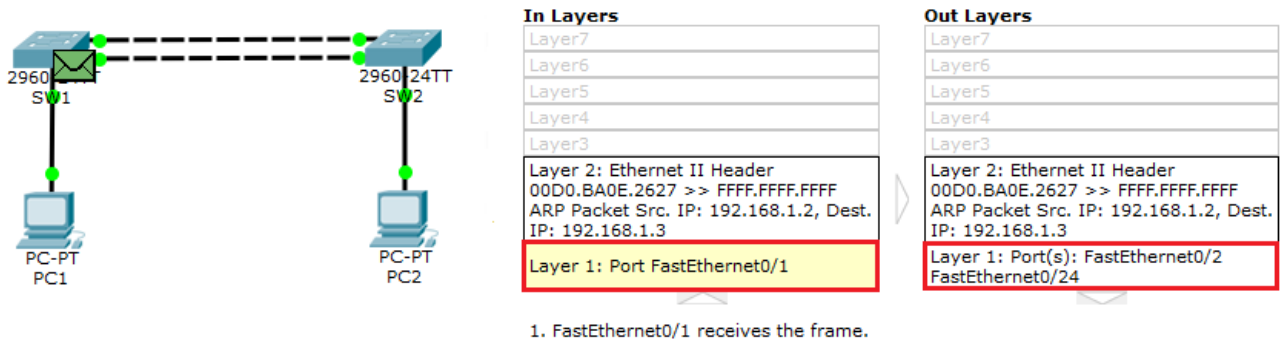
Помітно, як дивно і синхронно заблимали лінки. Це звать петлею. Щоб докладніше ознайомитися з нею, необхідно перейти в режим симуляції.



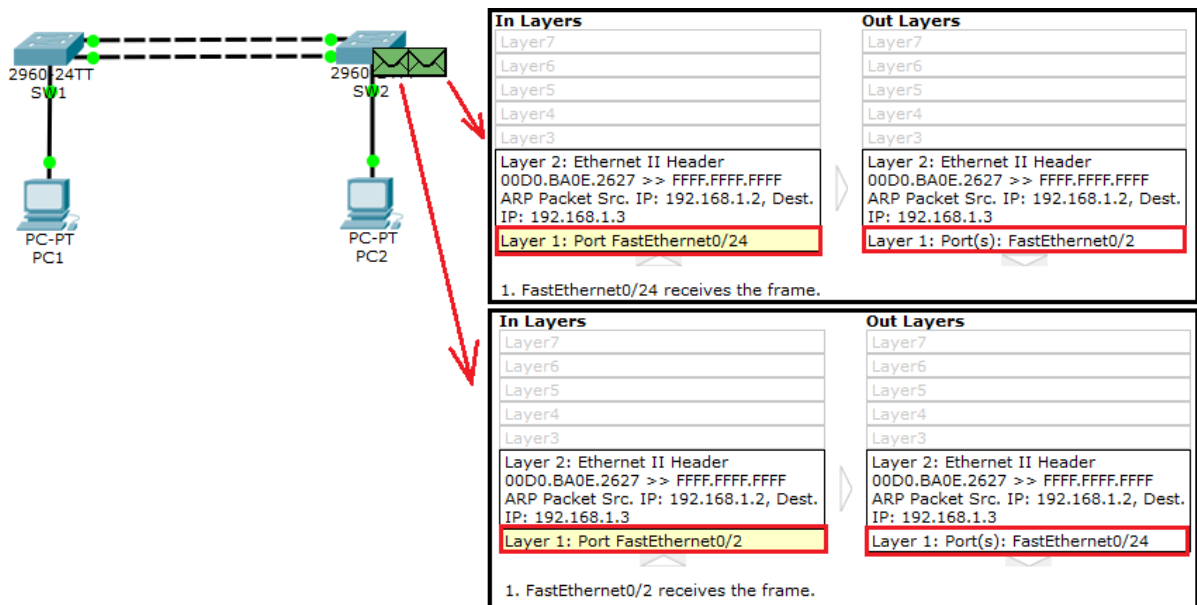
Комутаційна петля в режимі симуляції

Отже, PC1 вирішує надіслати пакет ICMP на комп'ютер PC2. Як правило, перед початком відправки, потрібно дізнатися його MAC-адресу, і він пускає в хід ARP. Згадуємо, як працюють комутатори з ARP. Вони надсилають його на всі порти, крім вихідного. Що відбувається у нас?

(ICMP (Internet Control Message Protocol – міжмережвий протокол керуючих повідомлень) – мережвий протокол, що входить в стек протоколів TCP/IP. В основному ICMP використовується для передачі повідомлень про помилки й інші виняткові ситуації, що виникли при передачі даних. Також на ICMP покладаються деякі сервісні функції, зокрема на основі цього протоколу заснована дія таких загальновідомих утиліт як **ping** та **tracert**.)



Комутатор, за своєю логікою, відправляє ARP на обидва порти (fa0/2 і fa0/24). Але не надсилає його на fa0/1.



SW2 надійде так само. Той ARP, який він отримав із порту fa0/24, він відправить на активний порт fa0/2. А другий ARP, отриманий із порту fa0/2, відправить на fa0/24. Здавалося б, що ми вже отримували з 24 порту ARP. Але тут нюанс. Ми отримували ARP з іншого порту та окремим ARP повідомленням. Тому для комутатора – це 2 різні кадри і обробляються вони незалежно один від одного.

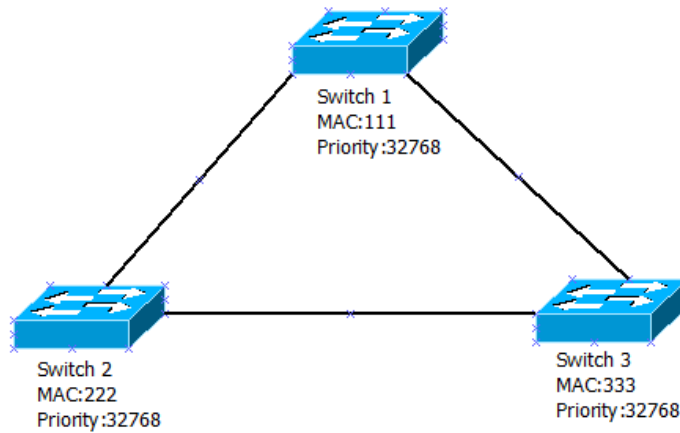
Ну а далі за аналогією. SW2 відправить один із ARP-ів назад на SW1, а той, у свою чергу, назад SW2. І гуляти він буде так до нескінченності, поки не буде висмикнутий кабель або поки комутатор не «захлинеться» кадрами і перестане відповідати. Це і є петля.

Відповідно, чим більше комутаторів, тим більше кадрів буде створено, що призведе до швидкої відмови мережі. Тому **підвищуючи надмірність з'єднань, ми підвищуємо можливість отримання петель.**

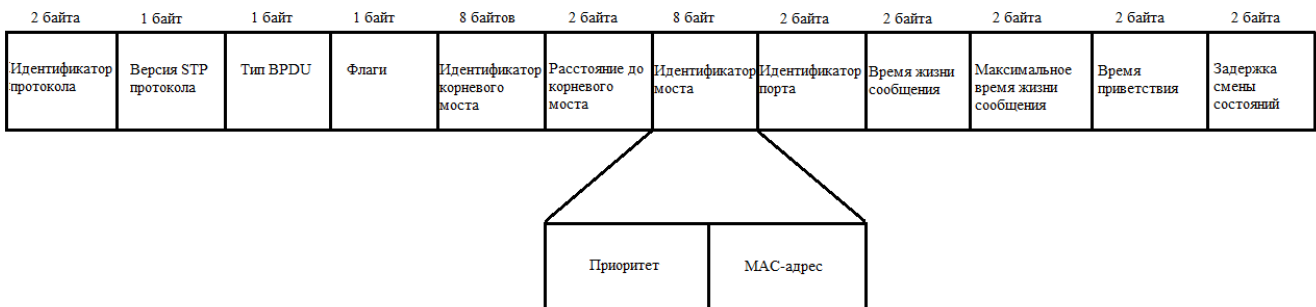
Зрозуміли, що це погано, і з цим треба боротися. Завдання це поклали на плечі видатного інженера Радію Перлман (Radia Joy Perlman) у 1985 році. У чому суть її технології? У вас є N-а кількість комутаторів, з'єднаних один з одним. І перед тим, як передавати дані користувача, вони ведуть переговори між собою на право стати кореневим комутатором або «root switch».

Інші комутатори залишають включеними тільки ті інтерфейси, які ведуть до кореневого комутатора, а решту відключають. Тим самим, до кожного комутатора можна потрапити лише по одному шляху. Розберемо цей процес докладніше.

У нас є 3 комутатори, з'єднані один з одним.



Я умовно дав їм імена та MAC-адреси. На кожному комутаторі включено класичний протокол STP (про інші поговоримо трохи нижче). Як я говорив вище, їм потрібно визначити, хто стане корневим комутатором. Для цього вони починають обмінюватися кадрами BPDU. Подивимося цей кадр.



Дуже багато незрозумілих полів. Ознайомимося із ними.

1. **Ідентифікатор протокола (Protocol Identifier)** – 2-х байтове поле, яке завжди = 0.
2. **Версія STP протокола (Protocol Version Identifier)** – поле розміром в 1 байт, значення якого, завжди дорівнює «0».
3. **Тип BPDU (BPDU type)** – 1 байт, приймає значення «0», якщо це конфігураційний BPDU (CBPDU), та «1», коли TCN BPDU.

- **CBPDU (Configuration Bridge Protocol Data Unit)** – кадр, що використовується для обрахування звязуючого дерева. Тобто, коли значення = 0.
- **TCN BPDU (Topology Change Notification Bridge Protocol Data Unit)** – кадр, який використовується, щоб повідомити інших про зміни в топології. Тобто коли значення = 1. Простіше кажучи, якщо комутатор бачить, що сталося, якась зміна в топології (лінк відвалився, «помер» сусід і т.д.), він пускає BPDU зі значенням «1» у полі BPDU Type. А далі працюють кадри зі значенням «0», щоб наново перебудувати дерево.

4. **Прапорці (Flags)** – у цьому полі використовується лише 1 байт. Ці прапори використовуються при зміні топології (біт «1») та при підтвердженні топології (біт «8»).
5. **Ідентифікатор корньового моста (Root Identifier)** – у цьому полі міститься інформація про кореневий комутатор, а саме його пріоритет і MAC-адресу.
6. **Відстань до корньового моста (Root Path Cost)** – тут міститься сумарна вартість до кореневого комутатора. Наведу таблицьку, де вказано вартість.

Швидкість каналу	Вартість
10 Гбит/с	2
1 Гбит/с	4
100 Мбит/с	19
10 Мбит/с	100

7. Ця таблиця змінена. Раніше ціна вважалася інакше, але з приходом нових стандартів швидкості, попередня специфікація втратила свою силу.

8. **Ідентифікатор моста (Bridge Identifier)** – сюди комутатор-відправник записує свої дані (пріоритет + MAC-адреса).

9. **Ідентифікатор порту (Port Identifier)** – сюди комутатор-відправник записує ідентифікатор порту (тобто той, з якого цей BPDU вийде).

10. **Час життя повідомлення (Message Age)** – тут міститься часовий інтервал (у секундах). Він потрібний для того, щоб розпізнати застарілі кадри та відкинути. Його формує кореневий комутатор і встановлює початкове значення «0». Далі кожен наступний комутатор збільшує це значення під час затримки. Як тільки це значення перевищить максимальне граничне значення, воно буде відкинуте.

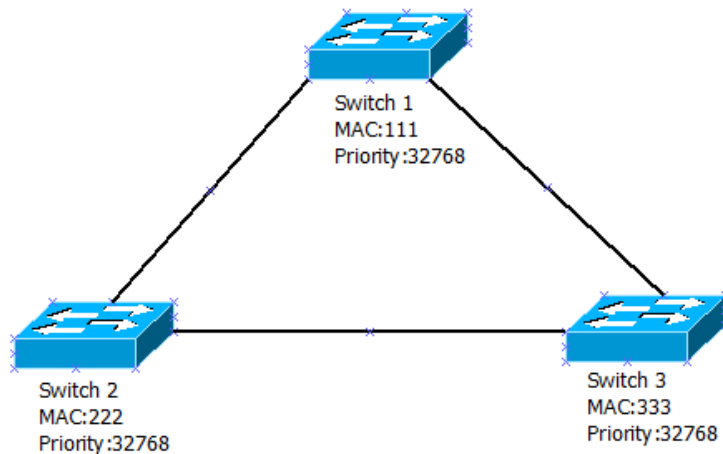
11. **Максимальний час життя повідомлення (Max Age)** – це поле відповідає саме за максимальний час життя. Перевищивши його, комутатор відкидає кадр.

12. **Час привітання (Hello Time)** – Часовий інтервал, через який комутатор надсилає BPDU кадри. За замовчуванням це 2 секунди.

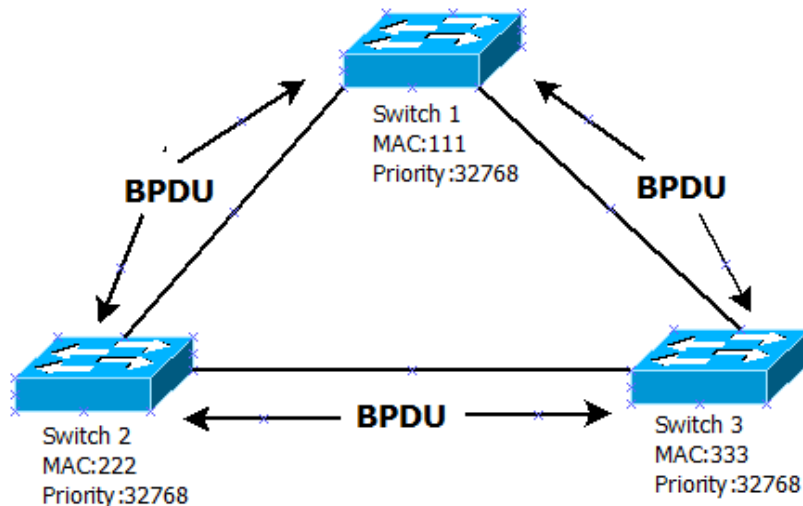
13. **Затримка зміни станів (Forward Delay)** – часовий інтервал, що вказує скільки секунд порт комутатора перебуватиме у стані прослуховування та навчання.

Це стандартний набір BPDU у STP. Залежно від версії поля можуть називатися інакше, але принцип роботи у них єдиний. Сам кадр великий і з ходу може не відразу все вкластися в голову. Це нормальне явище. Знати всі поля напам'ять необов'язково. Головне, що потрібно твердо розуміти в рамках CCNA – це поля **5, 6, 7 та 8**. Тому переходимо до аналізу роботи протоколу STP.

У багатьох виданнях роботу STP показують на прикладі 3 комутаторів, з'єднаних між собою. Зробимо аналогічно.



Умовно дамо їм імена та MAC-адреси, щоб не засмічувати голову довгою адресацією. Рухаємось далі.



І тому що на комутаторах працює протокол STP, їм **потрібно вибрати** того, хто буде головним у топології або кореневим (**root**). Для цього вони починають обмінюватися BPDU-кадрами. Ось тут важливі поля 5, 6 і 7. Я спеціально хочу зупинитися на них.

Спочатку комутатори в полі 5 (Ідентифікатор кореневого мосту або Root Identifier) починають записувати свою «пріоритет + MAC-адресу». Якщо вручну нічого не змінювати, то пріоритет дорівнює 32768. Далі комутатор, який отримує цей кадр від сусіда, порівнюватиме свій «Root Identifier» із новоприбулим. Якщо він побачить, що у сусіда цей Root ID нижче, то з цього моменту він ретранслюватиме його BPDU. В результаті в мережі залишиться лише один комутатор, який генеруватиме BPDU.

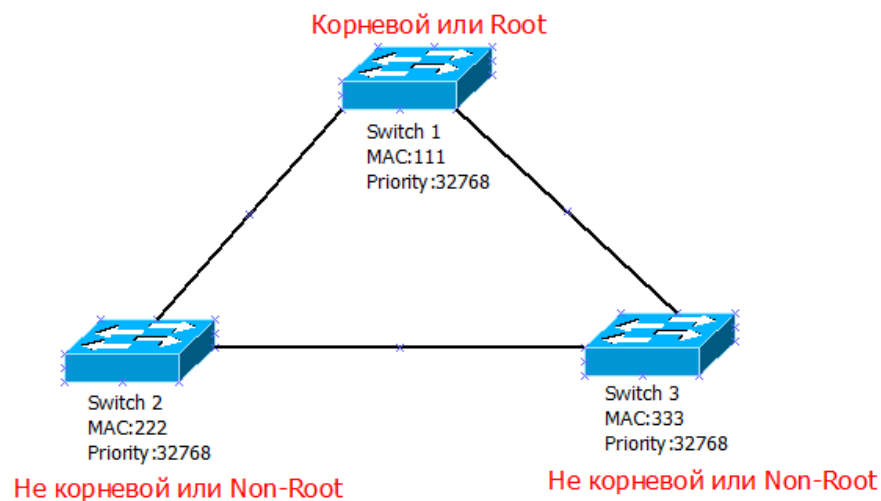
У полі 6 "Root Path Cost" комутатор запише вартість шляху. При створенні BPDU кореневий комутатор записує туди 0, так як це він і є. А ось наступні комутатори вже починають підсумовувати вартість таблиці, представленої вище.

І в полі 7 "Bridge Identifier" записується зв'язка "пріоритет + MAC-адреса" самого комутатора. Тобто, якщо в Root Identifier завжди записується зв'язка кореневого комутатора, то в це поле, він завжди записує свою. Тобто при ретрансляції BPDU від сусіда до сусіда комутатори сюди дописують свій Bridge ID.

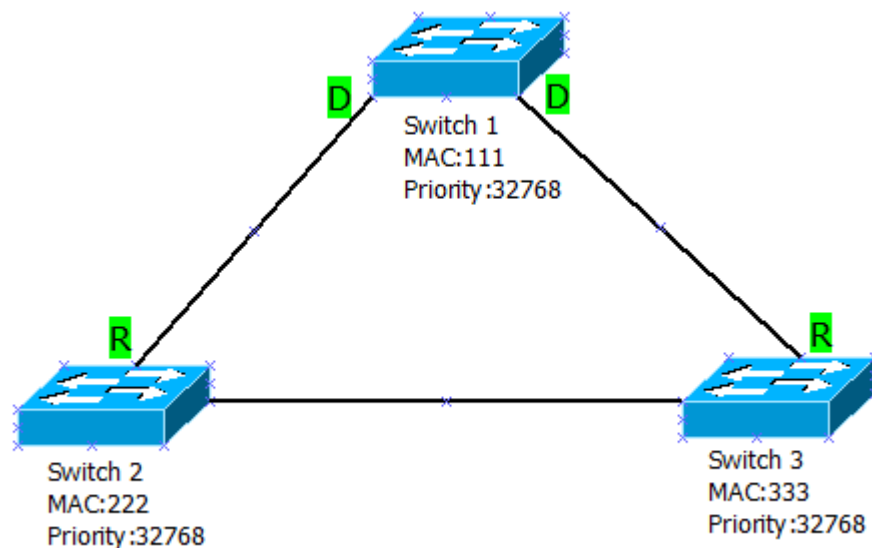
Пару слів про зв'язку «**пріоритет + MAC-адреса**». Вони в жодному разі не сумуються. Знак "+" я вставив у тому контексті, що вони завжди працюють разом. Спочатку комутатори при проведенні виборів дивляться на пріоритет. І якщо пріоритети рівні (а за замовчуванням вони рівні), то починає спиратися на MAC-адреси. І той, у кого MAC-адреса менша, стає головним, кореневим або root. Називайте, як вам зручно. Ось пріоритет потрібен якраз для того, щоб адміністративно впливати на вибір кореневого комутатора.

Уявіть ситуацію, що у вас є 2 комутатора. Один із них новий і продуктивний, а другий старий, стародавній і незабаром піде під списання. І тут з'ясовується, що у старого комутатора MAC-адреса менше, ніж у нового комутатора, а значить, за рівних пріоритетів, вигравати завжди буде старий комутатор. Ось для вирішення такого спірного завдання і потрібний пріоритет.

Причому, коли ви міняєте пріоритет, він має бути кратним 4096 (тобто 32768, 28672, 24576 тощо). Повертаємось до схеми.

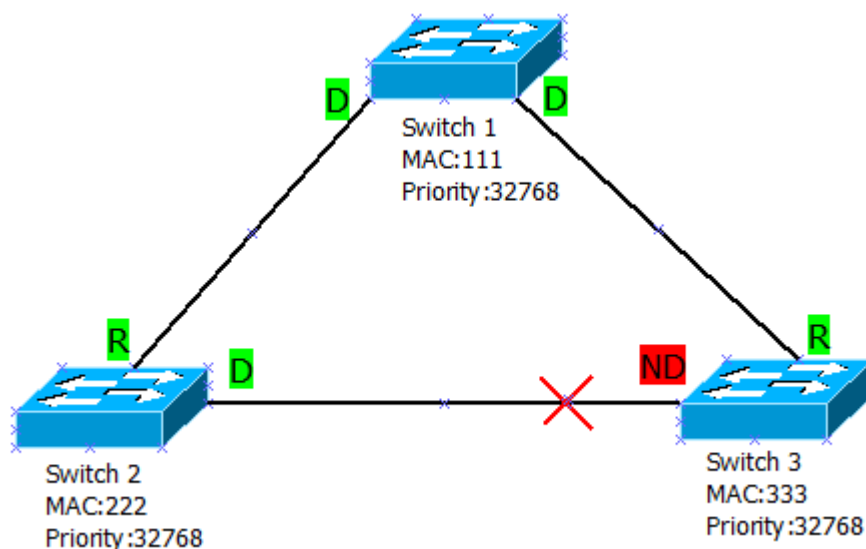


Ну і оскільки пріоритети у трьох комутаторів однакові, то вибори вони починають за MAC-адресами. **Найменша MAC-адреса у Switch 1 => він стає кореневим.**



Раз Switch 1 стає кореневим, він відразу переводить всі свої інтерфейси в режим «**Designated**». Тобто це порт, який має найкоротший шлях до кореневого комутатора (у цьому випадку до себе).

Далі **Switch 2** та **Switch 3** повинні вирішити для себе, який порт буде корневим. Тобто порт, який має найменшу вартість шляху до кореневого комутатора. Тут усе очевидно. Якщо раптом вийде, що вартість кількох портів буде однакова, він вибере порт з найменшим порядковим номером чи ім'ям. Наприклад, з портів fa0/1, fa0/2 та fa0/3, буде обраний fa0/1.



Root-порти визначені, але що робити з лінком між Switch 2 та Switch 3, адже він може створити петлю?! Для її запобігання вони домовляються, хто їх відключить свій порт.

Домовлятимуться вони також за Bridge ID. Пріоритети рівні, тому дивимося на MAC-адреси. У Switch 2 MAC-адреса менше, тому він переводить порт у режим «Designated», а Switch 3 у режим «Non-Designated». "Non-Designated" – такий режим, за якого порту заборонено передавати будь-які дані, але дозволено слухати, що відбувається в мережі. Тобто, якщо відвалиться якийсь лінк, він може увімкнутися та повноправно працювати.

Крім ролей, порти мають стани, які вони повинні пройти в обов'язковому порядку. Поясню з прикладу побудованої топології. Ось у нас збудовано вище дерево STP. Петелі немає і все чудово. Один із портів комутатора Switch 3 знаходиться у стані **Blocking**. Ось він слухає BPDU та нікого не чіпає. Але якщо раптом відвалиться десь лінк або станеться зміна топології, він одразу переходить у стан **Listening** або **Прослуховування**. У цьому стані він відправляє, слухає лише BPDU кадри та обробляє отриману інформацію.

Якщо він бачить, що у сусідів параметри гірші, ніж у нього, то через 15 секунд, переходить у наступний стан **Learning** або **Навчання**. Ця фаза триває також 15 секунд. У «Learning» порт робить практично все те саме, що й у попередньому стані, за винятком того, що тепер буде таблицю комутації на підставі отриманих кадрів.

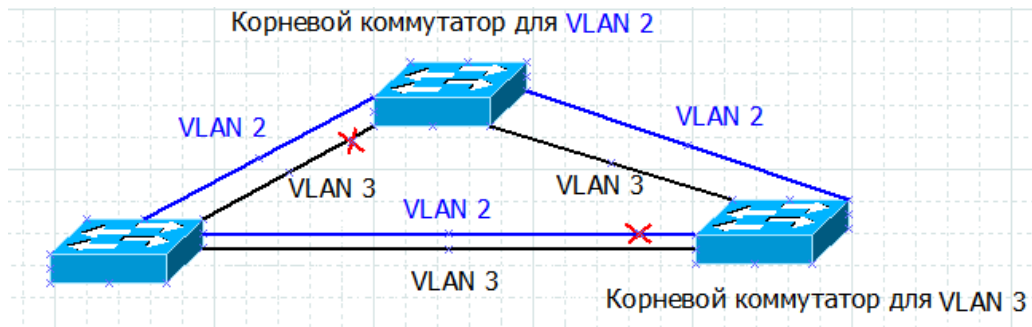
Якщо через 15 секунд, він не отримає BPDU з параметрами краще, ніж у нього, то перейде в останній стан **Forwarding** або **Просування**. Це такий фінальний та повноправний стан. Він обмінюється не тільки службовою інформацією, а й даними користувача. Тобто перехід із стану **Listening** у **Forwarding** триває 30 секунд.

Є ще стан **Disable** або **Вимкнений**, коли вручну відключаєте порт, але я не вважаю, що це стан STP. У цьому стані нічого не передаватиметься. Це, власне кажучи, фізичне відключення порту.

Наведений вище приклад - це робота класичного протоколу STP, який ще називають **CST (Classic Spanning Tree)**. Одним з його мінусів це те, що він буде одне єдине дерево для

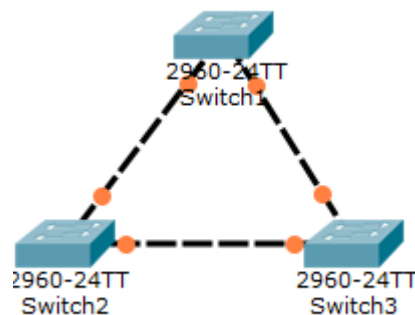
всієї топології. А враховуючи, що з'явилися VLAN, то потрібно було модифікувати цей протокол під них.

Cisco як піонер випустила протокол **PVST (Per-VLAN Spanning Tree)**. Він дозволяв будувати окреме дерево кожного VLAN. Єдине, що він працював із ISL (пропрієтарний цисковський протокол, який працює з тегованими кадрами), який застосовувався лише на пристроях даного виробника. Але з появою відкритого протоколу 802.1q, вони швидко модернізували PVST і дали йому ім'я **PVST+**. Працює він так само, як і його попередник, але з 802.1q. Намалюю схему та поясню докладніше.

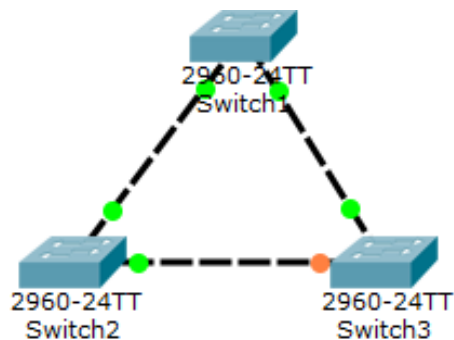


Ось, наприклад, у мене є 2 VLAN. І для кожного VLAN, протокол PVST+ буде окреме дерево. У принципі, це його відмінність від CST. Вибори та переходи проходять аналогічно і з тим самим інтервалом у часі. На жаль, або, на щастя, сучасні Cisco-комутатори вже не підтримують CST.

Тому попрактикуємось на PVST+. Тим більше, що при роботі мережі в одному VLAN-і (який є VLAN за замовчуванням), він мало чим відрізнятиметься від класичного STP. Я вже швиденько зібрав лабораторку з трьох комутаторів і зараз все наочно покажу.

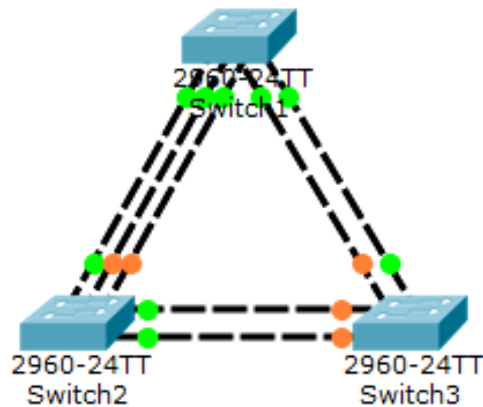


І як тільки комутатори пройшли всі стадії, утворюється **STP-дерево**.

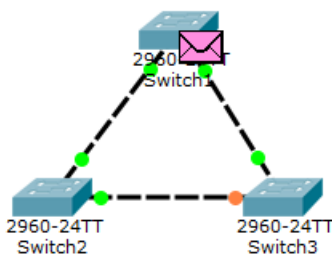


Я думаю ви помітили, що один з портів комутатора Switch3 горить помаранчевим. Це означає, що цей порт перебуває у стані **Blocking**. Не плутайте з **Disabled**. Тобто він не стосується

трафіку користувача, але слухає, що відбувається в мережі. І не важливо, скільки ми встроимо кабелів. У топології завжди відпрацюватиме STP і закриватиме петлі.



Власне, що й показано на рисунку. Тепер покажу, що відбувається з комутаторами, коли дерево вже збудовано. За логікою STP, кореневий комутатор повинен відправляти Hello-кадр «підлеглим» комутаторам з інтервалом часу 2 секунди.



Ethernet 802.3

0	4	7	8	14	19	Bytes
PREAMBLE: 1010 1010		S F D	DEST ADDR: 0180.C200.0000	SRC ADDR: 0060.70B9.0901		
LENGTH / TYPE: 0x3		DATA (VARIABLE LENGTH)			FCS: 0x0	

LLC

0	8	16	24 Bits
DSAP:0x42	SSAP:0x42	CONTROL BIT: 3	

STP BPDU

0	4	8	10	16	24	31Bits
PROTOCOL ID: 0		VERSION: 0		MESSAGE TYPE: 0		
T C	P R	PORT ROLE	L R	F W	A G	T C A
ROOT ID: 32769 / 0009.7C94.0557						
ROOT PATH COST : 0						
BRIDGE ID: 32769 / 0009.7C94.0557						
PORT ID: 32769					MESSAGE AGE: 0	
MAX AGE: 20					HELLO TIME: 2	
FORWARD DELAY: 15						

Що він являє собою, ви бачите на картинці вище. Прошу звернути увагу на поля кадру Ethernet 802.3. А саме "Source MAC-Address" та "Destination MAC-Address". У "Source MAC-Address" він записує MAC-адресу свого порту (в даному випадку FastEthernet 0/1). А в «Destination MAC-Address» мультикастова адреса «0180.C200.0000», яка надсилається всім учасникам, які знають, що таке STP і працюють з ним. Та й сам кадр STP BPDU. Тут купа полів. Але зверну увагу на найважливіші, які я відзначив червоним прямокутником.

- **Root ID** — 32769 / 0009.7C94.0557. Це якраз та сама зв'язка «пріоритет + MAC-адреса». Але тут цікавий момент. Якщо у класичному STP пріоритет за замовчуванням

дорівнював 32768, то тут ми бачимо 32769. У протоколі PVST+ та інших, які вміють працювати з VLAN, до пріоритету додається параметр system ID extension. Цей параметр містить у собі номер VLAN і по ньому комутатор розуміє, до якого процесу STP його віднести. Тобто в даному випадку у нас VLAN №1 => пріоритет = 32768 + 1 = 32769. Якби ми налаштували для десятого VLAN-а, то пріоритет дорівнював 32778. Ну а після дробової риси, сама MAC-адреса інтерфейсу.

- **Root Path Cost** — вартість шляху. Ми вивчаємо кадр, коли він виходить з корньового комутатора => там стоїть 0.
- **Bridge ID** – ідентифікатор комутатора, який ретранслює даний BPDU. В даний момент він такий самий, як і Root ID.
- **Port ID** — ідентифікатор порта. Такий самий, як і пріоритет — 32769.
- **Message Age** — інтервал часу. Оскільки BPDU «свіжий», то там стоїть 0.
- **Max Age** — максимальний час життя – 20 секунд.
- **Hello Time** — інтервал посилки привітання – 2 секунди.
- **Forward Delay** — вказує, скільки секунд знаходиться в одній фазі (прослуховування або навчання) — 15 секунд.

У принципі, тут нічого нового і все це ми розбирали вище. Я це показав, щоб ви розуміли, для чого я так довго вантажив сухим текстом.

Ми вже знаємо, хто є корневим комутатором та який порт заблокований для усунення петлі. Але на іспиті і в повсякденному житті ми оперуватимемо командами, за допомогою яких можна буде дізнатися, хто в сегменті є корневим, у кого заблоковано порт та іншу інформацію. Почнемо з комутатора Switch1 і з найважливішої команди **show spanning-tree**. Її **важливо запам'ятати**.

```
Switch1#show spanning-tree
VLAN0001
Spanning tree enabled protocol ieee
Root ID Priority 32769
    Address 0009.7C94.0557
    This bridge is the root
    Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
    Address 0009.7C94.0557
    Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
    Aging Time 20

Interface    Role Sts Cost    Prio.Nbr Type
-----
Fa0/1        Desg FWD 19      128.1  P2p
Fa0/2        Desg FWD 19      128.2  P2p
```

Ця команда виводить інформацію про всі процеси STP (тобто за кожен VLAN), у яких бере участь комутатор. У нашому випадку лише один VLAN.

Тепер поговоримо, що означають ці записи. Перше, що впадає у вічі — це блок **Root ID**.

```

Root ID    Priority    32769
          Address    0009.7C94.0557
          This bridge is the root
          Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

```

Він містить інформацію про пріоритет, MAC-адресу та таймери кореневого комутатора. Тут красується ще один важливий рядок **This bridge is the root**. Вона говорить про те, що саме цей комутатор є **кореневим за цей VLAN**. Тому, якщо вам треба буде знайти кореневий комутатор, то шукайте цей напис. На сусідньому комутаторі (що не є кореневим) цього рядка не буде.

Наступний блок — **Bridge ID**.

```

Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
          Address    0009.7C94.0557
          Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
          Aging Time 20

```

Тут, відповідно, інформація про поточний комутатор. На кореновому комутаторі цей блок ідентичний попередньому. Нижче розташовується таблиця.

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Desg	FWD	19	128.1	P2p
Fa0/2	Desg	FWD	19	128.2	P2p

У ній записані інтерфейси, які стосуються даного VLAN-у, їх ролі, статуси та інше. Зупинимося трохи на ній.

Оскільки це кореневий комутатор, то порти автоматично перетворюються на роль **«Designated»**.

Статус **Forwarding** говорить про те, що порти пройшли всі стадії і зараз знаходяться в активному режимі (пересилання).

Далі йде вартість і вона дорівнює 19. FastEthernet працює на швидкості 100 Мбіт/с і для цієї швидкості вартість дорівнює 19 (вище наведена табличка).

Далі йде колонка Prio.Nbr або Priority Number. Це пріоритет порту. За замовчуванням цей параметр дорівнює 128 а після точки записується порядковий номер порту. Відповідно для Fa0/1 – це 128.1, а для Fa0/2 – 128.2.

Тип **p2p** говорить про те, що порт комутатора працює в режимі **full-duplex**. Це означає, що порт може одночасно вести і передачу і прийом. Якщо там буде вказано **«shared»**, це означатиме, що порт працює у режимі **«half-duplex»**. Тобто або передає, або отримує (не одночасно).

Перейдемо до наступного комутатора **Switch2**. Аналогічно введу команду **show spanning-tree** і подивлюся, що він покаже.

```
Switch2#show spanning-tree
```

```

VLAN0001
Spanning tree enabled protocol ieee
Root ID Priority 32769
  Address 0009.7C94.0557
  Cost 19
  Port 1(FastEthernet0/1)
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
  Address 00D0.9776.B743
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Aging Time 20

```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/1	Root	FWD	19	128.1		P2p
Fa0/2	Desg	FWD	19	128.2		P2p

Зверніть увагу на блок **Root ID**.

```

Spanning tree enabled protocol ieee
  Root ID Priority 32769
    Address 0009.7C94.0557
    Cost 38
    Port 1(FastEthernet0/1)
    Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

```

Як говорилося раніше, тут міститься інформація про кореневий комутатор. Але тут вже немає напису **This bridge is the root**, тому що цей комутатор не кореневий.

Але є інший запис Port. У ньому вказаний порт, що веде до кореневого комутатора, і це FastEthernet0/1.

Вище є рядок **Cost і він дорівнює 19**. Не плутайте цей рядок Cost з таким самим рядком у таблиці інтерфейсів нижче. Якщо таблиці інтерфейсів вартість зазначена за конкретний порт, тут записується сумарна вартість до кореневого комутатора. Наприклад, якщо за комутатором Switch2 буде ще один комутатор з інтерфейсом FastEthernet, його вартість буде вищою.

```

Spanning tree enabled protocol ieee
  Root ID Priority 32769
    Address 0009.7C94.0557
    Cost 38
    Port 1(FastEthernet0/1)
    Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

```

Тобто він складе вартість свого інтерфейсу із вартістю інтерфейсу сусіда. Рухаємось далі та натикаємось на блок Bridge ID. Сюди записує інформацію про себе. Ви можете помітити, що MAC-адреси відрізняються. Далі йдуть таймери. Це важливий показник і намагайтеся про нього не забувати. Краще його не міняти. Але якщо все-таки виникла потреба це зробити, то міняйте і на сусідніх комутаторах. Інакше це може призвести до серйозних помилок і займе багато часу на усунення.

Таблиця інтерфейсів відрізняється від кореневого комутатора тим, що роль FastEthernet0/1 не "Designated", а "Root". Тобто цей порт веде до кореневого комутатора. Залишився останній комутатор **Switch3**

```
Switch3#show spanning-tree
VLAN0001
Spanning tree enabled protocol ieee
Root ID Priority 32769
    Address 0009.7C94.0557
    Cost 19
    Port 1(FastEthernet0/1)
    Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
    Address 00D0.BA84.7C38
    Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
    Aging Time 20
```

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Root	FWD	19	128.1	P2p
Fa0/2	Altn	BLK	19	128.2	P2p

Тут конфігурація аналогічна, крім порту **FastEthernet0/2**.

```
Fa0/2          Altn BLK 19          128.2    P2p
```

Він у ролі **Alternate**. Тобто як запасний. А статус Blocking свідчить про те, що порт заблокований, щоб обірвати петлю. Ось принцип роботи класичного STP.

Але цей вид вже не дуже актуальний, тому що ви не зустрінете серйозну організацію, яка має лише один VLAN. **Відповідно, наше завдання подружити STP із VLAN**. Тому створюємо VLAN на кожному комутаторі. Можна, звичайно, увімкнути VTP і вони автоматично синхронізуються, але я не прихильник цього протоколу.

Тому в блокноті підготував **шаблон команд, які я вставляю на кожен комутатор**.

Список команд

```
enable
configure terminal
vlan 2
exit
vlan 3
exit
interface range fastethernet 0/1-2
switchport mode trunk
```

І тепер перевірю, що вийшло на виході командою «**show spanning-tree**».

Switch1#show spanning-tree

VLAN0001

Spanning tree enabled protocol ieee
Root ID Priority 32769
Address 0009.7C94.0557
This bridge is the root
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
Address 0009.7C94.0557
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 20

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/1	Desg	FWD	19	128.1		P2p
Fa0/2	Desg	FWD	19	128.2		P2p

VLAN0002

Spanning tree enabled protocol ieee
Root ID Priority 32770
Address 0009.7C94.0557
This bridge is the root
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32770 (priority 32768 sys-id-ext 2)
Address 0009.7C94.0557
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 20

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/1	Desg	FWD	19	128.1		P2p
Fa0/2	Desg	FWD	19	128.2		P2p

VLAN0003

Spanning tree enabled protocol ieee
Root ID Priority 32771
Address 0009.7C94.0557
This bridge is the root
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32771 (priority 32768 sys-id-ext 3)
Address 0009.7C94.0557
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 20

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/1	Desg	FWD	19	128.1		P2p
Fa0/2	Desg	FWD	19	128.2		P2p

Вийшло довге полотно тексту, в якому описано процес STP для кожного VLAN. Якщо уважно подивитися, то можна побачити, що **Switch1 є кореневим для кожного VLAN**. Але не завжди так буває потрібне.

Зараз пояснимо. **Наприклад**, ми маємо Switch3, який блокує порт для усунення петлі. Погляньмо на його оновлену конфігурацію.

```
Switch3#show spanning-tree
```

VLAN0001

```
Spanning tree enabled protocol ieee
Root ID Priority 32769
  Address 0009.7C94.0557
  Cost 19
  Port 1(FastEthernet0/1)
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
  Address 00D0.BA84.7C38
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Aging Time 20
```

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Root	FWD	19	128.1	P2p
Fa0/2	Altn	BLK	19	128.2	P2p

VLAN0002

```
Spanning tree enabled protocol ieee
Root ID Priority 32770
  Address 0009.7C94.0557
  Cost 19
  Port 1(FastEthernet0/1)
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32770 (priority 32768 sys-id-ext 2)
  Address 00D0.BA84.7C38
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Aging Time 20
```

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Root	FWD	19	128.1	P2p
Fa0/2	Altn	BLK	19	128.2	P2p

VLAN0003

```
Spanning tree enabled protocol ieee
Root ID Priority 32771
  Address 0009.7C94.0557
  Cost 19
  Port 1(FastEthernet0/1)
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
```

```
Bridge ID Priority 32771 (priority 32768 sys-id-ext 3)
Address 00D0.BA84.7C38
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 20
```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/1	Root	FWD	19	128.1		P2p
Fa0/2	Altn	BLK	19	128.2		P2p

Бачимо, що він блокує інтерфейс FastEthernet0/2 у всіх трьох VLAN-ах. І ось виникла ситуація, що **потрібно зробити Switch3 корневим комутатором для VLAN 3**. Як описано раніше, на допомогу прийде гра з пріоритетом. Зараз він дорівнює 32 771 (32 786 + 3). Мені треба його зменшити. Зробити це можна кількома способами. **Перший спосіб це задати пріоритет вручну**. Заходжу на Switch 3 та пишу:

```
Switch3(config)#spanning-tree vlan 3 priority 30000
% Bridge Priority must be in increments of 4096.
% Allowed values are:
0 4096 8192 12288 16384 20480 24576 28672
32768 36864 40960 45056 49152 53248 57344 61440
```

Я вирішив задати пріоритет 30000, тому що він менший за 32768. Так, зверніть увагу, що ми змінюємо саме пріоритет без sys-id-ext. Але після введення, виходить повідомлення, що потрібно ввести число, кратне 4096. І нижче пропонує допустимий пріоритет. Можна ввести одне із запропонованих значень, і пріоритет зміниться.

Є інший спосіб змінити пріоритет.

```
Switch3(config)# spanning-tree vlan 3 root primary
```

При введенні цієї команди комутатор дивиться, який Bridge ID був у кореневого комутатора і змінює його на менше значення. Тільки забирає він не 4096, а 8192. Тобто робить менше на 2 порядки. **Введемо цю команду та подивимось, що зміниться.**

```
Switch3#show spanning-tree
VLAN0001
Spanning tree enabled protocol ieee
Root ID Priority 32769
Address 0009.7C94.0557
Cost 19
Port 1(FastEthernet0/1)
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
Address 00D0.BA84.7C38
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 20
```

```

Interface      Role Sts Cost   Prio.Nbr Type
-----
Fa0/1         Root FWD 19    128.1  P2p
Fa0/2         Altn BLK 19    128.2  P2p

```

VLAN0002

```

Spanning tree enabled protocol ieee
Root ID Priority 32770
  Address 0009.7C94.0557
  Cost 19
  Port 1(FastEthernet0/1)
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

```

```

Bridge ID Priority 32770 (priority 32768 sys-id-ext 2)
  Address 00D0.BA84.7C38
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Aging Time 20

```

```

Interface      Role Sts Cost   Prio.Nbr Type
-----
Fa0/1         Root FWD 19    128.1  P2p
Fa0/2         Altn BLK 19    128.2  P2p

```

VLAN0003

```

Spanning tree enabled protocol ieee
Root ID Priority 24579
  Address 00D0.BA84.7C38
  This bridge is the root
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

```

```

Bridge ID Priority 24579 (priority 24576 sys-id-ext 3)
  Address 00D0.BA84.7C38
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Aging Time 20

```

```

Interface      Role Sts Cost   Prio.Nbr Type
-----
Fa0/1         Desg FWD 19    128.1  P2p
Fa0/2         Desg FWD 19    128.2  P2p

```

Бачу, що секція VLAN 3 змінилася. Тепер там пріоритет 24579 (24576 + 3) і красується рядок "This bridge is the root", що вказує, що даний комутатор тепер кореневий для VLAN 3. Обидва порти в ролі "Designated" і статусі "Forward" (що вірно для кореневого комутатора) . Але дві верхніх секції з VLAN залишилися без зміни і для них FastEthernet 0/2 залишиться як і раніше заблокованим.

Тепер побачимо, як відреагував Switch 1.

```

Switch1#show spanning-tree
VLAN0001

```

Spanning tree enabled protocol ieee
Root ID Priority 32769
Address 0009.7C94.0557
This bridge is the root
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
Address 0009.7C94.0557
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 20

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/2	Desg	FWD	19	128.2		P2p
Fa0/1	Desg	FWD	19	128.1		P2p

VLAN0002

Spanning tree enabled protocol ieee
Root ID Priority 32770
Address 0009.7C94.0557
This bridge is the root
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32770 (priority 32768 sys-id-ext 2)
Address 0009.7C94.0557
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 20

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/2	Desg	FWD	19	128.2		P2p
Fa0/1	Desg	FWD	19	128.1		P2p

VLAN0003

Spanning tree enabled protocol ieee
Root ID Priority 24579
Address 00D0.BA84.7C38
Cost 19
Port 1(FastEthernet0/1)
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32771 (priority 32768 sys-id-ext 3)
Address 0009.7C94.0557
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 20

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/2	Desg	FWD	19	128.2		P2p
Fa0/1	Root	FWD	19	128.1		P2p

Бачимо, що він відреагував спокійно. Switch 1, як і раніше, є корневим для VLAN 1 і VLAN 2. І лише для VLAN 3 він змінив свій стан і стани портів.

Ось таким чином можна керувати різними процесами STP для кожного VLAN. Це все звичайно добре, що комутатор перед включенням порту, всіляко все перевіряє ще раз. Але якщо ми знаємо, що за портом комутатора знаходиться клієнтський комп'ютер, який не створить петлі, можна відразу перевести порт в режим «Forwarding», не чекаючи 30 секунд.

Для цього є **технологія Portfast**. Зайду на комутатор Switch2 і продемонструю на прикладі порту FastEthernet 0/3:

```
Switch2(config-if)#spanning-tree portfast
%Warning: portfast should only be enabled on ports connected to a single
host. Connecting hubs, concentrators, switches, bridges, etc... to this
interface when portfast is enabled, can cause temporary bridging loops.
Use with CAUTION

%Portfast has been configured on FastEthernet0/3 but will only
have effect when the interface is in a non-trunking mode.
```

Після введення, він відразу переводить порт у режим Forwarding, але виводить попередження про те, що цей порт повинен суворо підключатися до одного хосту користувача. Інакше, при підключенні комутаторів та інших пристроїв це може призвести до появи петлі. Під спойлером нижче показано, як це працює.

Як бачите, він проминув усі стадії та одразу перейшов до режиму «Forwarding». Не забувайте про цю технологію, але й користуйтеся нею з обережністю, так як опиниться там не хост користувача, а комутатор або інший пристрій, ви ризикуєте створити петлю.

Ось основний принцип PVST+. Як бачите, він мало чим відрізняється від класичного STP чи CST. Я думаю ви помітили, яке полотно тексту виводить команда «show spanning-tree». І що більше VLAN-ів, то більше вписувалося висновок. І якщо вам потрібно буде подивитися інформацію на комутаторі за 10-ий VLAN, то доведеться прокручувати весь висновок із самого початку, поки не дістанетеся до рядка з потрібним VLAN-ом. Для полегшення цієї ситуації є дуже хороша команда, що дозволяє дізнатися інформацію за конкретний VLAN. Це команда **show spanning-tree vlan X**. Перевірю цю команду.

```
Switch1#show spanning-tree vlan 3
VLAN0003
Spanning tree enabled protocol ieee
Root ID Priority 24579
    Address 00D0.BA84.7C38
    Cost 19
    Port 1(FastEthernet0/1)
    Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32771 (priority 32768 sys-id-ext 3)
    Address 0009.7C94.0557
    Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
    Aging Time 20
```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/2	Desg	FWD	19	128.2		P2p
Fa0/1	Root	FWD	19	128.1		P2p

І ось він мені по моїй команді виводить інформацію тільки за третій VLAN. Дуже зручна команда, тому беріть на замітку. Є ще одна цікава команда **show spanning-tree summary**.

```
Switch3#show spanning-tree summary
Switch is in pvst mode
Root bridge for: VLAN0003
Extended system ID      is enabled
Portfast Default        is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default      is disabled
EtherChannel misconfig guard is disabled
UplinkFast              is disabled
BackboneFast            is disabled
Configured Pathcost method used is short
```

Name	Blocking	Listening	Learning	Forwarding	STP Active
VLAN0001	1	0	0	1	2
VLAN0002	1	0	0	1	2
VLAN0003	0	0	0	2	2
3 vlans	2	0	0	4	6

Вона показує сумарну та коротку статистику. У якому STP режимі працює комутатор, для якого VLAN він є кореневим, які функції на ньому включені. І найголовніше, тут є таблиця, що містить імена VLAN-ів і кількість інтерфейсів у цьому VLAN-і, що у різних станах. Це дуже корисно, коли треба швидко зайти і подивитися чи є на комутаторі заблоковані порти і для якого VLAN вони заблоковані. У принципі, з усіх команд ці часто використовуються і для рівня CCNA їх більш ніж достатньо.

Насправді STP та PVST+ не єдині протоколи запобігання петель. Є ще RSTP та MSTP. Якщо MSTP у програмі CCNA практично не згадується, за винятком того, що він є, то про RSTP говорити відкрито і докладно Cisco почала з новою версією програми CCNA 3.0. Тому розберу його докладніше.

Напевно, ви помітили, що класичний STP, що PVST+ вимагають час на збіжність. А саме 30 секунд, при відмові або відключенні будь-якого лінка. Це, звичайно, не так багато, але чим більше мережа, тим більше часу це займає. І у великому корпоративному середовищі повна збіжність може тривати кілька хвилин. І ось для вирішення такої ситуації комітет IEEE випустив стандарт 802.1w або протокол **RSTP**. В чому суть.

Якщо класичному STP було 4 стани (Blocking, Listening, Learning, Forwarding), то RSTP їх поменшало. Всього 3 (Discarding, Listening та Forwarding). Тобто комутатор відкидає, вивчає чи пересилає. Але швидше він сходиться не через це.

Швидку збіжність протокол забезпечує тим, що заздалегідь прораховує, який порт включити, якщо відмовить працюючий. Тим самим, при відмові порту, він не починає судомно вивчати топологію і стрибати по різних станах, а просто переключається на заздалегідь прорахований.

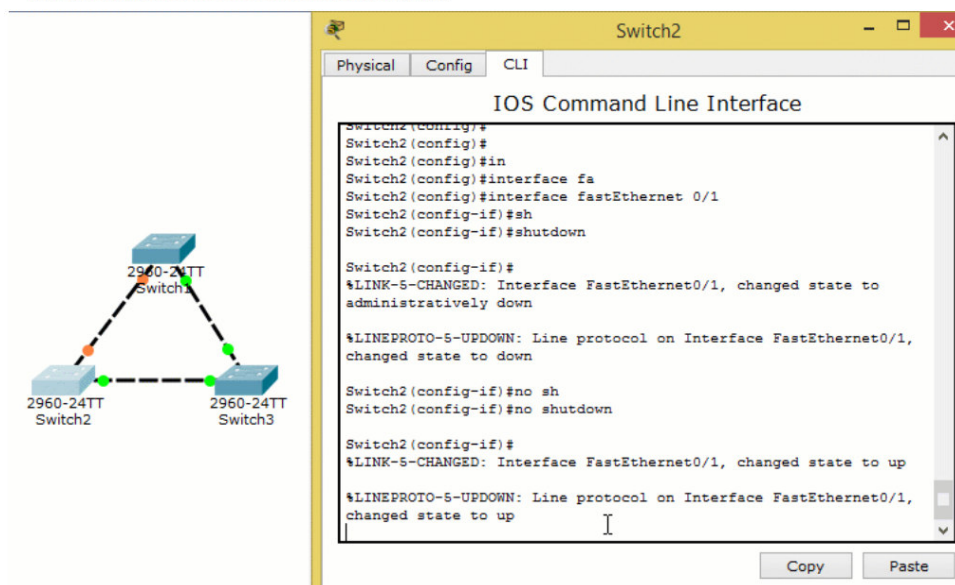
Швидку збіжність та реакцію на відмови в RSTP забезпечують:

- генерація BPDU повідомлень кожним пристроєм незалежно від кореневого комутатора;
- механізм proposal/agreement, при активації каналу «точка-точка», для швидкого переходу портів у стан «Forwarding»
- механізм використання альтернативного порту при втраті зв'язку через кореневий порт (на зразок технології UplinkFast);
- механізм негайної реакції на отримання BPDU з «гіршим» кореневим комутатором від сусіда, що має designated порти для даного сегмента мережі (на зразок технології backbonefast);
- більш оптимальна схема розсилки та обробки повідомлень TCN BPDU про зміни в мережі.

Включити протокол RSTP можна командою:

```
Switch2(config)#spanning-tree mode rapid-pvst
```

Було зібрано мережу, що включали на кожному комутаторі RSTP і перевірив, як швидко відбудеться перебудова дерева. **Перебудова відбувається за лічені секунди.**



Тепер ми можемо будувати процеси STP для кожного VLAN, управляти пріоритетом і багато іншого. А для швидкості збіжності можемо використовувати протокол RSTP.