VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

NÁVRH DATOVÝCH SÍTÍ POSKYTOVATELŮ PŘIPOJENÍ K INTERNETU
DESIGN OF INTERNET SERVICE PROVIDER DATA NETWORKS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **Stanislav Vodehnal**
AUTHOR

VEDOUCÍ PRÁCE — **doc. Ing. Vít Novotný, Ph.D.**
SUPERVISOR

BRNO 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní obor **Teleinformatika**
Ústav telekomunikací

Student: Stanislav Vodehnal ***ID:*** 158266
Ročník: 3 ***Akademický rok:*** 2015/16

NÁZEV TÉMATU:

Návrh datových sítí poskytovatelů připojení k Internetu

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se problematikou architektur sítí poskytovatelů připojení do Internetu různé úrovně a prostudujte v čem se sítě jednotlivých úrovní liší. Zaměřte se na poskytovatele využívající ve svých sítích přepojování na L2 s využitím standardů IEEE 802.1ah a 802.1aq. Porovnejte tyto technologie s původní technologií využívající VLAN a MSTP, a také mezi sebou z hlediska efektivity, spolehlivosti a nákladů na zavedení a na provoz. Na základě nabytých znalostí a možností Ústavu telekomunikací navrhněte laboratorní úlohu a vypracujte k ní návod.

DOPORUČENÁ LITERATURA:

[1] ALLAN, D., BRAGG,N. 802.1aq Shortest Path Bridging Design and Evolution: The Architect's Perspective. John Wiley & Sons, ISBN: 978-1-118-14866-2, USA, 2012

[2] HECKMANN,O. M. The Competitive Internet Service Provider: Network Architecture, Interconnection, Traffic Engineering and Network Design. ISBN: 978-0-470-03004-2, 2007

Termín zadání: 1.2.2016 ***Termín odevzdání:*** 1.6.2016

Vedoucí práce: doc. Ing. Vít Novotný, Ph.D.
Konzultant bakalářské práce:

doc. Ing. Jiří Mišurec, CSc., *předseda oborové rady*

UPOZORNĚNÍ:
Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRANKT

Tato bakalářská práce se zabývá technikami pro návrh datových sítí. Jsou zde popsány skupiny protokolů STP a virtuálních sítí VLAN, které jsou v dnešní době stále hodně používané. Dále jsou zde popsány moderní techniky pro návrh datových sítí, jako jsou protokoly PB, PBB a SPB. Závěrem práce je laboratorní úloha zabývající se sítěmi VLAN a QinQ.

KLÍČOVÁ SLOVA

LAN, VLAN, STP, MSTP, RSTP, PB, PBB, SPB IEEE, 802.1, L2 Vrstva, QinQ

ABSTRACT

This bachelor thesis deals with techniques for designing data networks. They described a group of protocols STP and virtual networks VLAN that are still getting a lot of use. They are further described advanced techniques for the design of data networking protocols such as PB, PBB and SPB. Conclusion of work is a laboratory exercise dealing with the VLAN and QinQ.

KEYWORDS

LAN, VLAN, STP, MSTP, RSTP, PB, PBB, SPB IEEE, 802.1, L2 Layer, QinQ

VODEHNAL, S. *Návrh datových sítí poskytovatelů připojení k Internetu.* Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2016. 56 stran, 3 přílohy. Vedoucí bakalářské práce doc. Ing. Vít Novotný, Ph.D..

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Návrh datových sítí poskytovatelů připojení k internetu“ jsem vypracoval(-a) samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil(-a) autorská práva třetích osob, zejména jsem nezasáhl(-a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(-a) následku porušení ustanovení § 11 a následujících autorského zákona c. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonu (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne ....

podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu práce panu doc. Ing. Vítu Novotnému, Ph.D. za odborné vedení, trpělivost a podnětné návrhy k této bakalářské práci.

V Brně dne ....

podpis autora

SIX research centre
sensor, information and communication systems

Faculty of Electrical Engineering
and Communication

Brno University of Technology
Technicka 12, CZ-61600 Brno, Czechia

http://www.six.feec.vutbr.cz

Výzkum popsaný v této bakalářské práci byl realizovaný v laboratořích podpořených projektem Centrum senzorických, informačních a komunikačních systémů (SIX); registrační číslo CZ.1.05/2.1.00/03.0072, operačního programu Výzkum a vývoj pro inovace.

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI

2007–13
OP Výzkum a vývoj pro inovace

Obsah

Seznam obrázků

Seznam tabulek

Úvod

V současnosti lze síť navrhnout několika způsoby, v této bakalářské práci se budu soustředit na návrh sítě především na spojové vrstvě. Pro návrh takovéto sítě je zapotřebí širší znalost toho, v jakém prostředí bude návrh sítě probíhat. Samozřejmě dnes, kdy jsou sítě už opravdu velmi rozlehlé, není dobré, aby takováto síť fungovala pouze s přepínači bez jakéhokoliv rozdělení komunikace, jako to je v sítí LAN. V dnešní době bývá nutností, aby byly sítě rozděleny ať už ze strany managementu, jednodušší správy a menšího zatížení jednotlivých zařízení v síti (v tomto případě přepínačů) ale i pro rozdělení komunikace.

V první kapitole se budu zabývat s již zavedenými technikami, které se pro návrh sítě dnes běžně používají, především u menších poskytovatelů internetového připojení. Technologie sítí VLAN je jedna z nich, která se dnes používá a troufám si tvrdit, že se jen tak používat nepřestane. S použitím těchto sítí VLAN se naskytuje obrovská možnost škálovatelnosti a dělení sítí na menší logicky oddělené sítě. Takovéto sítě umožňují především úsporu s ohledem na menší počet zařízení. Další technikou, která se dnes stále hodně používá je Spanning Tree Protocol.

V druhé kapitole se zaměřím na moderní techniky pro návrh sítí. Je nutné, aby se při návrhu sítě dbalo také na redundanci důležitých linek, nejen pro spokojenost zákazníků, ale také pro správce takové to sítě, kterého v případě výpadku nečeká spousta hovorů od zákazníků, ale v takovém případě získá cenný čas na opravu problému v síti. Síť poběží dál, v nejhorším případě zde bude minimální výpadek, který ale zákazník nezaznamená. Pro takovéto zapojení je však nutné, aby daný správce měl znalost ohledně protokolu Spanning Tree Protocol, který zabraňuje zhroucení sítě právě v případě, kdy jepoužit redundantní spoj na spojové vrstvě. Budu se zabývat novou technikou Shortest Path Briging jako náhradu za Spanning Tree Protocol. Dále proberu nové techniky, jako jsou Provider Bridging pro menší poskytovatele a Provider Backbone Bridging pro mnohem větší poskytovatele.

V třetí části bakalářské práce je zmíněn návrh laboratorní úlohy, který je uveden v příloze. Úloha se zaměřuje na funkčnost sítí VLAN a QinQ. Úloha se také zabývá protokoly Telnet a SSH, které slouží pro vzdálené přihlášení a správu. V druhé příloze jsou pak výsledky měření jednotlivých úkolů v laboratorní úloze.

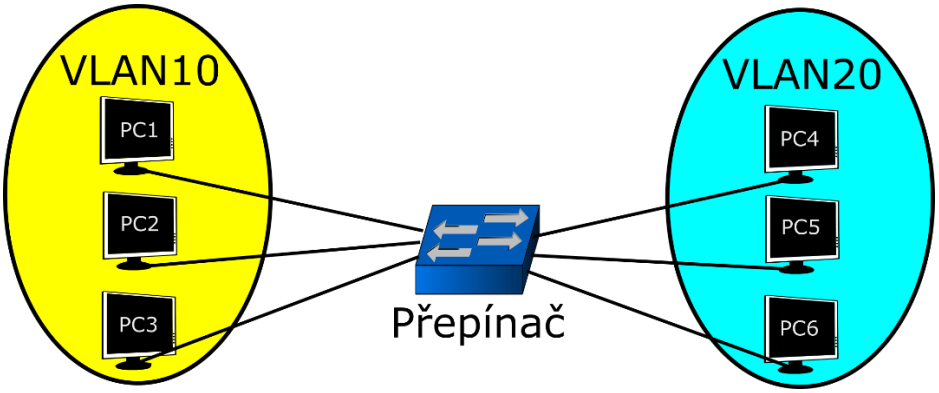
1 Základní techniky pro návrh sítí

Lokální sítě LAN (Local Area Network) mohou být různě velké, firmy nebo větší společnosti mají většinou mnoho takovýchto sítí LAN ve své infrastruktuře. V sítích tohoto typu je fyzicky propojen každý k jedné fyzické síti a tím se naskytuje možnost nejen sdílení souborů ale i zařízení v síti jako jsou tiskárny, faxy… Avšak u takovýchto sítí bývá omezení takové, že dojdou např. volné porty na přepínači. Dalším problémem je, že všechna sdílená data v síti mohou být zobrazena všemi uživateli této sítě a tím je snížena bezpečnost. Příklad takového propojení můžou být ve firmě různá oddělení jako je výroba, kanceláře vývoje, kanceláře účetní… Takovéto typy sítí však v dnešní době najdeme opravdu málo a jsou součástí jedné malé sítě například v domácnostech nebo v menších kancelářích.[2,3]

Technologie VLAN (Virtual Local Area Network) přišla o rok později (roku 1995) po uvedení přepínačů na trh. Samozřejmě si ihned získaly pozornost provozovatelů sítí ne jenom z toho důvodu, že jim byla poskytnuta lepší možnost managementu, ale i z důvodu úspor. VLAN je technologie, která umožňuje oddělení fyzického spojení od logického (nezávisle na fyzickém uspořádání). Uživatelé v této síti jsou stále fyzicky propojeni, jako to je to u sítí LAN s tím rozdílem, že je nutné, aby mezi nimi byl směrovač pro komunikaci mezi sebou. Pomocí Virtuálních LAN sítí dosáhneme takového efektu, jakoby jsme měli propojenou skupinu zařízení pomocí jednou skupinou přepínačů a druhou skupinu zařízení jinými přepínači. V praxi je samozřejmě nutná také komunikace mezi těmito sítěmi VLAN, která bude vysvětlena později.[1,2,5]

1.1 Komunikace VLAN sítí

Pro komunikaci v sítích VLAN je nutné mít přepínač, který umí pracovat se sítěmi VLAN (tzv. VLAN-aware zařízení).



Obrázek 1.1: Sítě VLAN na jednom přepínači

Na obrázku 1.1 je zobrazena komunikace pomocí jednotlivých zařízení v síti pomocí VLAN. Máme zde dvě sítě VLAN, jedna je VLAN 10 a druhá je VLAN 20.

V tomto případě nemůžou komunikovat zařízení mezi jednotlivými sítěmi VLAN a dosáhli jsme tak logického oddělení jednotlivých sítí.[1,2,4]

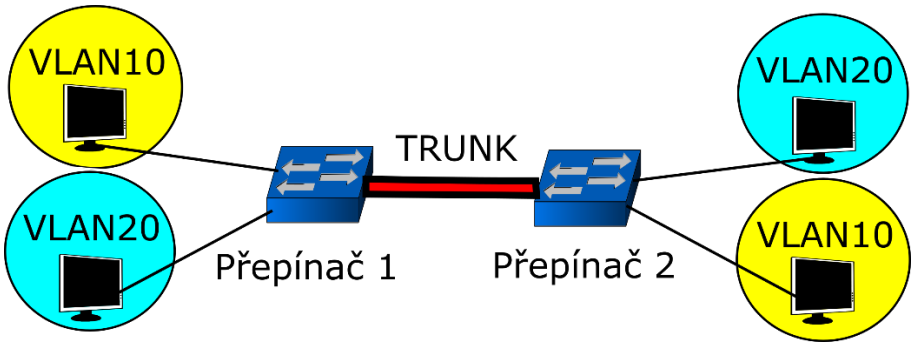
Přepínač v tomto případě funguje tak, jakoby zde byli dvě rozdílné sítě, každá fyzicky oddělená a každá na jiném přepínači. Ač jsou tyto sítě VLAN na jenom fyzickém přepínači, komunikace mezi jednotlivými sítěmi VLAN je možná pouze v případě, že by zde byl směrovač. V tomto případě tedy mezi sebou můžou komunikovat zařízení PC1, PC2 a PC3, která jsou ve VLAN 10. Dále může probíhat komunikace mezi zařízení PC4, PC5 a PC6, která jsou ve VLAN 20.[1]

Komunikace tedy probíhá tím způsobem, že pokud chce komunikovat PC0 s PC1 tak se na vstupu přepínače příchozí rámec označí VLAN 10 a může vstoupit pouze naporty kde je nastavena VLAN 10. Lze v tomto případě použít i trunk (bude vysvětleno později), ale to lze pouze v případě, že síťová karta umí přijímat a odesílat tagované rámce.[1]

1.1.1 Komunikace mezi jednotlivými sítěmi VLAN

Pokud tedy chceme, aby bylo možné komunikovat mezi jednotlivými sítěmi VLAN je nutné použít směrovač, jak bylo zmíněno výše. Tento způsob komunikace je možné provést několika způsoby.

- První je ten, že propojíme všechny porty přepínače, které chceme, aby mezi sebou komunikovaly do portu na směrovač. Pokud bude směrovač správně nakonfigurovaný, budou mezi sebou jednotlivé sítě VLAN komunikovat, aniž by věděly, že jsou připojeny do jednoho fyzického přepínače. Toto je ale složitější a zabírá zbytečně mnoho portů pro komunikaci. Z hlediska úspory nikterak efektivní.
- Další způsob je při použití trunku v kombinaci s L3 přepínačem nebo směrovačem, kde funkčnost trunku bude vysvětlena později. Pomocí trunku je možné přenést několik VLAN jediným portem, jak je zobrazeno na obrázku 1.2. To s sebou nese mnoho výhod. Jedna z výhod je ta, že je možné tyto sítě VLAN protáhnou téměř všude kam je potřeba. Další výhoda je, že se šetří rozhraním na směrovači. Nevýhoda bývá taková, že veškerý provoz je omezen na přenosovou kapacitu daného portu (to však v dnešní době nebývá problém). Při tomto zapojení je nutné, aby porty do kterých je připojen trunk, byly nastaveny jako trunk a k nim přiřazeny všechny sítě VLAN, které chceme přenášet. Na obrázku 1.2 je zobrazeno použití trunku s L3 přepínači.



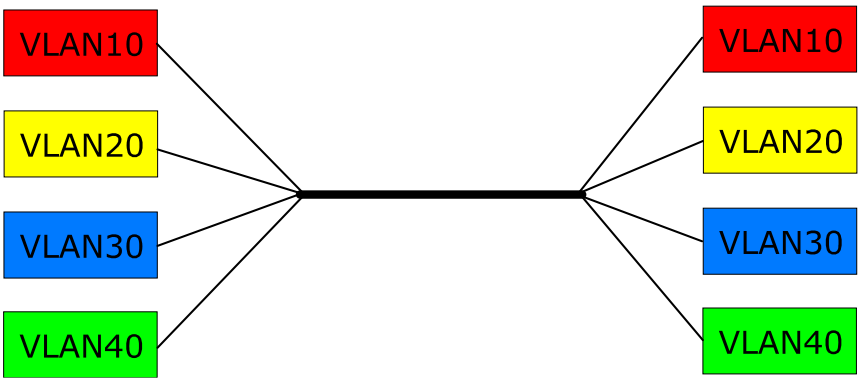
Obrázek 1.2: Sítě VLAN s použitím trunku

1.1.1.1 Možnosti uživatelské mobility

Jak již bylo řečeno, velkou výhodou použití sítí VLAN je takové, že je možné je dostat téměř všude uvnitř jedné sítě (především je to výhoda u velké sítě, například síť poskytovatele připojení k internetu). Mobilita spočívá v tom, že není omezena na jedno jediné místo, ale uživatel, který využívá právě tuto síť VLAN, se může pohybovat všude, kde je daná VLAN přivedená ať už pomocí jiného portu na přepínači, nebo pomocí trunku. Omezení tedy není tak velké jak je to u sítí LAN. Uživatel se tedy může pohybovat ne jen v rámci pater v dané budově nebo budov v areálu, ale může se pohybovat i v rámci měst nebo států. V případě, že se uživatel pohybuje v rámci měst nebo států, je zde možnost využití připojení pomocí VPN (Virtual Private Network) a díky správné konfiguraci dále využívat dané sítě VLAN. Samozřejmě při komunikaci s použitím VPN nebývá připojení již tak rychlé a výrazně záleží na tom, jakou rychlost mají obě sítě pro přístup do internetu pomocí WAN (Wide Area Network). Při tomto připojení se tedy musí počítat s vyšší latencí ale i nižší přenosovou rychlostí.[1,2,3]

1.1.2 Komunikace s použitím trunku

Ve složitých nebo velkých sítích může být správa sítí VLAN časově náročná a lze v ní snadno chybovat. VLAN Trunking protocol je prostředkem s jehož pomocí lze na centrálních zařízeních (často to bývá L3 přepínač, nebo směrovač) nastavit názvy a čísla sítí VLAN, přičemž výslednou konfiguraci lze distribuovat na ostatní zařízení. Jako trunk se označuje rozhraní, které je schopno přenést několik sítí VLAN současně. Pro přenos pomocí trunku potřebujeme tedy minimálně 2 přepínače. Trunking se obecně týká přepínačů, avšak k trunku lze připojit také směrovač nebo L3 přepínač, jak bylo zmíněno v kapitole 1.1.1. Obrázek 1.3 znázorňuje schématické znázornění trunku. Sítě VLAN 10, 20, 30, 40 existují na obou stranách trunku. To znamená, že veškerý provoz ze sítě VLAN 10 na Přepínači 1, který je určen pro VLAN 10 na Přepínači 2, musí projít trunkem. Aby přepínač věděl, na jaký port má předat rámec, musí rámec obsahovat odkaz na síť VLAN, do které je určen. Toto neplatí pro pakety, které o sítích VLAN nemají ponětí, sítě VLAN jsou vždy jen na spojové vrstvě.[1,2,5]



Obrázek 1.3: Schématické znázornění trunku

1.1.3 Značení rámců

Pro značení (nebo také taggování) rámců vyvinula např. firma Cisco protokol ISL (Internet-Switch Link), který však funguje pouze na zařízeních Cisco. Aby bylo docíleno komunikace sítí VLAN mezi různými přepínači různých firem, byl vyvinut protokol 802.1Q který podporuje až 4096 sítí VLAN a je standardem organizace IEEE. Protokol 802.1Q vkládá data do existující hlavičky ethernetového rámce mezi pole Zdrojová adresa a Typ/Délka další 4B pole značky jak je zobrazeno na obrázku 1.2. Níže je vedeno, co všechno ethernetový rámec obsahuje.[1,4,5]

6 bajtů	6 bajtů	2 bajty	2 bajty	2 bajty	46-1500 bajtů	4 bajty
Cílová adresa	Zdrojová adresa	VLAN Protokol ID	Tag control info	Lenght/ Type	DATA	FCS

Obrázek 1.4: Ethernetový rámec se značkováním sítí VLAN

- VLAN Protokol ID (EtherType) – jedná se identifikátor typu rámce, který říká, že se jedná od protokol 802.1q a obsahuje hodnotu 0x8100. Pro zařízení, která pracují se sítěmi VLAN to znamená, že další 2 bajty ponesou informace o síti VLAN.
- Tag control info obsahuje
  - PCP (Priority code point) o 3 bitech. Obsahuje uživatelskou prioritu rámce.
  - CFI (Canonical Format Indicator) – určuje v jakém pořadí je přenášen rámec.
  - VID (VLAN Indentifier) – zbylých 12 bitů (což je právě 4096 možných ID VLAN) identifikuje číslo sítě VLAN.

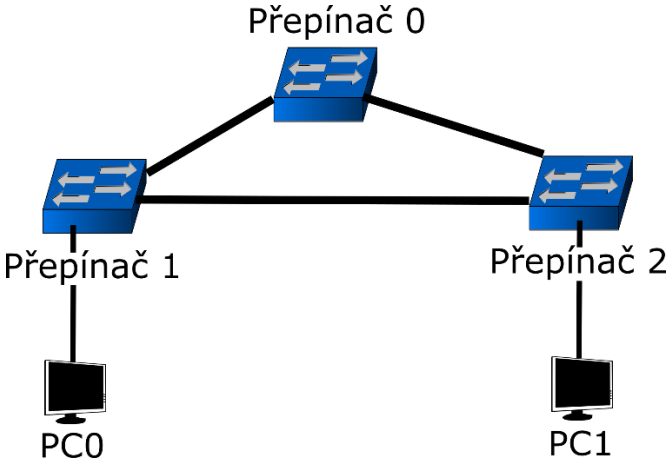
1.1.4 Metoda pro zařazení do sítí VLAN

Značkování rámců již bylo vysvětleno. Abychom byli schopni takový rámec označkovat, je potřeba vědět, jakým způsobem takovýto rámec chceme označkovat. Existuje několik způsobů mapování, podle kterého se dané rámce značkují a jsou uvedeny níže, avšak nejpoužívanější mapování je podle portu:

- **Podle portu** – při tomto nastavení, je port přepínače zařazen do jedné sítě VLAN (např. VLAN 10). Veškerá komunikace, která na tento port přijde, musí odpovídat dané síti VLAN (v našem případě musí odpovídat VLAN 10) jinak neprojde dál přes tento port. V opačném případě, kdy komunikace ochází z tohoto portu ven do sítě, označí se na tomto portu (v našem případě se označí jako VLAN 10). Tento způsob je nejjednodušší a nejvíce používaný.
- **Podle MAC adresy** – tento způsob značení je poněkud složitější než první způsob. Rámec se v tomto případě značí podle své MAC adresy. Výhoda spočívá v tom, že pokud dané zařízení přeneseme (přepojíme) k jinému portu na přepínači, bude stále přiřazeno do stejné sítě VLAN a na portech není třeba nic přenastavovat.
- **Podle protokolu** – tento způsob pracuje na 3. vrstvě TCP/IP protokolu, v tomto případě přepínač přidá rámec do dané sítě VLAN podle protokolu přenášeného paketu.
- **Podle aplikace** – způsob založený na protokolech vyšších aplikačních vrstev.[2,5,6]

1.2 Smyčky v sítích

V dnešní době je zvykem mít spoje, které jsou například páteřní (nebo z jiného důvodu více důležité) redundantní (záložní). To se dělá takovým způsobem, že např. přepínače (mosty) zapojíme do smyčky, jak je zobrazeno na obrázku 1.5 a to s sebou nese právě tu výhodu záložního spoje při výpadku hlavního spoje (obě spojení mohou mít stejné vlastnosti, stejnou propustnost…) nebo také při výměně jednoho z těchto přepínačů. Hlavním nedostatkem takovéhoto zapojení je, že v síti začnou nastávat smyčky. Smyčky vznikají například také neodbornou manipulací s přepínači jako je propojení dvou přepínačů dohromady, což je vlastně způsob popsaný výše s tím rozdílem, že o tom uživatel provede nechtěnou manipulací. Smyčky v sítí mají za následek zahlcování sítě a téměř okamžité zhroucení sítě. [2,6]



Obrázek 1.5: Názorná ukázka smyčky v síti

1.2.1 Jak vznikají smyčky

Podle obrázku 1.5 vznikne smyčka a ta způsobí to, že mezi PC0 a PC1 existuje více než jedna cesta. Pokud Přepínač 1 příjme unicastový rámec od PC0, který je určen pro PC1 pošle Přepínač 1 rámec na Přepínač 2 ale i na Přepínač 0. PC0 obdrží v tomto případě dvě kopie unicastového rámce.

Dalším, mnohem horším případem je vysílání všesměrové zprávy (broadcastu) v takovéto sítí podle obrázku 1.5. Pokud by PC0 začal vysílat broadcast, Přepínač 1 by tento rámec obdržel a odeslal ho na všechny odchozí porty kromě příchozího (tady na Přepínač 2 a Přepínač 0). Pokud by Přepínač 0 obdržel broadcastový rámec od Přepínače 1, poslal by rámec na všechny porty kromě příchozího (na Přepínač 2). Obdobně v prostředí se smyčkami by takováto komunikace pokračovala do té doby, dokud by nedošlo k úplnému zahlcení sítě. Tomuto jevu se říká všesměrová bouře (Broadcast storm). [2,6]

Při smyčkách tedy mohou nastat problémy:

- Broadcastová bouře
- Několikanásobné doručení rámců
- Problémy s připojením nebo nestabilita MAC přepojovací tabulky (CAM tabulky)

1.2.2 Spanning Tree Protocol

Aby se zamezilo smyčkám na spojové vrstvě, byl vyvinut protokol Spanning Tree Protocol (STP[1]). Defaultně bývá hlavně na Cisco přepínačích protokol STP zapnut. Avšak ne všechny přepínače (mosty[2]) protokol STP umí. Osobně se mi povedlo zablokovat lokální síť tím, že jsem jeden přepínač propojil jedním kabelem a to jen

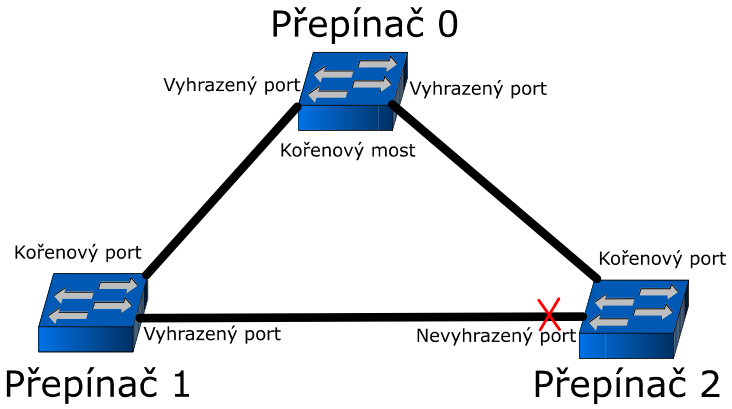
[1] Protokol STP je navržen tak, aby zabraňoval smyčkám mezi mosty. Jako mosty jsou označovány zařízení, která spojují více segmentů v rámci jedné kolizní domény (spojuje dvě části sítě na linkové vrstvě).
[2] Přepínač byl dříve nazývat víceportový most (později přejmenován na přepínač). Jako most lze považovat propojení mezi přepínači, proto je potřeba od sebe odlišit most a přepínač.

chybou toho, že jsem si spletl kabely. V jednom okamžiku je nutné, aby mezi mosty bylo pouze jedno spojení i v tom případě, pokud potřebujeme redundanci spojení. Lze to vyřešit i tím, že daný redundantní spoj zakážeme, ale v případě výpadku hlavního spoje je nutné ručně zapnou záložní spojení, což není efektivní. Protokol STP je tedy schopný toho, že zakáže port, díky kterému by vznikla smyčka (nebo redundantní spoj) a právě tím zamezí vzniknutím smyčkám v síti.[1]

Protokol STP si vybere podle svého algoritmu kořenový přepínač (most) v síti. Tento kořenový most je přepínač, který musí všechny přepínače v síti dosáhnout pomocí nejkratší možné cesty. Protokol STP vypočítá cenu každé cesty k tomuto kořenovému mostu, kde cesta s nejnižší cenou zůstane nedotčena, zatímco ostatní cesty přeruší. Každý přepínač v sítí, který podporuje protokol STP, odesílá rámce BPDU (Bridge-protocol data units) defaultně každé dvě sekundy.[1]

1.2.2.1 Role portů při použití protokolu STP

- **Určení kořenového mostu** – Při spuštění přepínače, si každý myslí, že je kořenovým mostem sítě. Nastaví tedy ID kořene na ID místního mostu (ID mostu je kombinace priority mostu a MAC adresy mostu) ve všech odchozích rámcích BPDU. Pokud přepínač obdrží ID rámce BPDU který obsahuje nižší ID kořene, považuje tento přepínač za kořenový. Místní přepínač pak používá toto ID kořene v jím odesílaných rámcích BPDU.
- **Určení nejlepší cesty ke kořenovému mostu** – Pokud přepínač příjme rámce BPDU na více než jednom portu, existuje tím pádem více cest ke kořenovému mostu. Jako nejlepší cesta se určí ta, která má nejnižší cenu cesty ke kořenovému mostu.
- **Určení kořenového portu** – kořenový port je takový port, který má ke kořenovému mostu nejkratší cestu nebo je přímo připojen ke kořenovému mostu.
- **Učení vyhrazeného (designated) portu** – kořenový most nemá kořenové porty ale vyhrazené porty. V segmentu je vyhrazený port takový port, který má ke kořenovému portu nejkratší cestu.
- **Blokování nepředávajících portů** – porty které obdrží rámec BPDU a nejsou vyhrazeným ani kořenovým portem jsou přepnuty do blokujícího stavu. Porty jsou stále zapnuty s tím rozdílem, že jim není umožněno předávání provozu.[1]

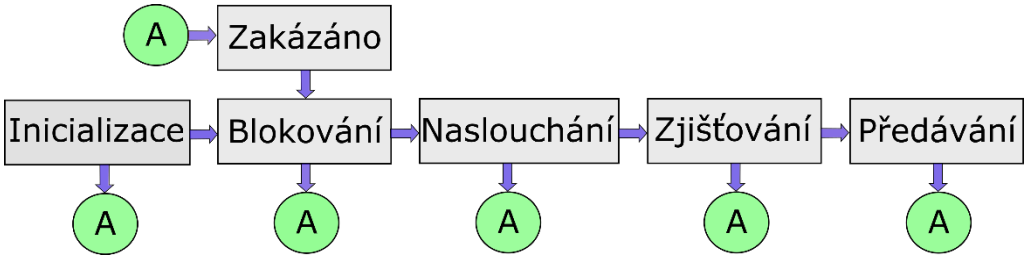


Obrázek 1.6: Zapojení s použitím protokolu STP

Na obrázku 1.6 je zobrazeno zapojení s použitím protokolu STP a jsou zde zobrazeny jednotlivé role portů, popsané v kapitole 1.2.2.1.

1.2.2.2 Stav portů protokolu STP

Před tím, než port protokolu STP přejde do režimu, kdy může přenášet data, prochází řadou stavů protokolu STP, jak je zobrazeno na obrázku 1.7.



Obrázek 1.7: Stavy portů protokolu STP

- **Inicializace** – port je v tomto stavu, pokud se například přepínač právě zapne nebo se daný port nacházel předchozím stavu jako vypnutý.
- **Zakázáno** – v tomto režimu port nepřijímá ani neodesílá žádná data včetně rámců BPDU. Port se typicky nachází v tomto režimu, pokud je nějakým způsobem poškozen, nebo byl jednoduše správcem sítě vypnut.
- **Blokování** – port je v tomto režimu zapnut ale veškerá komunikace, která by přes něj mohla eventuálně projít je zakázána. Přijímá pouze rámce BPDU (ale tyto rámce už dále nevysílá), přijímá a odpovídá na zprávy týkající se správy sítě.
- **Naslouchání** – v tomto režimu není stále povoleno vysílání rámců s tím rozdílem oproti Blokování, že port odesílá i přijímá rámce BPDU.
- **Zjišťování** – v tomto režimu se port připravuje na odesílání rámců. Zachycuje rámce, které přijdou na jeho port, a získává z nich MAC adresy,

které jsou následně uloženy do tabulky CAM. Tento čas se nazývá jako směrovací zpoždění (forwarding delay).

- **Předávání –** jakmile port stráví nějaký čas učením MAC adres příchozích rámců, je mu dovoleno vysílat datové rámce.[1]

1.2.3 Rapid Spanning Tree Protocol

Rapid Spanning Tree Protocol (RSTP) je vylepšený Spanning Tree Protocol a v dnešní době používaný v mnoha sítích. Oba protokoly (jak RSTP a STP) jsou navzájem kompatibilní. Protokol RSTP není jen zpětně kompatibilní, ale usnadňuje i konfiguraci oproti STP. Hlavní výhoda však spočívá v rychlejší konvergenci při změně topologie. Maximální čas konvergence u STP je 50s (v praxi kolem 30s) ale u RSTP se čas konvergence pohybuje kolem 1 až 2s.[2,8]

1.2.3.1 Stav portů protokolu RSTP

Oproti protokolu STP je u protokolu RSTP změna v tom, že se neprohází 5 stavů, které se kontrolují na portech, ale stavy Zakázáno, Blokování a Naslouchání jsou zde implementovány do jednoho procesu a tím je stav Vyřazeno (Discard).

- **Vyřazeno –** jak bylo zmíněno výše, je tento stav kombinací tří stavu z protokolu STP. Nejsou zde žádné rozdíly, které jsou ve stavu Naslouchání a Blokování.
- **Zjišťování –** V tomto stavu je port připraven předávat data. Funkce je prakticky stejné jako u protokolu STP.
- **Předávání –** jakmile port stráví nějaký čas zjišťováním MAC adres je připraven vysílat data. Tato funkce je také stejná jako u protokolu STP[2,8]

1.2.3.2 Role portů při použití protokolu RSTP

Spaning Tree Algoritmus je založen na zasílání rámců BPDU. Každý rámec BPDU obsahuje metodu, která je použita k porovnání ostatních rámců BPDU a poté dělá rozhodnutí na základě toho, která z těchto jednotek obsahuje užitečnější informace. Role portů u protokolu RSTP jsou Kořenový port, Vyhrazený port, Alternativní port a Záložní port. Alternativní a Záložní port mají podobnou funkci. Oba jsou blokovány a jediný rozdíl je v tom, že Alternativní port slouží jako alternativní cesta ke kořenovému. Naopak Záložní port slouží jako redundantní cesta k segmentu. [2][8]

1.2.4 Multiple Spanning Tree Protocol

Multiple Spanning Tree Protocol (MSTP) je rozšířením protokolu RSTP. Umožňuje nezávislou konfiguraci protokolu pro jednotlivé sítě VLAN. Jako základ používá RSTP, ale navíc používá sítě VLAN, které mohou být s tímto protokolem seskupeny do jednotlivých úrovní Multiple Spanning Tree Instance (MSTI). MSTP

uchovává všechny informace v jednom jediném rámci BPDU. To samozřejmě snižuje počet odesílaných rámců BPDU.[2,8]

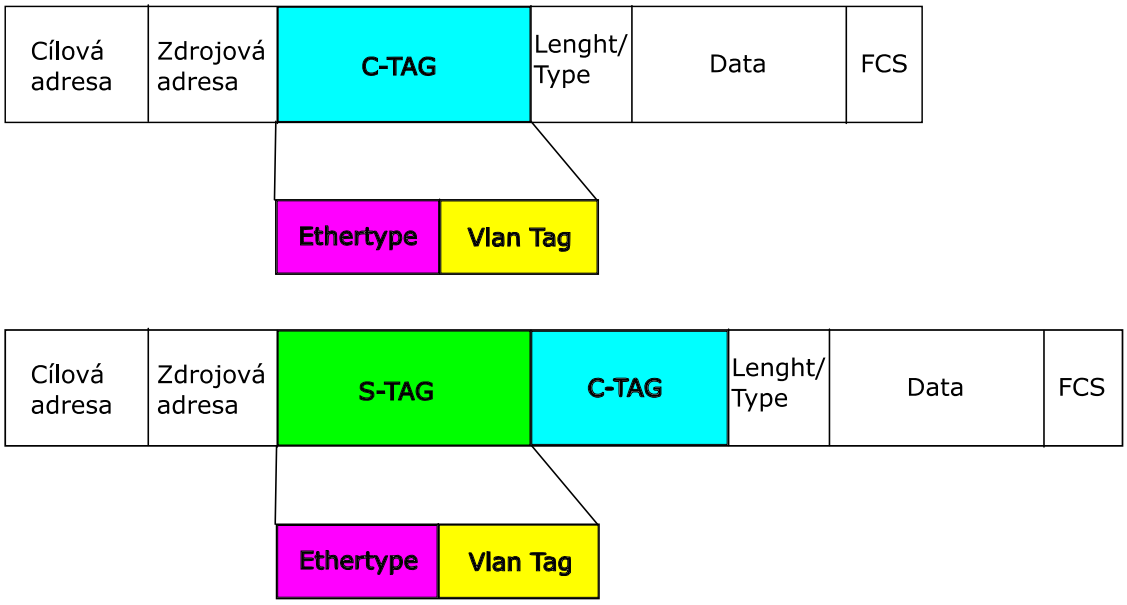
2 Moderní techniky pro návrh datové sítě

2.1 Provider Bridge

Původní standard (který se ale dnes stále hodně používá) 802.1Q, používaný pro tagování jednotlivých sítích VLAN, umožňuje vložení pouze jednoho identifikátoru VLAN do hlavičky rámce. Standart 802.1ad umožňuje zapouzdřit několik sítí VLAN (indentifikátorů sítí VLAN) do jedné jediné sítě VLAN a bývá označovaný jako „Q-in-Q“. Označení je odvozené od 802.1Q kde právě při Q-in-Q se myslí to, že VLAN ID je označkována další VLAN ID. Tímto způsobem je umožněno většímu poskytovateli používat své vlastní sítě VLAN a zákazník (např. firma, menší poskytovatel internetového připojení) s tím může nakládat tak, jak by v tomto případě používal trunk. Řešení spočívá v tom, že se do hlavičky rámce zavede další ID sítě VLAN. Původní limit počtu sítí VLAN byl u 802.1Q na 4096 možných sítích VLAN. Tímto způsobem se limit sítí VLAN zvedl na přibližně 16,7 (4096×4096) milionu. Stejně jako u sítích VLAN se pro rozlišení zákazníků (service provider) používá značkování (tagging). Nevýhodou PB je to, že přepínače musí být schopny pojmout všechny adresy MAC do svých CAM tabulek, což v dnešní době není možné vzhledem k obrovskému množství zařízení v síti. [9,10]

2.1.1 Formát rámce

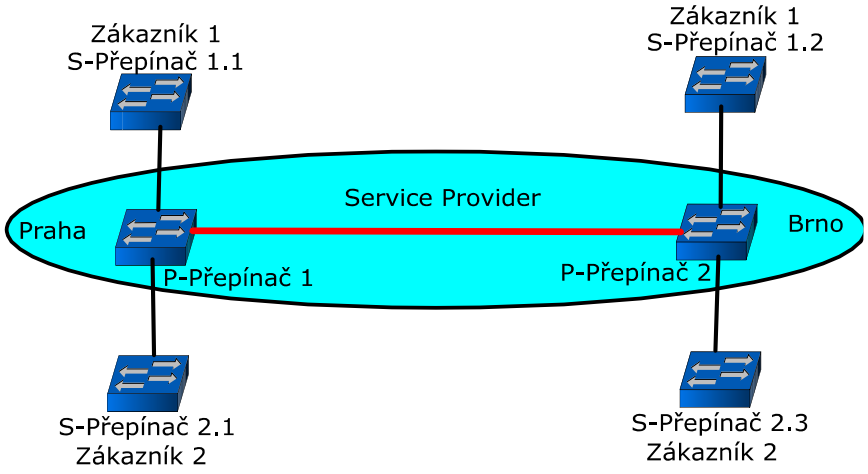
Přes jednoho velkého providera komunikuje spousta menších zákazníků a veškerá komunikace každého takového zákazníka musí být (nebo by měla být) izolována od komunikace jiných zákazníků. Samozřejmě na spojové vrstvě je řešení takové, že se jednoduše zavedou sítě VLAN a problém je vyřešen. Avšak každý takovýto zákazník, může používat několik stejných sítí VLAN jako ten druhý. Pro tento účel tu tedy máme Provider Bridging (PB) kde, přestože zákazníci užívají stejné sítě VLAN uvnitř jejich vlastních sítích, nijak se jich to po přístupu do internetu nedotkne, díky právě PB. Pro zákazníka se používá značka zvaná C-TAG (customer tag nebo také značka zákazníka) označovaný jako vnitřní (inner) VLAN a pro poskytovatele se používá S-TAG (service provider tag nebo také značka poskytovatele) označovaný jako vnější (outer) VLAN.



Obrázek 2.1: Formát rámce při použití Provider Bridging

Na obrázku 2.1 je znázorněno, jak vypadá otagovaný rámec zákazníka, který používá sítě VLAN a rámec poskytovatele, který používá Provider Bridging. Pole C-TAG je dlouhé 4 bajty, stejně jako u 802.1q. První 2 bajty jsou pro EtherType (TPID – Tag Protocol Identifier) a obsahuje hodnotu 0x8100. Druhé 2 bajty obsahují PCP, CFI a VID stejně jako u standardu IEEE 802.1q. Rámec v síti poskytovatele je také zobrazen na obrázku 2.1. Obsahuje jak C-TAG což je značka rámce zákazníka, tak i značku S-TAG. Pole S-TAG obsahuje stejné informace jako pole C-TAG s tím rozdílem, že EtherType má hodnotu 0x88a8 a tato hodnota je standardem. Aby to nebylo tak jednoduché, firma Cisco používá ještě další formáty značení QinQ EtherType a to 0x8100, 0x9100 a 0x9200. [9,10]

2.1.2 Komunikace pomocí Provider Bridging



Obrázek 2.2: Způsob komunikace pomocí PB

Postupně si popíšeme, jak probíhá komunikace podle obrázku 2.2. Podle obrázku 2.2 je zde jeden poskytovatel (service provider) a dva zákazníci. Poskytovatel poskytuje konektivitu na úrovni 2. vrstvy zákazníkům. Oba zákazníci mají pobočku jak v Praze, tak v Brně. Oba zákazníci jsou propojeni pomocí spojení od jednoho poskytovatele, tudíž jsou v Praze i v Brně na stejné síti LAN. Poskytovatel má dva přepínače. Jeden je v Praze (označený jako P-Přepínač 1) a druhý v Brně (označený jako P-Přepínač 2). Zákazníci mají vlastní přepínače (označené jako S-Přepínač + číslo, zobrazené na obrázku 2.2).

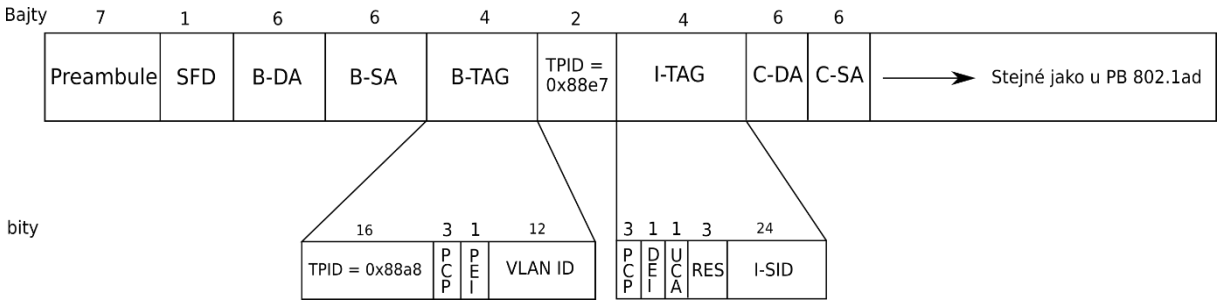
Zákazník 1 používá uvnitř své sítě VLANy 10, 20, 30 (tyto vlany jsou v tomto případě vnitřními VLANami, inner VLAN). Spojení mezi přepínači P-Přepínač 1 a S-Přepínač 1.1 je nastaveno jako trunk, obdobně mezi P-Přepínač 2 a S-Přepínač 1.2 je nastaveno jako trunk. Podobné zapojení je i u Zákazníka 2 s tím, že také používá sítě VLAN 10, 20, 30. Vzhledem k tomu, že oba mají jednoho společného poskytovatele, je nutné, aby tento poskytovatel oddělil komunikaci těchto 2 zákazníků, vhledem k tomu, že oba zákazníci používají stejné sítě VLAN, což by bez použití 802.1ad nebylo možné.

Řešením tedy v tomto případě je použití Provider Bridging (802.1ad). Poskytovatel jednoduše přidělí každému zákazníkovy jinou vnější síť VLAN (outer VLAN). Pro Zákazníka 1 se rozhodne použit VLAN ID 100 a pro Zákazníka 2 použije VLAN ID 150. Veškerá komunikace mezi Prahou a Brnem určená pro Zákazníka 1 bude označena v hlavičce rámce S-TAG jako VLAN ID 100 a veškerá komunikace mezi Prahou a Brnem určena pro Zákazníka 2 bude mít S-TAG označení VLAN ID 150. Tímto způsobem se oddělí komunikace obou zákazníku, přestože oba používají stejné sítě VLAN.[9,10]

2.2 Provider Backbone Bridging

Obdobně jako je Provider Bridging označováno jako „Q-in-Q“ je Provider Backbone Bridging (PBB) označováno jako „MAC-in-MAC“ a je to standard IEEE 802.1ah. Problém u PB je takový, že stále nedovoluje přímé oddělení sítě zákazníka a poskytovatele. Poskytovatel stále potřebuje znát všechny koncové adresy zákazníka. Přidáním PBB hlavičky do rámce zajistí právě to, že je zde komplexní oddělení sítě zákazníka a poskytovatele. To znamená, že přepínače tedy už nemusí znát všechny cílové adresy zákazníka pro přepínání ke koncovým zařízením, ale postačí mu pouze cílová MAC adresa páteřních zařízení v síti, konkrétně to jsou Bakbone Core Bridge a Backbone Edge Bridge. Hlavní výhoda spočívá i v tom, že chyby v síti zákazníka, nijak neovlivní chod sítě. Provider Backbone Bridging také nabízí velkou možnost škálovatelnosti poskytovateli, pro vybudování velké bridgované sítě.[11,12]

2.2.1 Formát Rámce PBB

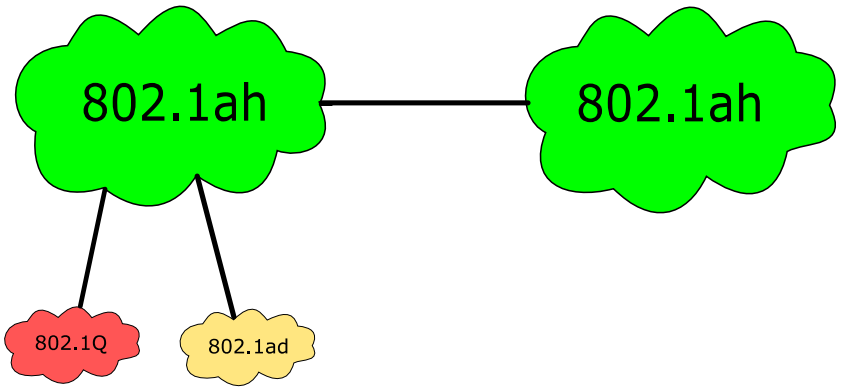


Obrázek 2.3: Formát Rámce PBB

PBB představuje hierarchickou síťovou architekturu spolu s novým formátem rámce, který rozšiřuje možnosti Provider Bridging. Na obrázku 2.3 vidíme formát rámce PBB. Pole B-DA, B-SA, B-TAG a I-TAG jsou pole patřící k PBB podle nichž probíhá komunikace v PBBN (Provider Backbone Bridge Network). Pole označené B-DA je adresa cílové adresy MAC uvnitř PBBN. Obdobně B-SA je zdrojová adresa uvnitř PBBN. Pole označené jako B-TAG je tvořeno prakticky stejným způsobem, jako to bylo u pole S-TAG u 802.1ad s tím rozdílem, že pole B-TAG je vytvořeno pouze pro komunikace mezi BEB (Backbone Edge Bridge) a BCB (Backbone Core Bridge) v síti PBBN jak je zobrazeno na obrázku 2.5. Pole B-TAG také obsahuje B-VLAN (někdy také označovaný VID) pro komunikaci uvnitř PBBN. Pole I-TAG obsahuje 24bitové I-SID, které je právě určeno k identifikaci zákazníka, pro kterého je rámec určen. I-SID se určuje podle poskytovatelské S-TAG, zákaznické C-TAG nebo pomocí obou C-TAG + S-TAG.

2.2.2 Druhy služeb na rozhraní

Máme 3 základní druhy služeb na rozhraních BEB a všechny jsou znázorněny na obrázku 2.4.



Obrázek 2.4: Příklad připojení do sítě používající 802.1ah

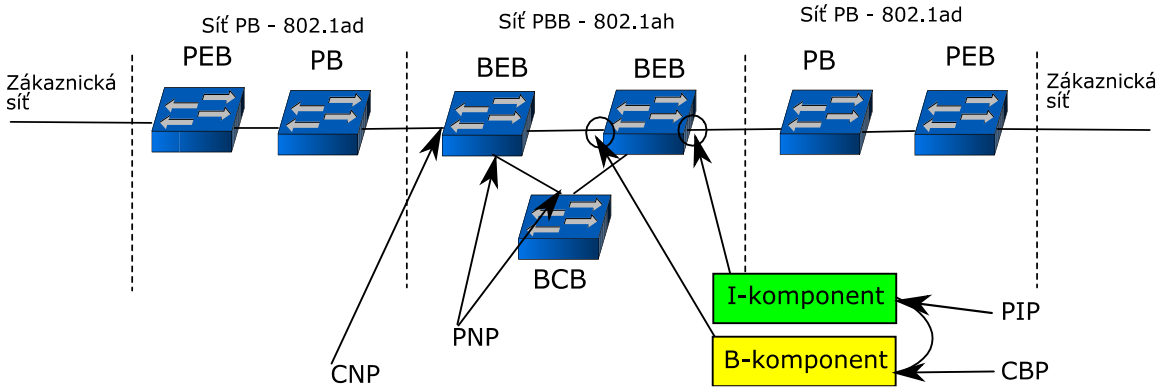
- **Port-based** – tato služba zajišťuje mapování všech neoznačených rámců a C-VLAN označených rámců. Zajištuje právě to, že se do PBBN mohou připojit i zákazníci bez toho aniž by používali nebo komunikovali přes PBN (Provider Bridge Network). Na obrázku 2.4 tomu odpovídá zapojení 802.1Q ↔ 802.1ah.
- **S-Tagged** – tato služba zajišťuje právě komunikaci mezi PBN a PBBN odpovídající zapojení 802.1ad ↔ 802.1ah podle obrázku 2.4. Jeden ze způsobů (označovaný Port Mode) je ten, že všechny rámce označeny S-VLAN z právě jednoho PBN jsou zařazeny do jediné I-SID v PBBN. Dalšími způsoby jsou S-VLAN Mode a S-VLAN Bundle Mode.
- **I-Tagged** – služba pro vnitřní propojení PBBN. Na obrázku 2.4 je znázorněno takovéto propojení 802.1ah ↔ 802.1ah. [12,13,14]

Existuje několik druhů Backbone Edge Bridge. I typ Backbone Edge Bridge (označovaný I-BEB) slouží pouze na propojení mezi 802.1ad nebo 802.1Q sítě do 802.1ah sítě přes port, který se nazývá CNP (Customer Network Port) a zahrnuje pouze I-komponentu. Samotný I-komponent je zodpovědný při obdržení rámce ze sítě zákazníka za jeho zabalení do dané I-SID. Každý z I-komponentu se učí návaznost mezi jednotlivými zákaznickými zdrojovými adresami. Komunikace v samotné síti PBB se uskutečňuje tak, jak to je zobrazené na obrázku 2.5 skrze PIP (Provider Instance Port) a CBP (Customer backbone port) na B-komponent BEB.

Dalším je typ B, Backbone Edge Bridge (B-BEB) na propojení uvnitř PBBN. Obsahuje pouze B-komponentu, která ověřuje přijaté I-SID a mapuje rámce k odpovídající B-VLAN. Propojuje buď I-BEB nebo jiné B-BEB přes CBP port. Přes port PNP (Provider Network port) propojuje BCB (Backbone Core Bridge).

Posledním je IB typ Backbone Edge Bridge (IB-BEB) typicky na propojení 802.1ad s 802.1ah. Z předešlých dvou komponent obsahuje jednu nebo více I-komponent a jednu B-komponent. B-komponent provádí bridgování (přepínání) v sítí poskytovatele

za pomocí B-MAC, B-VLAN. I komponent provádí přepínaní v síti zákazníka za pomocí C-MAC, S-VLAN.



Obrázek 2.5: Zobrazení topologie PBBN s PBN

Na obrázku 2.5 jsou zobrazeny IB – BEB. Konkrétně typy IB – BEB obsahují jeden nebo více I-komponentů a jeden B-komponent. Komunikace probíhá tímto způsobem.

1. Příchozí rámec z PBN přichází na I-komponent ve formě, kterou již známe. Pokud tedy takovýto rámec přijde na rozhraní, přijde na CNP (Customer network port) kde klasifikuje příchozí rámce na základě S-VLAN a určuje shodující se I-SID.
2. I-komponent rozezná C-SA (zdrojovou MAC adresu zařízení), vyhledá C-DA (koncovou adresu zařízení) a podle toho určí správnou B-DA.
3. Zde je rámec již na portu PIP (Provider instance port). Rámec se zabalí do tvaru PBB a přidá B-SA.
4. Na portu CBP (Customer backbone port) se klasifikují příchozí rámce na základě I-SID. Přidá B-TAG pomocí něhož probíhá komunikace v PBBN.
5. B- komponent rozezná B-SA a vyhledá B-DA a určí správný PNP (Provider network port).
6. PNP filtruje odchozí rámce na základě B-TAG a předává nevyfiltrované rámce na BCB. [11,12,13,14]

2.2.3 Příklad konfigurace PBB na směrovači Cisco řady ASR 9000

Konfigurace PBB na směrovači Cisco se skládá z několika kroků a není nijak jednoduchá, proto je nutné si předem vše důkladně promyslet a nejlépe udělat poznámky někam na papír. Je jednoduché se při takovéto konfiguraci někde ztratit a dohledávat zpětně, což při takové rozsáhlé konfiguraci zabírá drahocenný čas. Dále je nutné mít již nabyté nějaké znalosti ohledně Cisco zařízení a také znalost dané problematiky. Konfigurace je také volně dostupná na oficiálních stránkách firmy Cisco. [15]

Základní Ethernet Flow Points (EFP) nastavení CNP rozhraní

```
R_Test_PBB# configure t
```

Pomocí příkazu configure vstoupíme do konfiguračního režimu

```
R_Test_PBB(config)# interface GigabitEthernet 0/0/0/10.100
l2transport
```

Vzhledem k tomu, že konfigurace probíhá na routeru, je potřeba nastavit L2 přepínání na daném rozhraní kde, bude komunikace pomocí PBB probíhat

```
R_Test_PBB(config-subif)# encapsulation dot1ad 100
```

Nyní je nutné zadat ethertype a VLAN ID na daném portu. Tím nastavíme S-Tagged službu (802.1ad ↔ 802.1ah) jak je zobrazeno na obrázku 2.4.

```
R_Test_PBB(config-subif)# commit
```

Tímto příkazem potvrdíme a zapíšeme konfiguraci do „running-configuration“ souboru.

Konfigurace PBB Edge Bridge Domény a Service Instance ID

```
Krok 1: R_Test_PBB(config)# l2vpn
```

V konfiguračním módu vstoupíme do do L2VPN konfiguračního módu.

```
Krok 2: R_Test_PBB(config-l2vpn)# bridge group pbb
```

Vstoupíme do konfiguračního módu bridge skupiny „pbb“.

```
Krok 3: R_Test_PBB(config-l2vpn-bg)# bridge-domain pbb-edge
```

Nyní vstoupíme do konfiguračního módu skupiny „pbb-edge“ (lze napsat libovolné jméno)

```
Krok 4: R_Test_PBB(config-l2vpn-bg-bd)# interface
GigabitEthernet0/5/0/0.20
```

Zde přiřadíme VLAN ID a Ethertype na rozhraní. Tento EFP je považován za CNP pro Hraniční bridge (Edge Bridge).

```
Krok 5: R_Test_PBB(config-l2vpn-bg-bd)# pbb edge i-sid 1000
core-bridge pbb-core
```

V tomto kroku nastavíme bridge (most) doménu jako PBB Edge s Servisním identifikátorem (i-sid) 1000 a také přiřadíme Core Bridge doménu. Také v tomto kroku vstoupíme do PBB Edge konfiguračního sub-módu.

```
Krok 6: R_Test_PBB(config-l2vpn-bg-bd-pbb-edge)# commit
```

Potvrdíme konfiguraci a zapíšeme konfiguraci do „running-configuration“ souboru.

Konfigurace PBB Core Bridge domény

```
Krok: 1
```

Konfiguraci provádíme stejně jako v přechozí konfiguraci PBB Edge Bridge Domény až po krok 2.

```
Krok: 2 R_Test_PBB(config-l2vpn-bg-bd)# bridge-domain
pbb-core
```

V tomto kroku vstoupíme do konfiguračního módu dané bridge domény, v našem případě pojmenované „pbb-core“.

```
Krok 3: R_Test_PBB(config-l2vpn-bg-bd)# interface
GigabitEthernet0/5/0/0.20
```

Podobně jako v konfiguraci PBB Edge Bridge Domény v Kroku 4 přiřadíme VLAN ID a Ethertype na rozhraní.

```
Krok 4: R_Test_PBB(config-l2vpn-bg-bd)# pbb core
```

Nastavíme bridge doménu (v našem případě doménu „pbb-core“) jako PBB Core a vstoupíme do PBB Core konfiguračního sub-módu.

```
Krok 5: R_Test_PBB(config-l2vpn-bg-bd-pbb-core)# commit
```

Obdobně jako v přechozích případech potvrdíme konfiguraci a zapíšeme konfiguraci do „running-configuration“ souboru.

Konfigurace Backbone VLAN Tag pod PBB Core Bridge Doménu

```
Krok 1 :
```

Konfiguraci provádíme stejně jako v předchozí konfiguraci PBB Core Bridge domény až po krok 3.

```
Krok 2 : R_Test_PBB(config-l2vpn-bg-bd-ac)# interface
GigabitEthernet0/5/0/1.15
```

Přidá rozhraní do bridge domény (pbb-core), které dovolí paketům, aby byly směrovány a přijímány z jiných rozhraní, které jsou součástí stejné bridge domény (pbb-core). Tímto příkazem je rozhraní připojeno do skupiny této bridge domény (pbb-core).

```
Krok 3: R_Test_PBB(config-l2vpn-bg-bd)# pbb core
```

Nastavíme bridge doménu jako PBB Core a vstoupíme do PBB Core konfiguračního sub-módu. Tímto příkazem také vytvoříme vnitřní port Customer Bridge Port (CBP). Se všemi rozhraními patřícími k této bridge doméně (pbb-core) je zacházeno jako s Provicer Netwrok Port (PNP).

```
Krok 4: R_Test_PBB(config-l2vpn-bg-bd-pbb-core)# commit
```

Potvrdíme konfiguraci a zapíšeme konfiguraci do „running-configuration“ souboru.

Konfigurace zdrojové Backbone MAC adresy

```
Krok 1: R_Test_PBB(config)# l2vpn
```

```
Krok 2: R_Test_PBB(config-l2vpn)# pbb
```

Vstpupíme do PBB konfiguračního módu.

```
Krok 3: R_Test_PBB(config-l2vpn-pbb)#
backbone-source-address 0000.1111.22
```

Tímto příkazem nastavíme zdrojovou Backbone MAC adresu pro PBB.

```
Krok 4: R_Test_PBB(config-l2vpn-pbb)# commit
```

Potvrdíme konfiguraci a zapíšeme konfiguraci do „running-configuration“ souboru.

Konfigurace neznámé unicast (jednosměrové) Backbone MAC adresy pro PBB Esge Bridge doménu

```
Krok 1:
```

Nastavení provádíme stejně jako při konfiguraci PBB Edge Bridge Domény a Service Instance ID až po krok 5.

```
Krok 2: R_Test_PBB(config-l2vpn-bg-bd-pbb-edge)#
unknown-unicast-bmac 1.1.1
```

V tomto kroku nastavíme MAC adresu (1.1.1), která se bude přiřazovat neznámým Backbone MAC adresám.

```
Krok 3: commit
```

Potvrdíme konfiguraci a zapíšeme konfiguraci do „running-configuration“ souboru.

Konfigurace statické MAC adresy pro PBB Edge Bridge Doménu

```
Krok 1 :
```

Konfiguraci provádíme stejně jako v konfiguraci PBB Core Bridge domény až po krok 3.

```
Krok 2 : R_Test_PBB(config-l2vpn-bg-bd-ac)# interface
GigabitEthernet0/5/0/1.15
```

Přidá rozhraní do bridge domény (pbb-core), které dovolí paketům, aby byly směrovány a přijímány z jiných rozhraní, které jsou součástí stejné bridge domény (pbb-core). Tímto příkazem je rozhraní připojeno do skupiny této bridge domény (pbb-core).

```
Krok 3: R_Test_PBB(config-l2vpn-bg-bd)# pbb edge i-sid 1000
core-bridge pbb-core
```

V tomto kroku nastavíme bridge doménu jako PBB Edge se servisním identifikátorem (i-sid = 1000), přiřadíme ji k PBB Core Bridge Doméně a zároveň vstoupíme do konfiguračního sub-módu PBB Edge.

Tento příkaz také vytváří Virtual Instance Port (VIP), která přiřadí PBB Bridge Doménu ke specifikované PBB Core Bridge doméně (v našem případě „pbb-core“).

Dále je se všemi rozhraními pod touto Bridge Doménou zacházeno jako s Customer Network Port (CNP).

```
Krok 4: R_Test_PBB(config-l2vpn-bg-bd-pbb-edge)#
static-mac-address 0022.2222.2222 bmac 0033.3333.3333
```

Nastavíme staticky CMAC pro BMAC mapování pro PBB Edge sub-mód.

```
Krok 5: R_Test_PBB(config-l2vpn-bg-bd-pbb-edge)# commit
```

Potvrdíme konfiguraci a zapíšeme konfiguraci do „running-configuration“ souboru.

2.3 Shortest Path Bridging

Shortest Path Bridging (SPB) bylo standardizováno jako nástupce rodiny protokolů STP. Protokol STP neumožňuje používání více než jedné cesty. Jednoduše ostatní cesty zablokuje, aby nedocházelo v síti ke smyčkám. Aby tedy bylo možné využít tyto cesty, aniž by se musely tyto cesty (smyčky) blokovat, byl vyvinut protokol Shortest Path Bridging (SPB), který je náhradou za protokoly rodiny STP (STP, RSTP, MSTP) se kterými doposud musel pracovat i výše zmíněný PBB. SPB umožňuje všem cestám v sítí na spojové vrstvě, aby byly aktivní, umožňuje mnohem větší topologie na spojové vrstvě a disponuje rychlou konvergencí při změnách topologie (řádově 50-100 ms). [16,17]

SPB kombinuje datové cesty za pomocí protokolu IS-IS (Intermediate Systém to Intermediate System). IS-IS je běžně známý jako routovací protokol v sítích Cisco, zde byl ale upraven tak, aby pracoval na 2. vrstvě, nepotřebuje tedy mít pro komunikaci nakonfigurované IP adresy na uzlech pro vznik IS-IS spojení se sousedícími mosty. SPB podporuje více režimové operace na základě určení podle typu datové úrovně a způsobu jeho chování. SPB se dělí do dvou různých verzí. Jednou z nich je Shortest Path Bridging VID (SPBV) a je vytvořena spíše pro menší sítě VLAN (2-100 mostů). Také umožňuje tyto sítě VLAN distribuovat jako zátěž do různých nejkratších cest stromů (Shortest Path Trees). Výhodou SPBV je především zpětná kompatibilita s protokoly STP. Druhou verzí je Shortest Path Bridging MAC (SPBM) používané především v sítích PBBN (Provider Backbone Bridge Network) a je vytvořena pro mnohem větší sítě (2-1000 mostů).

Jak již bylo řečeno, SPB disponuje dvěma verzemi, jedním z nich je SPBV a druhým je SPBM. Pouze SPBM podporuje plnou virtualizaci s použitím 802.1ah MAC-in-MAC zapouzdřením. SPBV nabízí směrování nejkratšími cestami pouze však s omezením funkčnosti použití 802.1ad Q-in-Q tagováním pro zařízení, která nepodporují zapouzdření 802.1ah MAC-in-MAC. Virtualizace služeb SPBM je definovaná pomocí I-SID identifikátory, kde I-SID je jednoduše přidělen na BEB buď VLAN pro virtualizaci služeb na druhé vrstvě, všesměrové skupině pro virtualizaci všesměrových skupin nebo

VRF (Virtual Routing and Forwarding) pro virtualizaci služeb na třetí vrstvě (síťové vrstvě). V SPBM síti každý most propaguje svou vlastní unikátní MAC adresu s použitím IS-IS známou jako systém-id. Systém-id může být také manuálně přidělen pro snadnější řešení problémů na druhé vrstvě (spojové). Každá SPBM síť je přidružená k alespoň jedné backbone VLAN (B-VLAN) v síti s použitím SPBM. Obrovskou výhodou SPBM je to, že chrání všechny mosty v infrastruktuře (edge a core) proti tomu, aniž by o nich koncové stanice věděli, díky zapoudření MAC-in-MAC. [18,19]

2.3.1 Rozšíření protokolu IS-IS pro SPB

Shortest Path Bridging (SPB) 802.1aq rozšiřuje Intermediate System to Intermediate Systém (IS-IS) o Protocol Data Unit (PDU), která přenáší informace o síti SPB. Tato informace obsahuje identifikátory uzlů, stavy jednotlivých linek a informace o přilehlých mostech reprezentovanou sítí SPB. IS-IS protokol pracuje na druhé vrstvě. IS-IS protokol je založený na způsobu směrování podle stavu linek (link-state protokol). Při používání SPB dovoluje právě na druhé vrstvě používat více cest a rozkládat tak zátěž několika cestami se stejnou cenou linky (obdobně to funguje i s routovacími protokoly, to je však na síťové vrstvě). Použití je vhodné ve všech možných topologiích, především však v sítích se smíšenou topologií (mesh networks) Pokud uvažujeme takové sítě, které propojují státy a kontinenty, tak bývají velice rozlehlé. [18,19,20]

Kontrolní protokol SPB a rozšíření IS-IS používá čtyři hlavní části pro komunikaci.

- Prvním je část IS-IS Hello (IIH). Tato část se používá pro detekci sousedících uzlů, které jsou schopny provozovat SPB protokol. IIH je také používán pro výměnu a přehled informací. Tato informace obsahuje dva identifikátory (prioritu mostu a systémové-ID mostu) a informace o stavu linek mezi sousedícími mosty. SPB používá IIH pouze pro detekci sousedících uzlů.
- Druhou částí je SPB úroveň (SPB instance – SPB-Inst), která zajišťuje informace o uzlech. SP (shortest path) source ID je 20-bitová hodnota a obsahuje název uzlu. Priorita mostu (Bridge priority) je 16-bitová hodnota a se 4-bity System-ID tvoří identifikátor mostu (Bridge Identifier). Obsahuje informaci o počtu větví a VLAN-ID.
- Třetí částí je informace obsahující SPB-Link Metric (váhu nebo cenu linky) a Port Identifier (identifikátor portu).
- Čtvrtou částí je služba, která má informace o B-MAC adresách a I-SID. I-SID slouží v tomto případě k tomu, že pokud dva různé uzly používají stejný I-SID, mezilehlé uzly se nastaví tak, že mezi uzly se stejným I-SID budou přenášet data.

IS-IS počítá nejkratší cestu k MAC adrese páteřního (B-MAC) uzlu. B-MAC adresy jsou pomocí IS-IS propagovány jednou nebo více páteřními VLAN ID (B-VID). V součtu jsou rámce směrovány pomocí systémového-ID jako Backbone Source Adress (B-SA)

ke konkrétnímu uzlu který má Backbone Destination Adress (B-DA). SPB směrovací databáze (forwarding database - FDB) obsahuje kombinaci unicástových a mluticastových MAC adres. [17, 18]

2.3.2 Jak 802.1aq funguje

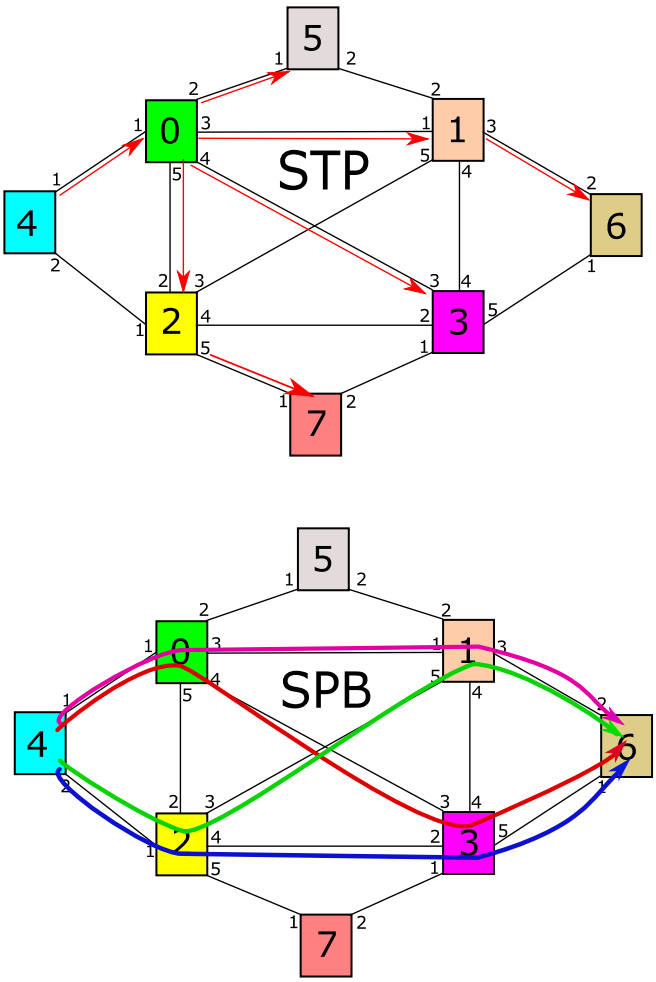
V SPBM určení MAC adres probíhá pomocí kontrolní úrovně, pomocí níž jsou distribuovány mezi mosty, které používají IS-IS. V případě SPBV se MAC adresy učí na datové úrovni, prozkoumáním obsahu příchozího rámce.

Směrování unicastových rámců probíhá následujícím způsobem. IS-IS určuje topologii, kde všechny hraniční mosty musejí vypočítat alespoň jednu větev, která je schopna dosáhnout všech koncových stanic v síti. Smysl je v tom, že zdrojový most může být zároveň kořen větve pro každý příchozí rámec a celá cesta od začátku do konce je vypočtená a známá ještě před tím, než jsou rámce směrovány. Výpočet těchto větví je určen takovým způsobem, aby všechny mezilehlé mosty měli stejný „názor“ o každé větvi mezi sebou (všechny mosty se dohodnou na stejné cestě ke stejnému cíli). Proto rámec potřebuje pouze označení zdrojovou a cílovou MAC adresou mostu a všechny mezilehlé mosty směrují rámec podle předem dohodnuté cesty, aniž by musely dělat rozhodnutí, jak s daným rámcem naložit (přes který port ho poslat dál). Tyto cesty jsou vypočteny na základě informací od protokolu IS-IS. To má za následek, komunikace po větvi, které jsou takovýmto způsobem vytvořeny, je oběma směry stejná (provoz mezi dvěma hosty je symetrický) jak je zobrazeno na obrázku 2.6 SPB. Zabalení rámce je obdobné jako u protokolu PBB. Použije se B-SA, B-DA a I-TAG ve kterém je obsaženo I-SID. [16,17,19,20]

Směrování všesměrových nebo vícesměrových rámců probíhá obdobně jako u rámců unicastových. Rozdíl je pouze v tom, že rámce nejdou skrze síť pouze k jednomu cíli, ale jdou ke všem cílům dané větve, které používají stejný ISID. [10,18,19,20]

2.3.3 ECMT

Equal Cost Multiple Trees (ECMT) je algoritmus, který hledá nejkratší cesty mezi dvěma mosty (cesty s nejnižší cenou). Je zde také mnohem větší předvídatelnost ohledně datového toku, protože cesty oběma směry jsou symetrické. Tyto nalezené cesty jsou následně uloženy do kontroléru a provoz je mezi tyto cesty následně rozdělen. ECMT dovoluje výrazně větší využití topologie oproti STP, kde se všechny redundantní cesty blokovaly.



Obrázek 2.6: Možné cesty protokolu STP a SPB

Jak je vidět na obrázku 2.6, rozdíl mezi STP a SPB je zřejmý. Zatímco STP blokuje redundantní cesty, aby nedocházelo ke smyčkám v síti, při použití SPB jsou všechny cesty použitelné, jak je zobrazeno na obrázku 2.6 (je možné použít i cesty na mostu 5 a 7). Na obrázku 2.6 je zobrazeno použití cest barevnými čárami, kde SPB datový tok se rozdělí mezi tyto čtyři cesty a rozdělí tak zátěž. Jak je vidět na obrázku 2.6 rozdělení zátěže se provádí mezi uzly 0, 1, 2 a 3. Existuje několik druhů algoritmů pro výpočet různých ECMT. [16,17,18,19,20]

2.3.4 Rozdíl mezi STP a SPB

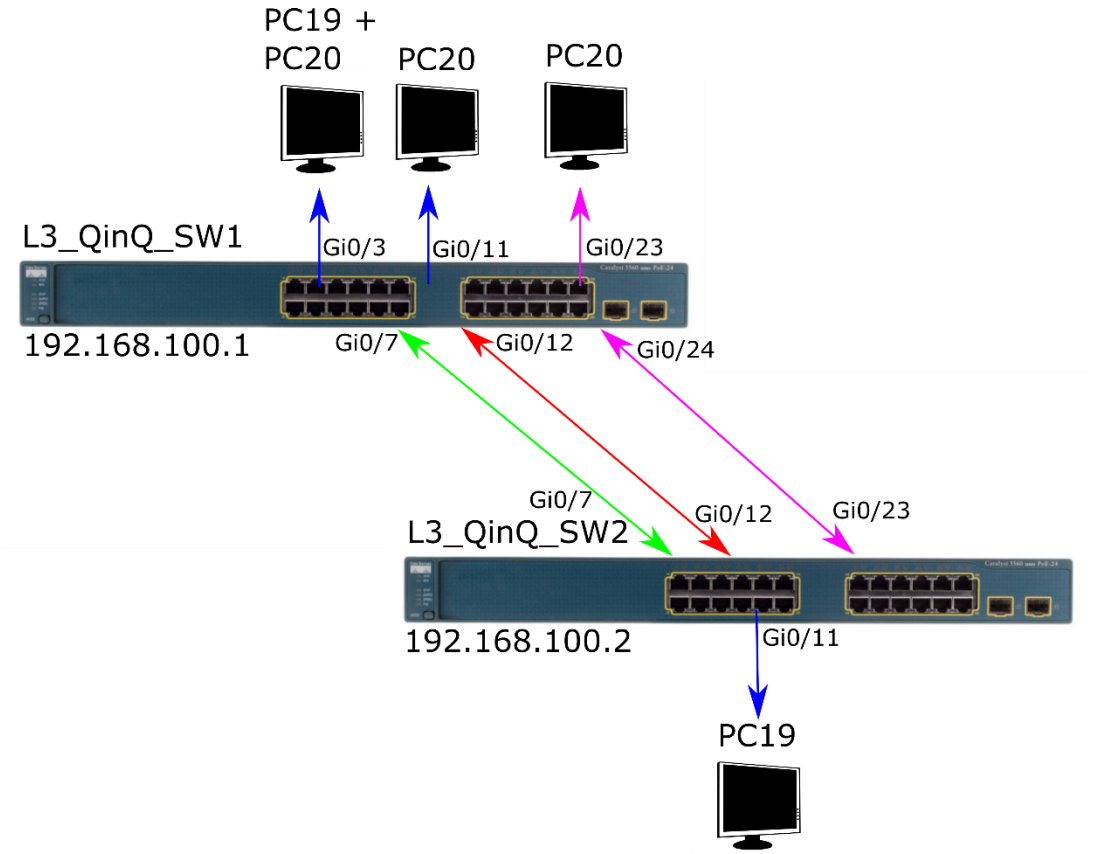
V Tabulka 1 je zobrazeno porovnání protokolů Shortest Path Bridging a Spanning Tree Protokol.

Tabulka 1: Porovnání STP a SPB

Spanning Tree Protocol (STP)	Shortest Path Bridging (SPB)
Záložní (redundantní) cesty jsou blokovány.	Výběr redundantních cest je založen na algoritmu ECMT. Všechny cesty je možné použít
Používá spanning tree algoritmus pro výpočet cesty s použitím BPDU jednotek.	Používá protokol IS-IS pro určení nejkratší cesty.
Směruje rámce větví na základě znalostí CAM tabulky.	Po výpočtu cesty pro celou cestu až k cíli je jednoduché předpokládat, kudy daný provoz poteče. Cesty oběma směry jsou symetrické.
STP nemá žádný mechanismus na ochranu proti smyčkám, jednoduše blokuje redundantní cesty.	SPB má Reverse Path Forwarding Check (RPFC) na ochranu proti smyčkám.
Směrování všesměrového a vícesměrového provozu je možné pouze jednou větví.	Směrování je možné více větvemi a cesty všesměrového a vícesměrového provozu jsou stejné.
STP nemá žádné extra zapouzdření.	SPB je možné použít jako SPBV (Q-in-Q) nebo jako SPBM (MAC-in-MAC).

3 Tvorba laboratorní úlohy

Tvorba laboratorní úlohy bude tvořena ze znalostí získaných z teoretické části Bakalářské práce a měla by se zabývat problematikou QinQ. V úloze bude student konfigurovat podle návodu propojení přepínače pomocí trunku a poté pomocí QinQ. Ze získaných měření bude mít student za úkol tyto dvě technologie porovnat a měl by být schopen popsat hlavní rozdíly obou použitých technologií. Měření bude probíhat pomocí programu Wireshark, který se hodí snad na vše, co jo třeba změnit. Během měření by zde měl student vysledovat, co dané technologie dělají při komunikaci právě pomocí programu Wireshark. Určit to lze s vysledovaného typu EtherType.



Obrázek 3.1: Zapojení laboratorní úlohy v učebně

Na obrázku 3.1 je zobrazené zapojení laboratorní úlohy. Laboratorní úloha bude probíhat na dvou PC (jeden virtuální PC se systémem Ubuntu Linux a druhý PC s virtuálním systémem Windows XP), ze kterých bude probíhat komunikace a měření pomocí programu Wireshark. Dalšími zařízeními jsou přepínače firmy Cisco C3560, které podporují QinQ. Samotný návod k laboratorní úloze je uveden v příloze. Oba PC používají 2 síťové karty Intel, které podporují komunikace pomocí tagovaných VLAN. K samotnému měření bylo nutné v registrech PC, na kterém bude probíhat měření přidat DWORD MonitorMode a jeho hodnotu nastavit na „1“. To nám umožní obdržení špatných/CRC paketů. Tento mód také neodstraňuje VLAN tag z paketů.

Závěr

V bakalářské práci jsem se věnoval protokolům pro návrh datové sítě. Probral jsem protokol STP, který je sice zastaralý, ale stálé velmi hodně používaný u menších poskytovatelů internetového připojení. Probral jsem sítě VLAN, jak fungují a jaké přinášejí výhody v případě virtuálně oddělených sítí.

V druhé části jsem se věnoval moderním protokolům pro návrh datové sítě zaměřené především na větší poskytovatele. Probral jsem Provider Bridging, který je rozdílný od VLAN v tom, že zabalí jednu síť VLAN do jiné VLAN. Provider Backbone Bridging je protokol, který používají nadnárodní společnosti, který poskytují konektivitu mezi státy či kontinenty. Výhoda tohoto protokolu spočívá především v tom, že poskytuje naprosté oddělení od sítě, která do ní vstupuje (nemusí ukládat všechny MAC adresy zařízení uvnitř sítě providera, kterému poskytuje konektivitu). V poslední radě je zde SPB, který používá nejkratší cesty na základě stavu linky a protokolu IS-IS. Na konci jsem také provedl srovnání protokolů Spanning Tree Protocol a Shortest Path Bridging, ze které je zřejmé, že protokol SPB je výrazně efektivnější oproti STP protokolu.

V poslední kapitole je stručně popsána laboratorní úloha, ke které je v příloze postup a také příklady řešení jednotlivých úkolů. Na přiloženém DVD jsou také použité programy a základní konfigurační soubory přepínačů. Také je tam podrobný návod pro nastavení MonitorMode, který je nutné v registrech OS nastavit pro monitorování VLAN v síti.

Literatura

[1] DONAHUE, Gary A. *Kompletní průvodce síťového experta*. Vyd. 1. Brno: Computer Press, 2009, 528 s. ISBN 978-80-251-2247-1.

[2] SEIFERT, Rich a James EDWARDS. *The all-new switch book: the complete guide to LAN switching technology*. 2nd ed. Indianapolis, IN: Wiley Pub., c2008, xxxi, 784 p.

[3] *VLAN (1) - historie a význam* [online]. http://www.svetsiti.cz, 2003, 2. dubna 2003 [cit. 2015-11-16]. Dostupné z: http://www.svetsiti.cz/CLANEK.ASP?CID=VLAN-1-HISTORIE-A-VYZNAM-242003

[4] IBM Corporation 1990, 2008. *VLAN-Aware or VLAN-Unaware* [online]. [cit. 2015-12-03]. Dostupné z: http://www-01.ibm.com/support/knowledgecenter/SSB27U_5.4.0/com.ibm.zvm.v54.hcpa6/hcsc9b3180.htm%23wq102

[5] BOUŠKA, Petr. *VLAN - Virtual Local Area Network* [online]. 2007, 02.06.2007 [cit. 2015-12-03]. Dostupné z: http://www.samuraj-cz.com/clanek/vlan-virtual-local-area-network/

[6] Wikipedia. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2006, 21. 4. 2015 [cit. 2015-12-03]. Dostupné z: https://cs.wikipedia.org/wiki/VLAN

[7] BOUŠKA, Petr. *Cisco IOS 9 - Spanning Tree Protocol* [online]. 2007, 03.05.2009 [cit. 2015-12-03]. Dostupné z: http://www.samuraj-cz.com/clanek/cisco-ios-9-spanning-tree-protocol/

[8] BOUŠKA, Petr. *Cisco IOS 10 - Rapid Spanning Tree Protocol* [online]. 2007, 01.09.2007 [cit. 2015-12-03]. Dostupné z: http://www.samuraj-cz.com/clanek/cisco-ios-10-rapid-spanning-tree-protocol/

[9] IEEE 802.1ah-2008. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2015-12-09]. Dostupné z: https://en.wikipedia.org/wiki/IEEE_802.1ah-2008

[10] CISCO SYSTEMS, INC. *IEEE 802.1ad Support on Provider Bridges* [online]. 2010, May 26 2011 [cit. 2015-12-03]. Dostupné z: http://www.cisco.com/c/en/us/td/docs/ios/cether/configuration/guide/ce_cfm-ieee_802_1ad.html#wp1060656

[11] Understanding PBB. *Google.com* [online]. [cit. 2015-12-09]. Dostupné z: https://sites.google.com/site/amitsciscozone/home/pbb/understanding-pbb

[12] *Implementing IEEE 802.1ah Provider Backbone Bridge* [online]. 2010, May 26 2011 [cit. 2015-12-03]. Dostupné z: http://www.cisco.com/c/en/us/td/docs/routers/asr9000/software/asr9k_r4-2/lxvpn/configuration/guide/lesc42book/lesc42pbb.html

[13] *IEEE 802.1ah on Provider Backbone Bridges* [online]. 2010, May 26 2011 [cit. 2015-12-03]. Dostupné z: http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/cether/configuration/15-s/ce-15-s-book/ce-mac-evc-pbb.html

[14] Leveraging the Benefits of Provider Backbone Bridges [online] [cit. 2015-12-03] Dostupné z: https://www.google.cz/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0ahUKEwii6q_HtJ_JAhXhmHIKHbGYCjEQFggiMAE&url=http%3A%2F%2Fwww.brocadechina.com%2Fdownload%2FLeveraging_Benefits_PBB_WP-01.pdf&usg=AFQjCNHyQn7LoMfLn3uhoeLBYRQ7k29yKw&bvm=bv.107763241,bs.1,d.d24&cad=rja

[15] Ethernet Interfaces Commands, Cisco [online] Dostupné z:http://www.cisco.com/c/en/us/td/docs/routers/crs/software/crs_r5-1/lxvpn/command/reference/b_vpn_cr51xcrs/b_vpn_cr51xcrs_chapter_00.html

[16] IEEE 802.1aq. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2015-12-09]. Dostupné z: https://cs.wikipedia.org/wiki/IEEE_802.1aq

[17] IEEE 802.1aq. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2015-12-09]. Dostupné z: https://en.wikipedia.org/wiki/IEEE_802.1aq

[18] FEDIK D., ASHWORD-SMITH P., ALLAN D., BRAGG N., UNBEHAGEN P., *IS-IS Extension Supporting IEE 802.1aq Shortest Path Bridging* [online]. [cit. 2015-12-09]. Dostupné z: https://tools.ietf.org/html/rfc6329#section-4.1

[19] CHANG Y., *Design and Implementation of Shortest Path Bridging For Network Simulator 3* [online]. [cit. 2015-12-09]. Dostupné z: www.sce.carleton.ca/.../Yoonsoon-Thesis-Final.pdf

[20] *Shortest Path Bridging (802.1aq) Technical Configuration Guide* [online]. [cit. 2015-12-09]. Dostupné z: https://support.avaya.com/css/P8/documents/101008798

Seznam symbolů, veličin a zkratek

LAN	Local Area Network
MAN	Metropolitan Area Network
WAN	Wide Area Network
VLAN	Virtual Local Area Network
VPN	Virtual Private Network
IEEE	Institue if Electrical and Electronics Engineers
PCP	Priority Code Point
CFI	Canonical Format Indicator
VID	VLAN identifier
L3	Layer 3
STP	Spanning Tree Protocol
MSTP	Multiple Spanning Tree Protocol
RSTP	Rapid Spanning Tree Protocol
BPDU	Bridge Protocol Data Unit
ID	Identifier
MAC	Media Access Control
MSTI	Multiple Spanning Tree Instance
CAM	Content Addressable Memory
PB	Provider Bridging
PBB	Provider Backbone Bridging
SPB	Shortest Path Bridging
C-TAG	Customer tag
S-TAG	Service tag
BCB	Backbone Core Bridge
BEB	Backbone Edge Bridge
B-BEB	B type of Backbone Edge Bridge

I-BEB	B type of Backbone Edge Bridge
IB-BEB	IB type of Backbone Core Bridge
PNP	Provider Network Port
CBP	Customer Backbone Port
CNP	Customer Network Port
C-SA	Customer Source Adress
C-DA	Customer Service Adress
B-DA	Backbone Destination Adress
B-SA	Backbone Source Adress
IS-IS	Intermediate System to Intermediate Systém
SPBV	Shortest Path Bridging VID
SPBM	Shortest Path Bridging MAC-in-MAC
IIH	IS-IS Hello
ECMT	Equal Cost Multiple Trees

Seznam příloh

Příloha 1 – Laboratorní úloha zahrnující tunk, QinQ a přihlašovací protokoly telnet a SSH

Příloha 2 – Vzorové řešení laboratorní úlohy.

Příloha 3 – Přiložené CD obsahuje konfigurační soubory obou přepínačů, návod na nastavení síťových karet pro sledování VLAN a elektronickou verzi bakalářské práce.

Příloha 1

Laboratorní úloha

Cíl

Cílem úlohy je seznámit se s principem fungování technologií QinQ IEEE 802.1ad a porovnání s technologií Vlan trunking protocol. Jejich porovnání bude probíhat na základě znalostí získaných měřením pomocí programu Wireshark. V závěru je po studentovi vyžadováno tyto technologie porovnat. V poslední řadě se student přihlásí pomocí telnet a SSH a provede analýzu pomocí programu Wireshark.

Vybavení pracoviště

2x počítač se dvěma síťovými kartamy

2x L3 Přepínač Cisco 3560

Virtuální Windows XP

Virtuální Linux Ubuntu

Úkoly

1. Nastavte na přepínači Cisco C3560 Vlan trunking protocol (VTP) a úspěšně proveďte komunikaci pomocí PC a měření výsledné komunikace pomocí programu Wireshark.
2. Nastavte na přepínači Cisco C3560 komunikaci tak, aby probíhala pomocí QinQ. Měření proveďte stejně jako u měření trunku.
3. Nastavte zabezpečené připojení SSH na oba přepínače Cisco a pomocí programu Wireshark analyzujte oba způsoby připojení (Telnet a SSH) k přepínači.

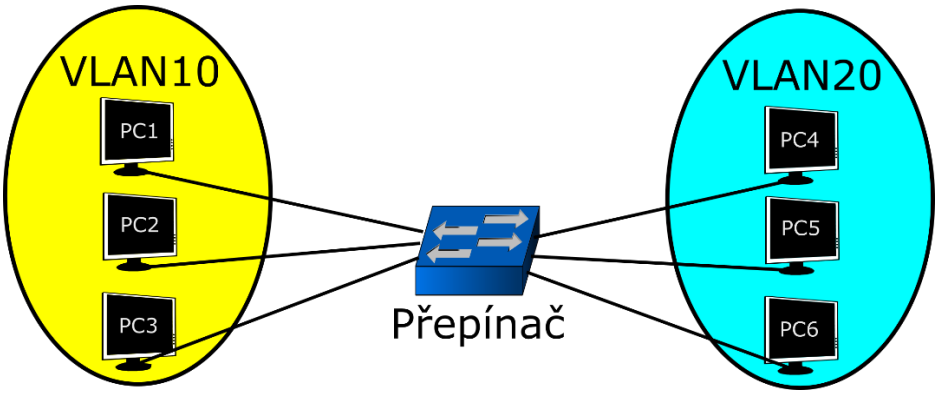
Teoretický úvod

K oddělení sítí se dříve používali přepínače, to však znamenalo, že pro oddělení každé skupiny (kanceláře účetním, kanceláře technici…) bylo nutné použít přepínač pro každou skupinu zvlášť. To se tak dělalo z důvodu zabezpečení, vzhledem k tomu, že vrstva L2 je náchylná na odposlechy. To však mělo za následek nadbytečné používání přepínačů, ale také možnost vzniku smyček (pokud přepínače již nepoužívali STP) například neodbornou manipulací s přepínači. Jako společné zařízení se pak používal směrovač (router).

Virtuální sítě

Levnější, efektivnější a snadnější pro údržbu jsou již v dnešní době na takovéto instalace používány přepínače (L2 i L3 přepínače), na kterých je možné nastavit sítě

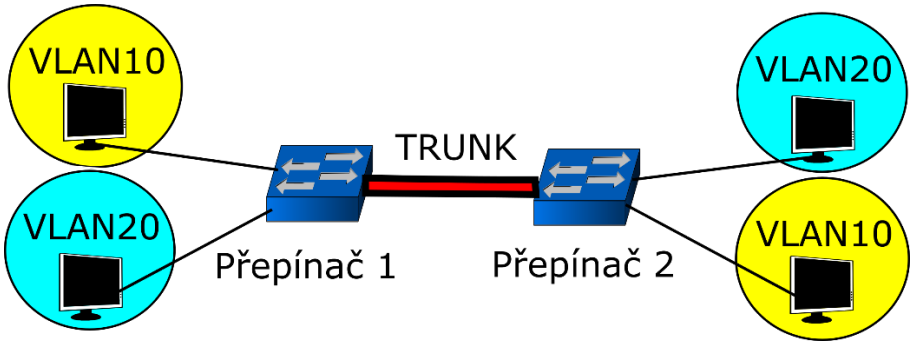
VLAN, případně VTP protokol. Technologie VLAN (Virtual Local Area Network) přišla o rok později (roku 1995) po uvedení přepínačů na trh. Samozřejmě si ihned získaly pozornost provozovatelů sítí ne jenom z toho důvodu, že jim byla poskytnuta lepší možnost managementu, ale i z důvodu úspor. VLAN je technologie, která umožňuje oddělení fyzického spojení od logického (nezávisle na fyzickém uspořádání). Uživatelé v této síti jsou stále fyzicky propojeni, jako to je to u sítí LAN s tím rozdílem, že je nutné, aby mezi nimi byl směrovač pro komunikaci mezi sebou. Sítě VLAN používají VLAN tag, který je zobrazen na obrázku 4.3. VLAN tag se používá k označovaní virtuálních sítí a je možno použít až 4096 různých VLAN v jedné síti.



Obrázek 4.1: Komunikace pomocí sítí VLAN

Na obrázku 4.1 je zobrazena komunikace pomocí jednotlivých zařízení v síti pomocí VLAN. Máme zde dvě sítě VLAN, jedna je VLAN 10 a druhá je VLAN 20. V tomto případě nemůžou komunikovat zařízení mezi jednotlivými sítěmi VLAN a dosáhli jsme tak logického oddělení jednotlivých sítí.

Přepínač v tomto případě funguje tak, jakoby zde byli dvě rozdílné sítě, každá fyzicky oddělená a každá na jiném přepínači. Ač jsou tyto sítě VLAN na jenom fyzickém přepínači, komunikace mezi jednotlivými sítěmi VLAN je možná pouze v případě, že by zde byl směrovač.

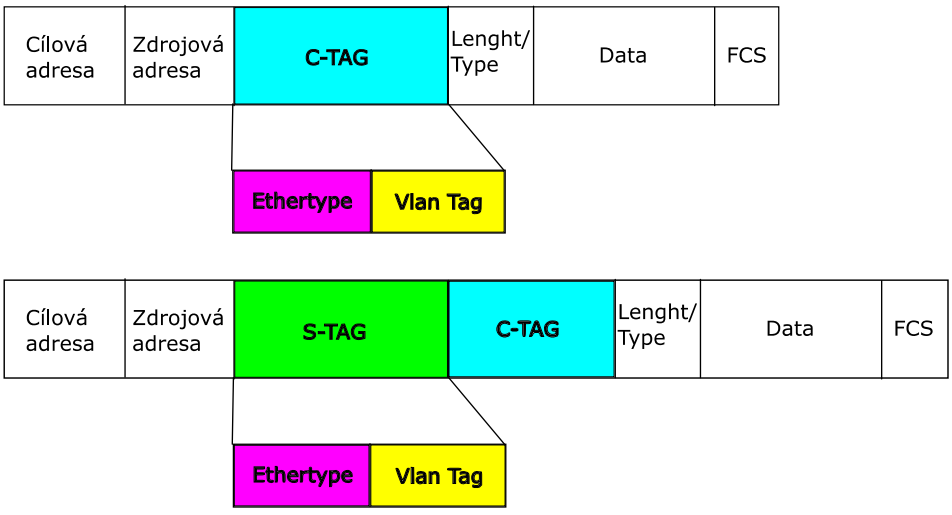


Obrázek 4.2: Komunikace pomocí protokou VTP

VLAN Trunking protocol (VTP) je prostředek, s jehož pomocí lze na centrálních zařízeních (často to bývá L3 přepínač, nebo směrovač) nastavit názvy a čísla sítí VLAN, přičemž výslednou konfiguraci lze distribuovat na ostatní zařízení. Jako trunk se označuje rozhraní, které je schopno přenést několik sítí VLAN současně. Pro přenos pomocí trunku potřebujeme tedy minimálně 2 přepínače. Komunikaci zobrazenou na obrázku 4.2 si můžeme představit tak, že přepínač 1 je v budově 1 a přepínač 2 je v budově 2. Tyto dvě budovy jsou propojeny trunkem. Tím je umožněno ze PC, které jsou označeny VLAN 20 mohou mezi sebou komunikovat, přestože jsou oba v jiné budově. Stejně takováto komunikace platí pro PC označené VLAN10. Komunikace probíhá odděleně mezi VLAN10 a VLAN20. Tím se ušetřila jak kabeláž, tak přepínače.

Service provider VLAN (QinQ)

U větších internetových providerů, může dojít časem k tomu, že jim dojdou možnosti používání sítí VLAN, kterých je celkem 4096. Takový problém se dá samozřejmě vyřešit tím, že do dané lokality, kde již nestačí celkový počet 4096 sítí VLAN přidáme směrovač a dál již budeme pomocí směrování komunikaci šířit dál. Pokud však takovéto řešení nechceme, nabízí se zde řešení pomocí technologie QinQ, která nabízí další zapouzdření do určité VLAN, kterou si zvolí provider. QinQ defakto znamená, že jedna VLAN je zabalená v druhé VLAN. Takto použitá konfigurace nabízí 4096 x 4096 možností, jak s těmito VLANami naložit. Komunikace s touto technologií funguje stejně jako je to u sítí VLAN s tím rozdílem, že v jedné VLAN přenášíme další VLAN. Nevýhodou PB je to, že přepínače musí být schopny pojmout všechny adresy MAC do svých CAM tabulek. Na obrázku 4.3 jsou zobrazeny rámce VLAN a QinQ.



Obrázek 4.3: Rámec pro komunikaci promocí QinQ

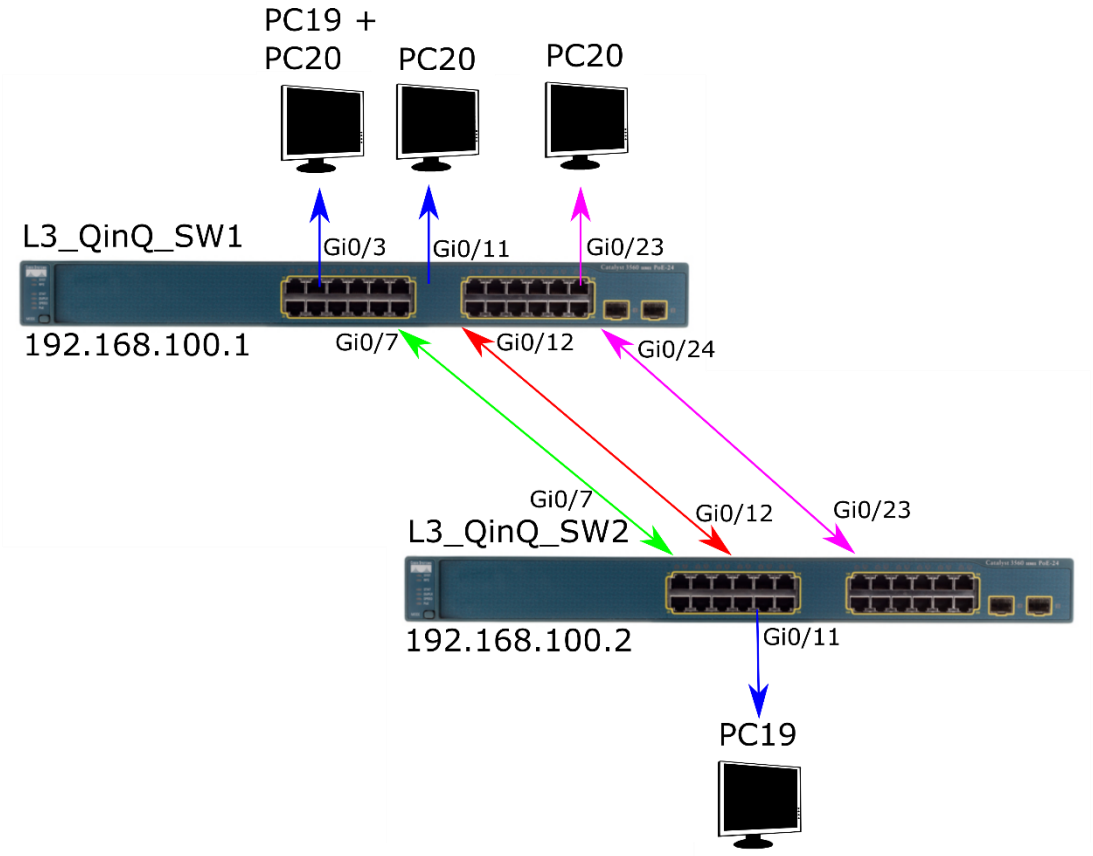
Jako první je na obrázku 4.3 zobrazený rámec při komunikaci v sítích VLAN. Ethertype v tomto případě vyznačuje to, že se jedná o hlavičku s VLAN tagem. Ethertype má v tomto případě hodnotu 0x8100. VLAN tak je volitelný od 1 do 4096.

Níže v obrázku 4.4 je rámec typický pro komunikaci pomocí QinQ. Ethertype zde opět označuje, že se jedná o hlavičku s VLAN tag, ale v tomto případě má Ethertype hodnotu 0x8a88, která je standardem pro QinQ. U přepínačů Cisco se však tato hodnota ne vždy dodržuje. U zařízení Cisco může mít QinQ Ethertype hodnotu 0x8100, 0x9100, 0x9200 a 0x8a88. Záleží na typu zařízení. Na některých přepínačích (L2 i L3) lze tuto hodnotu ručně změnit, v této laboratorní úloze to však přepínače neumožňují.

Zabezpečené a nezabezpečené připojení

Telnet se ve většině případů (obzvlášť u zařízení Cisco) používá pro přihlášení. To však nese svá rizika, především v bezpečnosti. Proto je vhodné používat přihlášené pomocí SSH, pomocí kterého přihlášené probíhá šifrovaně. Telnet je protokol používaný pro přihlášení se na vzdálený PC. Primární nevýhodou protokolu telnet je absence šifrování při používaní tohoto protokolu. Telnet nešifruje ani hesla při přihlašování se na vzdálený PC, to nese právě tu nevýhodu, že každý v sítí může odposlouchávat komunikaci na síti a zachytávat ji, tyto údaje pak může zneužít. Dnes se již použití protokolu telnet nedoporučuje právě z hlediska bezpečnosti. K tomuto účelu byl vyvinut protokol SSH, který veškerou komunikaci šifruje. V úloze si pomocí obou metod (telnet i SSH) provedeme přihlášení na přepínače a provedeme měření pomocí programu Wireshark a odhalíme slabiny při přihlašování pomocí telnet. Protokol telnet používá pro svůj přenos port 23 a SSH používá port 22. Tyto porty jsou defaultně v programu Putty (použitý v lab. úloze) nastaveny.

Zapojení laboratorní úlohy



Obrázek 4.4: Zapojení laboratorní úlohy

Na obrázku 4.1 je zobrazeno zapojení laboratorní úlohy. Zapojení laboratorní úlohy nijak neměňte!

Zapojení zobrazené modrou šipkou ⟷ mezi přepínačem L3_QinQ_SW2 a PC19 je připraven port GigabitEthernet0/11 pro konfiguraci QinQ. Mezi přepínačem L3_QinQ_SW1 a PC19 + PC20 je připraven port GigabitEthernet 0/3 pro nastavení trunku. Port Gi0/11 je mezi přepínačem L3_QinQ_SW1 a PC20 připraven pro konfiguraci QinQ.

Zapojení zobrazené fialovou šipkou ⟷ je základní zapojení laboratorní úlohy, kde právě přes PC20 probíhá prvotní nastavení. Na přepínači L3_QinQ_SW1 a portu GigabitEthernet0/23 je později v laboratorní úloze tento port nataven jako monitoring a již na přepínače přes tento port není přístup (přístup je umožněn po zrušení konfigurace).

Zapojení zobrazené červenou šipkou ⟷ je zapojení připravené pro konfiguraci VTP mezi přepínači L3_QinQ_SW1 a L3_QinQ_SW2 pro přenos QinQ. Pro toto zapojení jsou určeny porty GigabitEthernet0/12 na obou přepínačích.

Zapojení zobrazené zelenou šipkou ⟷ je zapojení připravené pro konfiguraci VTP mezi přepínači L3_QinQ_SW1 a L3_QinQ_SW2 pro přenos VLAN 600. Pro toto zapojení jsou určeny porty GigabitEthernet0/7 na obou přepínačích.

Prvotní nastavení a přístup na přepínače:

Neukládejte nastavení na žádném z přepínačů! Předejdete tak k případným chybám v konfiguraci, kdy přepínač jednoduše restartujete.

Při přihlášení pomocí telnet je **heslo: bars (**neplatí při přístupu přes consolu**).**

Pro přístup do enable módu je **heslo: qinq.**

Nejprve si spusťte na obou PC ve VirtualBoxu Windows XP na PC19 (název QinQ) a Linux Ubuntu na PC20 (název UbuntuQinQ). Všechny přepínače by měli být na začátku úlohy v základním nastavení a přístupné pouze z PC20. Konfigurace vždy začíná v konfiguračním módu. Pokud se přes PC20 nemůžete dostat na přepínače, pravděpodobně je na přepínačích konfigurace z předešlé úlohy. V takovém případě postupujte takto:

1) Pokud se nemůžete přihlásit na přepínače, pomocí výše uvedených přihlašovacích údajů, přepínače jsou pravděpodobně z předešlé úlohy nastaveny na přihlašování pomocí SSH. V takovém případě použijte **user: bars, heslo: qinq**, přistup do enable módu **heslo: qinq.** Dále pracujte podle postupu 2) a 3).
2) Nalogujte se na přepínač **L3_QinQ_SW2** přes PC19 pomocí putty (pokud nelze, tak zkuste PC20 a terminál), která je na ploše virtuálního WinXP pomocí telnetu aadresy **192.168.99.2**. Jakmile se přihlásíte do přepínače pomocí výše zmíněných přihlašovacích údajů, napište příkaz **`reload`** a pokud se přepínač zeptá, jestli chcete uložit konfiguraci, napište **`no`**.
   System configuration has been modified. Save? [yes/no]: no
3) Nalogujte se na přepínač **L3_QinQ_SW1** přes PC19 pomocí putty a adresy **192.168.99.1** a proveďte **reload** stejně jako na přepínači **L3_QinQ_SW2**.

Konfiguraci doporučuji dělat podle návodu, protože kabeláž zapojená v racku je zapojená přesně tak jak je na obrázku a podle toho bude probíhat také návod na konfiguraci. Veškerá konfigurace probíhá již v konfiguračním módu:

```
L3_QinQ_SWX # configure terminal
```

Postup konfigurace:

Úkol 1

Konfigurace probíhá na **PC20** kde pomocí Terminálu a příkazu **telnet 192.168.100.1** se dostanete na přepínač L3_QinQ_SW1. Mezi tím si na **PC20 nebo PC19** spusťte

pomocí příkazové řádky (terminálu v Linuxu) ping na adresy **192.168.99.1** a **192.168.99.2**. Na přepínači **L3_QinQ_SW1** nastavte port **GigabitEthernet0/3** jako trunk pro VLAN 600 a do této VLAN přidejte adresu z rozsahu **192.168.99.X**. Postup je následující:

```
L3_QinQ_SW1(config)# interface GigabitEthernet 0/3
```

Tímto příkazem se dostaneme do konfigurace rozhraní GigabitEthernet0/3, na které je připojen PC19 a PC20 trunkem.

```
L3_QinQ_SW1(config-if)# switchport trunk encapsulation
dot1q
```

Tímto příkazem zajistíme to, že port bude možné nastavit do trunk módu. Označení dot1q je to samé jako 802.1Q. Pomocí encapsulation je zajištěno zapouzdřování a dot1q znamená, že to bude zapouzdřování rámce do VLAN, kterou si zde nastavíme.

```
L3_QinQ_SW1(config-if)# switchport mode trunk
```

Port nastavíme do módu trunk (VTP).

```
L3_QinQ_SW1(config-if)# switchport trunk allowed vlan 600
```

Zde nastavíme jaké VLAN chceme propouštět a přijímat na tomto portu, v našem případě VLAN 600, kde už od PC19 a PC 20 chodí otagované rámce VLAN 600. Jiné rámce otagované jinými VLAN tímto portem neprojdou. Lze je ale přidat pomocí příkazu `switchport trunk allowed vlan add XXXX`.

```
L3_QinQ_SW1(config-if)# no shutdown
```

Port je z důvodu nechtěného zapojení smyček zakázaný, proto je nutné ho povolit.

```
L3_QinQ_SW1(config-if)# exit
```

Nyní máme nastavenou konfiguraci pro komunikaci mezi přepínačem **L3_QinQ_SW1** a PC20 a PC19. Dále je však nutné zajistit adresaci, bez toho bychom se na přepínač nedostali.

```
L3_QinQ_SW1(config)# interface vlan 600
```

Tímto příkazem se dostaneme do konfigurace rozhraní VLAN 600.

```
L3_QinQ_SW1(config-if)# ip address 192.168.99.1
255.255.255.0
```

Na rozhraní VLAN 600 nastavíme IP adresu 192.168.99.1 a masku 255.255.255.0. Adresa je ve stejném rozsahu jako adresa na PC19 a PC20.

```
L3_QinQ_SW1(config-if)# exit
```

Vystoupíme z konfigurace rozhraní VLAN 600.

Nyní byste měli být schopni (cca po 20 sekundách od nastavení adresy) pingat z PC20 nebo PC19 adresu 192.168.99.1. Pokud ne, zkontrolujte konfiguraci přepínače.

Nastavení trunku mezi přepínači

Dalším krokem je konfigurace trunku mezi přepínači **L3_QinQ_SW1** a **L3_QinQ_SW2** a přidání adresy z VLAN 600 na **L3_QinQ_SW2. Konfigurace stále probíhá z PC20**. Obdobně jako v přechozím návodu, se na přepínač L3_QinQ_SW2 dostaneme přes adresu **192.168.100.2** Na obou přepínačích je pro nastavení trunku mezi nimi připraven port GigabitEthernet0/7. Postup je následující:

```
L3_QinQ_SW1(config)# interface GigabitEthernet0/7
```

Konfigurace na obou přepínačích je stejná jako v předchozím návodu, kde na přepínač L3_QinQ_SW1 již není potřeba nastavení IP adresy.

```
L3_QinQ_SW1(config-if)# switchport trunk encapsulation
dot1q
L3_QinQ_SW1(config-if)# switchport mode trunk
L3_QinQ_SW1(config-if)# switchport trunk allowed vlan 600
L3_QinQ_SW1(config-if)# no shutdown
L3_QinQ_SW1(config-if)# exit
```

Tímto máme vyřešenou konfiguraci na straně L3_QinQ_SW1. Adresu na L3_QinQ_SW1 již máme nastavenou, proto není nutné ji nastavovat. Dalším krokem po přihlášení se do L3_QinQ_SW2 z PC20 konfigurace L3_QinQ_SW2.

```
L3_QinQ_SW2(config)# interface GigabitEthernet0/7
L3_QinQ_SW2(config-if)# switchport mode encapsulation dot1q
L3_QinQ_SW2(config-if)# switchport mode trunk
L3_QinQ_SW2(config-if)# switchport trunk allowed vlan 600
L3_QinQ_SW2(config-if)# no shutdown
L3_QinQ_SW2(config-if)# exit
```

Nyní máme nastavený trunk mezi přepínači. Je však nutné nastavit IP adresu na tento přepínač. Nastavení je stejné jako v případě nastavení IP adresy na přepínač L3_QinQ_SW1 s tím rozdílem, že zde bude jiná adresa.

```
L3_QinQ_SW2(config)# interface vlan 600
L3_QinQ_SW2(config-if)# ip address 192.168.99.2
255.255.255.0
L3_QinQ_SW2(config-if)# exit
```

Nyní byste měli být schopni pingat adresu 192.169.99.2 z PC20 a PC19 (obdobně jako u přepínače L3_QinQ_SW1 počkejte cca 20 sekund). Nyní máte mezi přepínači nastavený trunk a měli byste být schopni se na oba přepínače nalogovat i z PC19.

Nastavení portu pro monitorování provozu

Nyní bude probíhat konfigurace monitoring portu pro **PC20**, na kterém pomocí **Wiresharku** bude probíhat měření. Po tomto kroku nebude z **PC20** možný přístup na přepínače z adres 192.168.100.X (možný bude pouze po zrušení této konfigurace). Na přepínači **L3_QinQ_SW1** nastavíme port **GigabitEthernet0/23** jako monitoring portu **GigabitEthernet0/7**. To znamená, že veškerá komunikace, která projde přes port **GigabitEthernet0/7** se bude zrcadlit na port **GigabitEthernet0/23.** Postup je následující:

```
L3_QinQ_SW1(config)# monitor session 1 source interface
Gi0/7
```

Tímto příkazem říkáme, že chceme do monitor session 1 (monitor session lze nastavit 1 až 66) jako zdroj informací rozhraní GigabitEthernet0/7. Na tomto rozhraní GigabitEthernet0/7 je nastavený trunk mezi přepínači.

```
L3_QinQ_SW1(config)# monitor session 1 destination
interface Gi0/23 encapsulation replicate
```

Zde nastavíme port GigabitEthernet0/23 jako cílový port, kam se bude veškerá komunikace z portu GigabitEthernet0/7 kopírovat. Příkazem encapsulation replicate říkáme, že chce zachovat tagování. Cisco standardně nemonitoruje VTP a odstraňuje tak tagování. Pomocí tohoto příkazu veškeré tagování zanecháme. Tento příkaz vás také odřízne od L3_QinQ_SW1, to ale nevadí, protože se na oba přepínače nyní dostanete z obou PC přes nově nastavené adresy.

Nyní máme hotovou veškerou konfiguraci potřebnou pro měření pomocí Wiresharku na PC20.

Měření trunku pomocí programu Wireshark

Nyní změřte pomocí **Wiresharku** na **PC20 (měření provádějte na rozhraní eth0 v Linuxu)** jestli tagování funguje tak jak má. Z **PC19** si pingejte na přepínač **L3_QinQ_SW1** nebo **L3_QinQ_SW2**. Ve Wiresharku byste měli vidět probíhající komunikaci pomocí protokolu **ICMP,** kde budou adresy **PC19** a **L3_QinQ_SW1/ L3_QinQ_SW2**. Ve Wiresharku hledejte označení 802.1Q kde jsou informace o VLAN. **Výsledek měření a danou VLANu (v našem případě 600) ukažte vyučujícímu.**

Úkol 2

Nyní budeme nastavovat **QinQ [1]**. Jedná se zabalení jedné VLAN do druhé. Na přepínačích Cisco 3560 se QinQ realizuje pomocí **`dot1q-tunnel`** a portu nastaveného na **`access`**. Funguje to tím způsobem, že pokud na port nastavený jako QinQ přijde otagovaný rámec (v našem případě VLAN 600) lze tuto VLAN zabalit do další VLAN (Service provider tag) a tím zvětšit možnost použití VLAN (do jedné VLAN můžeme zabalit až 4096 zákaznických VLAN). Dohromady se tedy použitelnost VLAN zvyšuje na 4096 x 4096 = 16 777 216 VLAN.

Přepínače **L3_QinQ_SW2 a L3_QinQ_SW1** jsou propojeny porty Gi0/12 (porty připravené pro QinQ) a zde budeme nastavovat trunk pro Service Provider VLAN. Přepínač **L3_QinQ_SW1** je na portu Gi0/11 propojen s PC20 ze kterého trunk s VLAN 600. Přepínač **L3_QinQ_SW2** je s PC19 propojen pomocí portu Gi0/11 na který jde již PC jako otagovaný VLAN 600. Oba porty, do kterých jsou připojeny PC je potřeba nastavit jako PE (Provider Edge). Celé zapojení je zobrazené na obrázku 4.1. Konfigurace je přepínače **L3_QinQ_SW1** je následující.

```
L3_QinQ_SW1(config)# interface GigabitEthernet0/11
L3_QinQ_SW1(config-if)# switchport access vlan 300
```

Tímto říkáme, že veškerá komunikace, které bude mít označení VLAN300 bude odeslána na tento port a označení VLAN 300 bude odstraněno. Obdobně pokud přijde jakákoliv komunikace skrze tento port, bude označena VLAN 300. Pokud tedy přijde skrz tento port již označená komunikace s VLAN 600 (trunkem), bude tato VLAN 600 zabalena do další VLAN 300 a dál se již bude šířit jen jako VLAN 300, dokud nedojde na port, který bude nastavený jako `switchport access vlan 300.` Pokud tedy na tento port Gi0/11 příjde QinQ rámec s VLAN 300 na výstupu toho portu se označení VLAN 300 odstraní a dále už prochází jen VLAN 600. Proto je na druhé straně nastaven trunk s VLAN 600.

```
L3_QinQ_SW1(config-if)# switchport mode dot1q-tunnel
```

Příkazem switchport mode dot1q-tunnel zajistíme, že port bude nastaven jako QinQ.

```
L3_QinQ_SW1(config-if)# no shutdown
L3_QinQ_SW1(config-if)# exit
```

Konfigurace QinQ na přepínači L3_QinQ_SW1 je hotová a nyní je potřeba přidanou VLAN (v našem případě VLAN 300) šířit dál. Na to je připravený port Gi0/12 a je nutné ho již podle známé konfigurace nastavit jako trunk pro VLAN 300.

```
L3_QinQ_SW1(config)# interface GigabitEthernet0/12
L3_QinQ_SW1(config-if)# switchport trunk encapsulation
dot1q
```

```
L3_QinQ_SW1(config-if)# switchport trunk allowed vlan 300
L3_QinQ_SW1(config-if)# switchport mode trunk
L3_QinQ_SW1(config-if)# no shutdown
L3_QinQ_SW1(config-if)# exit
```

Nastavení QinQ na přepínači L3_QinQ_SW2

Obdobně nastavíme přepínač L3_QinQ_SW2. Zapojení je stejné jako u přepínače L3_QinQ_SW1. Postup je následující:

```
L3_QinQ_SW2(config)# interface GigabitEthernet0/11
L3_QinQ_SW2(config-if)# switchport access vlan 300
L3_QinQ_SW2(config-if)# switchport mode dot1q-tunnel
L3_QinQ_SW2(config-if)# no shutdown
L3_QinQ_SW2(config-if)# exit
```

Zde jsme nastavili QinQ s VLAN 300 stejně jako při konfiguraci přepínače L3_QinQ_SW1.

```
L3_QinQ_SW2(config)# interface GigabitEthernet0/12
L3_QinQ_SW2(config-if)# switchport trunk encapsulation
dot1q
L3_QinQ_SW2(config-if)# switchport trunk allowed vlan 300
L3_QinQ_SW2(config-if)# switchport mode trunk
L3_QinQ_SW2(config-if)# no shutdown
L3_QinQ_SW2(config-if)# exit
```

Momentálně máme pro komunikaci mezi PC19 a PC20 hotovou. Ověříme to pingem na adresy obou PC, kde **PC19 -> 192.168.98.19 a PC20 -> 192.168.98.20**. Pokud mezi sebou počítače nezačnou komunikovat do 20 sekund, je pravděpodobně chyba v nastavení a je nutné celou konfiguraci projít znovu.

Nastavení monitoring portu pro sledování QinQ

V posledním kroku budeme nastavovat opět port monitoring. Tentokrát však budeme sledovat pomocí programu Wireshark QinQ. Zabalení QinQ se nám projeví, až na portu Gi0/12 kde je nastaven trunk s VLAN 300. Nastavení probíhá na přepínači L3_QinQ_SW1. Postup je následující:

```
L3_QinQ_SW1(config)# monitor session 1 source interface
Gi0/12
L3_QinQ_SW1(config)# monitor session 1 destination
interface Gi0/23 encapsulation replicate
```

Nastavení je téměř stejné jako v případě předchozího nastavení s tím rozdílem, že port, který se monitoruje je Gi0/12.

Měření pomocí programu Wireshark

Spusťte program Wireshark na PC20 (pokud jste již tak neprovedli) a z PC19 začněte pingat adresy PC20 nebo z PC20 pingejte adresu PC19 (adresy jsou uvedené v návodu výše). Zachyťte v programu Wireshark procházející komunikaci a najděte v něm komunikaci pomocí QinQ. **Výsledek ukažte vyučujícímu.**

Konfigurace SSH

```
L3_QinQ_SW1(config)# aaa new-model
```

Pro možnost nastavení SSH na přepínači je nutné zadat příkaz `aaa new-model`, který povolí autentizaci, autorizaci a přihlášení pod uživatelem. Bez toho příkazu SSH nastavit nelze.

```
L3_QinQ_SW1(config)# username XXXX secret XXXX
```

Vytvoříme uživatele de libosti. Za XXXX doplníme jak jméno uživatele a také heslo.

```
L3_QinQ_SW1(config)# ip domain-name XXXX.XXXX
```

Vytvořte libovolný doménový název, například `ip domain-name bars.vut`

```
L3_QinQ_SW1(config)# crypto key generate rsa
How many bits in the modulus [512]: 768
```

Pro používání SSH je nutné na přepínači vytvořit RSA šifrovací klíč. Pokud by již byl na přepínačích RSA klíč vytvořen, dostanete varování, zda chce aktivní klíč přepsat klíčem novým, který vytvoříte. V takovém případě napište yes, ale v úloze by klíč neměl být vytvořený. Číslo 768 se zadává z toho důvodu, že je nutné použít minimálně 768 bitový klíč pro použití SSH verze 2.

```
L3_QinQ_SW1(config)# ip ssh version 2
```

Nastavíme verzi SSH na verzi 2.

```
L3_QinQ_SW1(config)# ip ssh autentication-retries 2
```

Tímto příkazem nastavujeme počet pokusů pro přihlášení. V našem případě je počet nastavený na 2 pokusy, po těchto dvou pokusech se přihlášení v dane relaci zablokuje a nutné opětovné připojení přes putty.

```
L3_QinQ_SW1(config)# ip ssh time-out 60
```

Pomocí time-out nastavujeme dobu vypršení při přihlašování na přepínač. Po 60 sekundách a neaktivitě při přihlašování se spojení s přepínačem uzavře.

```
L3_QinQ_SW1(config)# monitor session 1 source interface
Gi0/3
```

Přes tento port se přihlašujete na oba přepínače, proto je nutné nastavit tento port do monitorování. Bez toho příkazu nebudete vidět komunikaci SSH ani Telnet v programu Wireshark.

Pokud jste úspěšně nastavili přihlášení pomocí SSH můžete začít měření. Na PC20 si spusťte program Wireshark. Měření si nastavte na port, který přistupuje na přepínač pomocí VLAN 600 (na PC20 je to port eth1). Připravte si na PC19 přihlášení pomocí Terminálu Linuxu a telnetem na adresu L3_QinQ_SW1. Spusťte měření a začněte psát přihlašovací údaje. Po přihlášení na přepínač vypněte měření a vložte do Wiresharku filtr pro telnet. Měla by se vám vyfiltrovat veškerá komunikace pomocí telnet a také byste zde měli být schopni vyhledat vše, co jste psali při přihlášení (uživatele u heslo).

Dalším krokem je přihlášení pomocí SSH na přepínač. Postupujte stejně jako v předešlém případě. Na konec porovnejte oba způsoby přihlašování a výsledek ukažte vyučujícímu.

Po ukončení všech bodů v úloze restartujte přepínače pomocí příkazu reload a nastavení neukládejte! Jako první vždy rebootněte přepínač L3_QinQ_SW2, jinak budete muset čekat, než přepínači L3_QinQ_SW1 najede systém a až poté se na druhý přepínač dostanete (ale už jen přes PC20).

```
L3_QinQ_SW2# reload
System configuration has been modified. Save? [yes/no]: no
```

Otázky

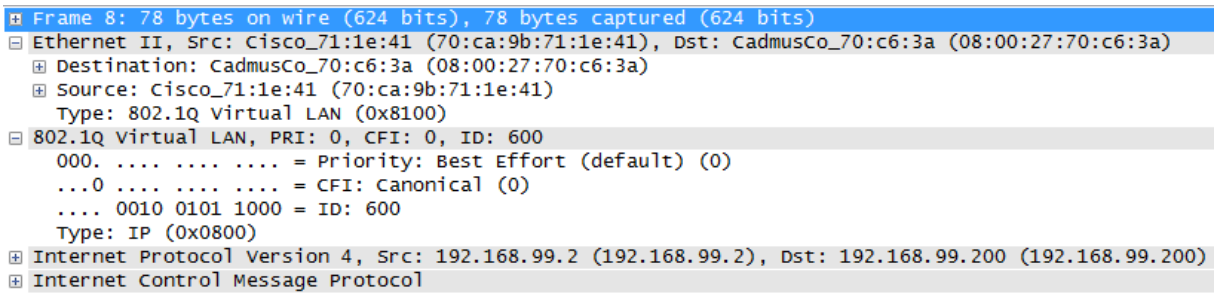
1) **K čemu slouží virtualizace sítí pomocí sítí VLAN?**
2) **Jaký je rozdíl v používání sítí VLAN a QinQ?**
3) **Jaký je rozdíl při přihlášení se na přepínač pomocí telnet a SSH?**

Literatura

[1] Catalyst 3560 Software Configuration Guide, Release 12.2(52)SE: Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling. In: *Www.cisco.com* [online]. [cit. 2016-05-10]. Dostupné z: http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst3560/software/release/12-2_52_se/configuration/guide/3560scg/swtunnel.html

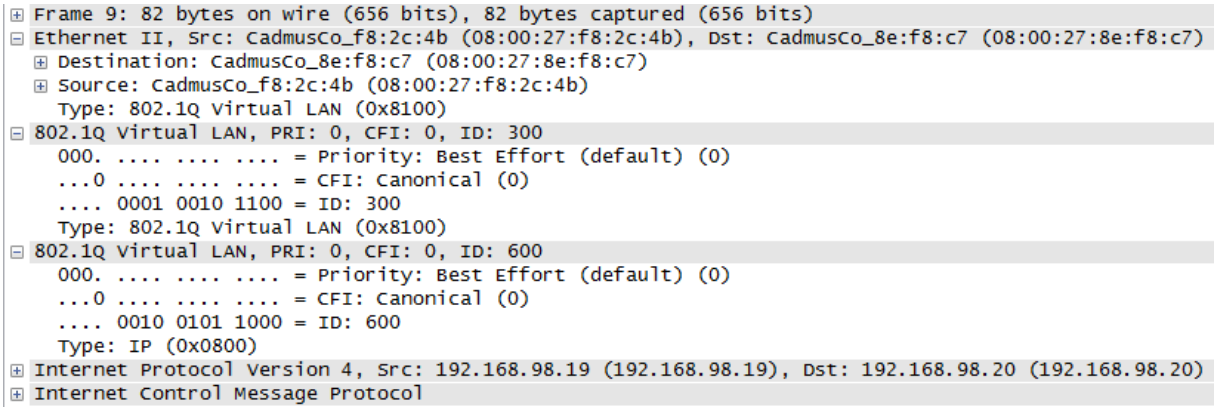
Příloha 2

Úkol1 – výsledek měření



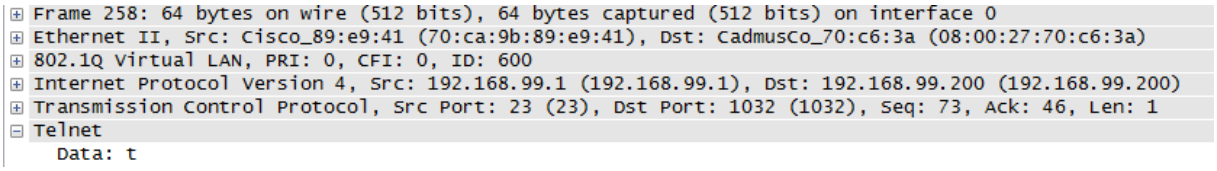
Pomocí VLAN 600 probíhá od PC, které jsou použité pro laboratorní úlohu. Na obrázku je vidět, že používáme trunk, protože je zde EtherType 0x8100 typický pro komunikaci pomocí trunku. EtherType 0x0800 značí začátek hlavičky IPv4.

Úkol 2 – výsledek měření



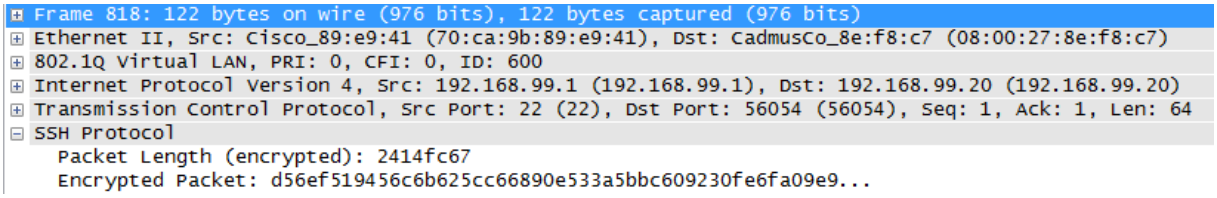
Na obrázku je zobrazen výsledek měření QinQ. Jsou zde vidět jednotlivá čísla VLAN a také jim souhlasící EtherType. Lze tedy z toho obrázku vyvodit to, že VLAN 300 má EtherType 0x8100 a je to tedy VLAN při přenosu QinQ (VLAN 300 tedy zabaluje VLAN 600). VLAN 600 má EtherType 0x8100 a je určena pro přenos pomocí trunku nebo na porty, které jsou touto VLAN označeny.

Úkol 3 – výsledek měření



Na obrázku je zřetelně vidět, že uživatel napsal písmeno „t“, komunikace probíhá pomocí telnet protokolu a komunikace zde není šifrovaná. Pokud bychom prošli celou komunikaci od přihlášení se po zadání heslo uživatele, je při použití telnetu veškerá komunikace viditelná. Pokud po vás stanice, na kterou se přihlašujete, požaduje jméno uživatele, zobrazí se vám v příkazové řádce „User“ nebo „Username“. To samé se také zobrazí při zachytávání komunikace telnet v programu Wireshark. Pokud si tedy projdete

komunikaci v programu Wireshark od začátku komunikace, není těžké v ní veškeré přihlašovací údaje dohledat.



Zde je zobrazena komunikace pomocí SSH. Na obrázku je vidět, že komunikace je šifrovaná, tudíž z ní nelze nic vyčíst.

Otázky

1) **K čemu slouží virtualizace sítí pomocí sítí VLAN?**
   VLAN je technologie, která umožňuje oddělení fyzického spojení od logického (nezávisle na fyzickém uspořádání). Uživatelé v této síti jsou stále fyzicky propojeni, jako to je to u sítí LAN s tím rozdílem, že je nutné, aby mezi nimi byl směrovač pro komunikaci mezi sebou.
2) **Jaký je rozdíl v používání sítí VLAN a QinQ?**
   Celkový počet sítí VLAN se v klasickém pojetí pohybuje od 1 do 4096. QinQ dovoluje zvýšit počet použitých sítí VLAN na 4096 x 4096 = 16 777 216 VLAN. QinQ oproti použití sítí VLAN používá dvojité tagování. Jako nevýhoda přepínačů používajících dvojité tagování (QinQ) je taková, že musí pojmout velké množství MAC adres do svých CAM tabulek.
3) **Jaký je rozdíl při přihlášení se na přepínač pomocí telnet a SSH?**
   Jak je vidět v laboratorní úloze, přihlášení pomocí protokolu telnet není bezpečné. Veškerá komunikace je viditelná, tudíž pro útočníka není složité zjistit heslo a uživatelské jméno. SSH na rozdíl od protokolu telnet šifruje veškerou komunikaci, tudíž z ni nelze nic vysledovat.