

Počítačové sítě 2

1, Základní konfigurace zařízení

Název bloku	Náplň
Počáteční nastavení přepínače	Konfigurace základního nastavení přepínače CISCO.
Konfigurace portů přepínače	Konfigurace portů přepínače dle požadavků sítě.
Zabezpečený vzdálený přístup	Konfigurace zabezpečeného přístupu na přepínač.
Základní konfigurace směrovače	Konfigurace základního nastavení směrovače pro směrování mezi dvěma přímo propojenými sítěmi s použitím CLI.
Verifikace přímo propojených sítí	Verifikace propojení dvou sítí, které jsou přímo připojené na směrovač.

1.1 Konfigurace počátečního nastavení přepínače

V prvním bloku se budeme věnovat přepínači, přičemž začneme jeho počátečním nastavením.

Jde o situaci, kdy bud' vybalíte z krabice právě dodaný přepínač a nebo jste provedli jeho reset, neboli uvedení do továrního nastavení.

Po prvním zapnutí projde přepínač Cisco následující pětistupňovou spouštěcí sekvencí:

1: Nejprve přepínač načte program pro autotest po zapnutí (tzv POST). Ten je uložený v paměti ROM. POST kontroluje subsystém procesoru, při kterém testuje CPU, DRAM a část flash zařízení, která tvoří systém souborů flash.

Ve druhém kroku přepínač načte software zavaděče. Zavaděč je malý program uložený v ROM, který se spustí ihned po úspěšném dokončení POST.

3 Krok. Zavaděč provádí inicializaci procesoru na nízké úrovni. Inicializuje jeho registry, které sledují, kde je mapována fyzická paměť, množství paměti a její rychlosť.

Ve 4 kroku zavaděč inicializuje systém souborů flash na základní desce.

V posledním pátém kroku zavaděč vyhledá a načte výchozí snímek softwaru operačního systému IOS do paměti a dá kontrolu nad přepnutím na IOS.

Přepínač se pokusí automaticky spustit pomocí informací v proměnné prostředí BOOT. Pokud tato proměnná není nastavena, přepínač se pokusí načíst a spustit první spustitelný soubor, který najde.

```
S1(config)# boot system flash:/c2960-lanbasek9-mz.150-2.SE/c2960-lanbasek9-mz.150-2.SE.bin
```

Operační systém IOS poté inicializuje rozhraní pomocí příkazů Cisco IOS nalezených v souboru startup-config. Spouštěcí konfigurační soubor se nazývá config.text a je umístěn ve flash paměti.

V uvedeném příkladu je proměnná prostředí BOOT nastavena pomocí příkazu režimu globální konfigurace spouštěcího systému. Všimněte si, že IOS je umístěn v odlišné složce a je zadána cesta k této složce. Pomocí příkazu show boot můžete zjistit, na co je aktuální zaváděcí soubor IOS nastaven.

Příkaz	Definice
boot system	Hlavní příkaz
flash:	Paměťové zařízení
c2960-lanbasek9-mz.150-2.SE/	Cesta k souborovému systému
c2960-lanbasek9-mz.150-2.SE.bin	Název IOS souboru

LED indikátory přepínače

Zda je zařízení funkční a v jakém stavu lze zjistit nejen z výpisů z příkazového řádku.

Prvotní informaci lze získat při pohledu na tělo přepínače, na indikační LED diody.

Systémová LED (SYST) indikuje, zda je systém napájen a správně funguje.

LED redundantního napájecího zdroje (RPS) zobrazuje jeho stav.

Pokud **LED dioda stavu portu (STAT)** svítí zeleně, znamená to, že je vybrán režim stavu portu, což je defaultní nastavení.

LED duplexního režimu portů (DUPLX): Pokud svítí zeleně, znamená to, že je vybrán duplexní režim.

LED rychlosti portu (RYCHLOST): Pokud svítí zeleně, znamená to, že je vybrán režim rychlosti portu.

Power over Ethernet LED (PoE): Je k dispozici, pokud přepínač podporuje PoE.

Tlačítko Mode se používá k přepínání mezi různými mody - STAT, DUPLX, SPEED, a PoE

Konkrétní stav daného portu ve zvoleném režimu lze pak určit podle světla spojeného s každým portem. Jednotlivé varianty jsou uvedeny na následujícím snímku.

Svit LED mohou indikovat různé stavy, i proto je zde přepínač MODE, aby mohla být postihnuta většina problémů. Když jsou popsány v následující tabulce.

	Vypnutý	Zelená	Blikající zelená	Žlutá	Blikající žlutá	Střídavá zelená/žlutá
RPS	Vypnuto/ Není RPS	RPS připravená	RPS zapnutá, ale nedostupná	RPS v pohotovostní poloze, nebo porucha	Interní PS selhalo, RPS zajišťuje napájení	N/A
PoE	Není vybráno, bez problémů	Vybráno	N/A	N/A	Není vybráno, jsou problémy s portem	N/A
Když je vybrán pojmenovaný režim, světlo spojené s každým fyzickým portem označuje:						
STAT	Žádné propojení nebo vypnutí	Propojeno	Aktivita	Zablokovaný port zabraňující smyčce	Zablokovaný port zabraňující smyčce	Chyba spojení
DUPLEX	Half-duplex	Full-duplex	N/A	N/A	N/A	N/A
SPEED	10Mbps	100Mbps	1000Mbps	N/A	N/A	N/A
PoE	PoE vypnute	PoE zapnuté	N/A	PoE zakázáno	PoE vypnuto kvůli poruše	PoE zamítnuto (přes rozpočet)

Obnova po havárii systému

Havárie se nevyhýbají ani aktivním prvkům, které jsou napájeny přes záložní zdroje a to i přes to že volitelně mohou obsahovat redundantní napájecí zdroj. V takovém případě se provádí obnova, je-li to samozřejmě technicky možné.

V takovém případě, pokud operační systém nelze použít kvůli chybějícím nebo poškozeným systémovým souborům, zavaděč poskytuje přístup do přepínače. Obsahuje příkazový řádek, který poskytuje přístup k souborům uloženým v paměti flash. K zavaděči lze přistupovat prostřednictvím připojení konzoly pomocí následujících kroků:

Připojte počítač kabelem konzoly k portu konzoly přepínače. Nakonfigurujete software emulace terminálu (Putty) pro připojení k přepínači.

Odpojte napájecí kabel přepínače.

Znovu připojte napájecí kabel k přepínači a do 15 sekund stiskněte a podržte tlačítko Mode, po dobu co kontrolka systému stále bliká zeleně a do chvíle,

dokud nezačne krátce oranžově a poté trvale zeleně svítit. V té chvíli tlačítko Mode uvolněte.

V posledním kroku v software pro emulaci terminálu se v počítači zobrazí výzva přepínače zavaděče.

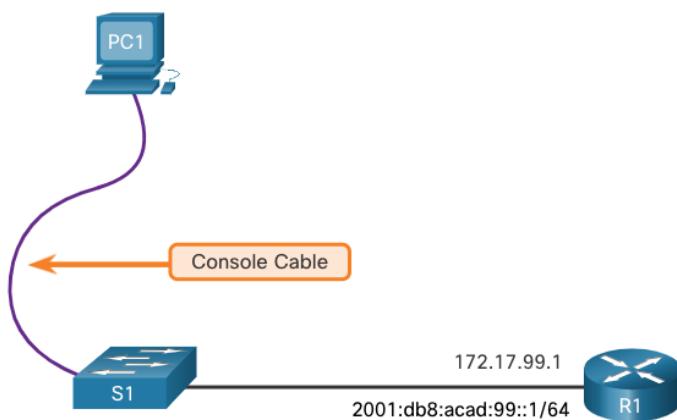
Příkazový řádek zavaděče podporuje příkazy pro formátování systému souborů flash, přeinstalování softwaru operačního systému a obnovení ztraceného nebo zapomenutého hesla. Například příkaz **dir** lze použít k zobrazení seznamu souborů v zadaném adresáři.

Z uvedeného je důležité, že touto cestou lze překlenout problém ztráty hesla.

Přístup ke správě přepínače

Chcete-li připravit přepínač pro přístup ke vzdálené správě, musí být přepínač nakonfigurován s IP adresou a maskou podsítě a výchozí bránou. Je to velmi podobné konfiguraci informací o IP adrese na koncových zařízeních.

Na obrázku by virtuálnímu rozhraní přepínače (SVI – switch virtual interface) na S1 měla být přiřazena IP adresa.



SVI je virtuální rozhraní, nikoli fyzický port na přepínači. Kabel konzoly se používá k připojení k PC, aby bylo možné přepínač inicializovat.

Příklad SVI konfigurace na přepínači

Ve výchozím nastavení je přepínač nakonfigurován tak, aby byl spravován prostřednictvím VLAN 1. Všechny porty jsou ve výchozím nastavení přiřazeny k VLAN 1. Z bezpečnostních důvodů se pro správu VLAN považuje za nejlepší postup použít VLAN s jiným ID než VLAN 1.

Provést to můžeme pomocí následujících kroků a to v režimu konfigurace rozhraní VLAN

V prvním kroku se na správu SVI přepínače použije adresa IPv4 a maska podsítě.

Nutno podotknout, že SVI pro VLAN 99 se nezobrazí jako „up / up“, jednak

dokud není vytvořena VLAN 99,

A také dokud neexistuje zařízení připojené k portu přepínače, k němuž je přiřazena VLAN 99.

Pokud je nutné nakonfigurovat přepínač pro protokol IPv6 je třeba mít přehled o verzi IOSu. Například dříve, než budete moci konfigurovat adresování IPv6 na Cisco Catalyst 2960 se systémem IOS verze 15.0, budete muset zadat globální konfigurační příkaz *sdm prefer dual-ipv4-and-ipv6 default* a poté přepínač restartovat.

V tabulce je uvedena sada příkazů příkazové řádky, kterou je třeba pro konfiguraci vykonat.

Úkol	IOS příkazy
Vstup do globálního konfiguračního módu.	S1# configure terminal
Vstup do konfiguračního módu rozhraní pro SVI.	S1(config)# interface vlan 99
Konfigurace IPv4 adresy rozhraní pro správu.	S1(config-if)# ip address 172.17.99.11 255.255.255.0
Konfigurace IPv6 adresy rozhraní pro správu.	S1(config-if)# ipv6 address 2001:db8:acad:99::1/64
Povolte rozhraní pro správu.	S1(config-if)# no shutdown
Návrat do privilegovaného EXEC módu.	S1(config-if)# end
Uložení běžící konfigurace do spouštěcí konfigurace.	S1# copy running-config startup-config

Ve druhém kroku Nakonfigurujte default gateway.

Přepínač by měl být nakonfigurován s default gateway, pokud bude spravován vzdáleně ze sítí, které nejsou přímo připojeny.

Pokud tomu tak není, default gateway být nakonfigurována být nemusí, čímž se eliminuje možnost části útoků, které jsou vedené z externí sítě.

Task	IOS Commands
Vstup do globálního konfiguračního módu.	S1# configure terminal
Konfigurace default gateway přepínače.	S1(config)# ip default-gateway 172.17.99.1
Návrat do privilegovaného EXEC módu.	S1(config-if)# end
Uložení běžící konfigurace do spouštěcí konfigurace.	S1# copy running-config startup-config

Ještě poznámka k IPv6. Přepínač nevyžaduje zadání IPv6 default gateway, neboť ji přijme z RA zprávy.

V posledním, třetím, kroku je třeba ověřit provedená nastavení. K tomu slouží příkazy **show ip interface brief** a **show ipv6 interface brief**.

Dají informaci o stavu fyzického i virtuálního rozhraní. Zobrazený výstup potvrzuje, že rozhraní VLAN 99 bylo nakonfigurováno s adresami IPv4 a IPv6.

Poznámka: IP adresa použitá pro SVI je pouze pro přístup vzdálené správy k přepínači, ale neumožňuje přepínači směrovat pakety třetí vrstvy.

1.2 Konfigurace portů přepínače

Full-duplexní komunikace zvyšuje efektivitu šířky pásmo tím, že umožňuje oběma koncům připojení současně vysílat a přijímat data. Toto je také známé jako obousměrná komunikace, která vyžaduje mikrosegmentaci.

Mikrosegmentovaná LAN se vytvoří, když je k portu přepínače připojeno pouze jedno zařízení, které pracuje v plně duplexním režimu. S portem přepínače, pracujícím v plně duplexním režimu, není svázana žádná kolizní doména.

Na rozdíl od plně duplexní komunikace je half-duplex komunikace jednosměrná. Vytváří problémy s výkonem, protože data mohou proudit v daném čase pouze jedním směrem, což často vede ke kolizím. Často toto bylo nastaveno na tiskárnách, která primárně přijímají datový tok tištěných dokumentů, a zpět ke zdroji jde jen potvrzení.

Síťové adaptéry Gigabit Ethernetu a 10 Gb vyžadují pro provoz výhradně plně duplexní připojení. V tomto režimu je detekce kolize na rozhraní deaktivována.

Full-duplex nabízí stoprocentní účinnost v obou směrech, myšleno vysílání i příjem. To má za následek zdvojnásobení potenciálního využití uvedené šířky pásma.

Při konfiguraci aktivního prvku je možné porty konfigurovat ručně na konkrétní nastavení duplexu/half-duplexu a rychlosti. Příslušné konfigurační příkazy rozhraní jsou duplex a speed.

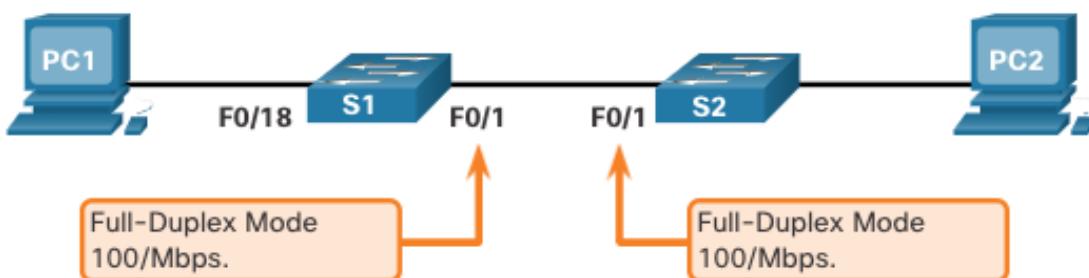
Výchozí nastavení těchto parametrů portů je u přepínačů Cisco Catalyst 2960 a 3560 automatické.

Porty 10/100/1000 fungují v režimu polovičního nebo plně duplexního režimu, pokud jsou nastaveny na 10 nebo 100 Mb / s, a fungují pouze v plně duplexním režimu, pokud je port nastaven na 1000 Mbps (1 Gbps).

Při připojování ke známým zařízením, jako jsou servery, vyhrazené pracovní stanice nebo síťová zařízení, je nejlepší rychlosť a duplexní mód nastavit ručně. Občas dochází k problémům s vendor kompatibilitou a zařízení se nemohou dohodnout na automatickém nastavení.

Při řešení problémů s portem přepínače je důležité zkontrolovat nastavení duplexu a rychlosti, případně ho nastavit ručně.

Všechny optické porty, například porty 1000BASE-SX, fungují pouze s jednou přednastavenou rychlosťí a jsou vždy plně duplexní.



S fyzickou konfigurací na obrázku souvisí příkazy příkazového řádku v tabulce níže:

Úloha	iOS příkazy
Vstup do globálního konfiguračního módu.	S1# configure terminal
Vstup do konfiguračního módu pro rozhraní.	S1(config)# interface FastEthernet 0/1
Konfigurace duplexního rozhraní.	S1(config-if)# duplex full
Konfigurace rychlosťi rozhraní.	S1(config-if)# speed 100
Návrat do privilegovaného EXEC módu.	S1(config-if)# end
Uložení běžící konfigurace do spouštěcí konfigurace.	S1# copy running-config startup-config

Když hovoříme o automatickém režimu, myslí se tím také Auto-MDIX, neboť používáte-li auto-MDIX na rozhraních, jejich rychlosť a duplex musí být nastaveny také na **auto**, aby funkce fungovala správně.

Když je povoleno automatické křížení rozhraní závislého na médiu (auto-MDIX), rozhraní přepínače automaticky detekuje požadovaný typ propojovacího kabelu (přímý nebo křížený) a odpovídajícím způsobem nakonfiguruje připojení.

Při připojování k přepínačům bez funkce auto-MDIX je nutné použít přímé kably pro připojení k zařízením, jako jsou servery, pracovní stanice nebo směrovače.

K propojení s jinými přepínači nebo opakovači je nutné použít křížené kably.

Je-li povolen automatický MDIX, lze k připojení k jiným zařízením použít jakýkoli typ kabelu a rozhraní se automaticky přizpůsobí pro fungující komunikaci.

U novějších přepínačů Cisco tuto funkci povoluje příkaz konfiguračního módu **mdix auto** na daném rozhraní.

Je třeba poznamenat, že funkce auto-MDIX je ve výchozím nastavení povolena u přepínačů Catalyst 2960 a Catalyst 3560, ale není k dispozici u starších přepínačů Catalyst 2950 a Catalyst 3550.

K prozkoumání nastavení auto-MDIX pro konkrétní rozhraní nám slouží příkaz **show controllers ethernet-controller** s klíčovým slovem **phy**. Chcete-li omezit výstup na řádky odkazující na auto-MDIX, použijte **include Auto-MDIX filter**.

Příkazy verifikace nastavení přepínače

Tabulka obsahuje příkazy pro ověření konfigurace přepínače. Můžeme je použít pro zobrazení uvedených nastavení a stavů.

Úloha	IOS příkazy
Zobrazit stav a konfiguraci rozhraní.	S1# show interfaces [interface-id]
Zobrazit aktuální spouštěcí konfiguraci.	S1# show startup-config
Zobrazit aktuální běžící konfiguraci.	S1# show running-config
Zobrazit informace o souborovém systému v paměti flash.	S1# show flash
Zobrazit stav hardwaru a softwaru systému.	S1# show version
Zobrazit historii zadaných příkazů.	S1# show history
Zobrazit IP informace o rozhraní.	S1# show ip interface [interface-id] OR S1# show ipv6 interface [interface-id]
Zobrazit tabulku MAC adres.	S1# show mac-address-table OR S1# show mac address-table

Ověření konfigurace portu přepínače

Také příkaz **show running-config** lze použít k ověření, že přepínač byl správně nakonfigurován.

Na obrázku je ze zkráceného výstupu na S1 zřejmě některé důležité informace:

- Rozhraní Fast Ethernet 0/18 je konfigurováno se správou VLAN 99
- VLAN 99 je konfigurována s IPv4 adresou 172.17.99.11 255.255.255.0
- Default gateway má adresu 172.17.99.1

```
S1# show running-config
Building configuration...
Current configuration : 1466 bytes
!
(output omitted)
interface Vlan99
  ip address 172.17.99.11 255.255.255.0
  ipv6 address 2001:DB8:ACAD:99::1/64
!
ip default-gateway 172.17.99.1
```

Dále lze k ověření použít příkaz **show running-config**, že přepínač byl správně nakonfigurován.

Na snímku je ze zkráceného výstupu na S1 zřejmě některé důležité informace:

- Rozhraní Fast Ethernet 0/18 je konfigurováno se správou VLAN 99
- VLAN 99 je konfigurována s IPv4 adresou 172.17.99.11 255.255.255.0
- Default gateway má adresu 172.17.99.1

```
S1# show running-config
Building configuration...
Current configuration : 1466 bytes
!
(output omitted)
interface Vlan99
  ip address 172.17.99.11 255.255.255.0
  ipv6 address 2001:DB8:ACAD:99::1/64
!
ip default-gateway 172.17.99.1
```

Příkaz **show interfaces** je další běžně používaný příkaz, který zobrazuje stavové a statistické informace o síťových rozhraních přepínače.

Používá se často při konfiguraci a monitorování síťových zařízení.

```
S1# show interfaces fastEthernet 0/18
FastEthernet0/18 is up, line protocol is up (connected)
  Hardware is Fast Ethernet, address is 0025.83e6.9092 (bia 0025.83e6.9092)
  MTU 1500 bytes, BW 100000 Kbit/sec, DLY 100 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, media type is 10/100BaseTX
```

V ukázce na obrázku první řádek výstupu příkaz **show interfaces fastEthernet 0/18** označuje, že rozhraní FastEthernet 0/18 je up / up. Dále se dočteme, že je použitý full-duplex a rychlosť je nastavena na 100 Mbps.

Problémy se síťovou vrstvou

Interpretaci výstupu příkazu **show interfaces** lze použít pro detekci běžných problémů s médii.

První parametr (FastEthernet0 / 18 je up) odkazuje na hardwarovou vrstvu a označuje, zda rozhraní přijímá signál, je tedy detekovaný nosič. Druhý parametr (linkový protokol je up) odkazuje na datovou vrstvu a označuje, zda jsou přijímány protokoly o protokolu datové vrstvy.

```
S1# show interfaces fastEthernet 0/18
FastEthernet0/18 is up, line protocol is up (connected)
Hardware is Fast Ethernet, address is 0025.83e6.9092 (bia 0025.83e6.9092)MTU 1500 bytes, BW
100000 Kbit/sec, DLY 100 usec,
```

Díky tomu lze následujícím způsobem vyřešit tyto problémy:

Pokud je rozhraní up a linkový protokol je down, nastává problém. Mohlo by dojít k neshodě typu zapouzdření, rozhraní na druhém konci by mohlo být deaktivováno kvůli chybě nebo mohl nastat hardwarový problém.

Pokud jsou jak linkový protokol, tak i rozhraní down, nejspíš není připojený kabel, nebo nastal jiný problém s rozhraním. Například v připojení typu back-to-back může být druhý konec připojení administrativně nefunkční (down). To znamená, že bylo v aktivní konfiguraci zřejmě ručně deaktivováno (byl vydán příkaz vypnutí - down).

Výstup dále zobrazuje čítače a statistiky pro definované rozhraní, které vyjadřují chyby na vstupu a výstupu rozhraní.

Některé chyby médií nejsou dostatečně závažné, aby způsobily selhání obvodu, ale způsobují problémy s výkonem sítě. Tabulka vysvětluje některé z těchto běžných chyb, které lze zjistit pomocí příkazu **show interfaces**.

Typ chyby	Popis
Vstupní chyby	Celkový počet chyb. Zahrnuje runts, giants, no buffer, CRC, frame, overrun, and ignored counts.
Runt (zákrsek)	Pakety, které jsou zahrozeny, protože jsou menší než minimální velikost paketu pro médium. Například jakýkoli ethernetový paket, který má méně než 64 bajtů, je považován za runt.
Gigant (obr)	Pakety, které jsou zahrozeny, protože překračují maximální velikost paketu pro médium. Například jakýkoli ethernetový paket, který je větší než 1518 bajtů, je považován za gigant.
CRC	Chyby CRC se generují, když vypočítaný kontrolní součet není stejný jako přijatý kontrolní součet.
Výstupní chyby	Součet všech chyb, které zabránily konečnému přenosu datagramů z rozhraní, které je zkoumáno.
Kolize	Počet zpráv přenášených znovu kvůli Ethernetovské kolizi.
Pozdní kolize	Kolize, ke které dojde po přenosu 512 bitů rámce.

„Chyby vstupu“ je součet všech chyb v datagramech, které byly přijaty na prověřovaném rozhraní. Zahrnují runts, giants, CRC, no buffer, frame, overrun, and ignored counts. Hlášené chyby vstupu z příkazu **show interfaces** zahrnují následující:

- **Runt rámce** - Ethernetové rámce, které jsou kratší než minimální povolená délka 64 bajtů, se nazývají runts. Nefunkční síťové karty jsou obvyklou příčinou nadměrného výskytu těchto rámců, ale mohou být také způsobeny kolizemi.

- **Giganti** - Ethernetové rámce, které jsou větší než maximální povolená velikost, se nazývají giganti.
- **Chyby CRC** - Na ethernetových a sériových rozhraních chyby CRC obvykle znamenají chybu média nebo kabelu. Mezi běžné příčiny patří elektrické rušení, uvolněné nebo poškozené spoje nebo nesprávná kabeláž. Pokud je mnoho chyb CRC, je na lince vysoký šum a je nutno kabel zkontovalovat. Je vhodné také hledat a eliminovat zdroje rušení.

Dále jsou zde chyby výstupu.

Jde o součet všech chyb, které zabránily konečnému přenosu datagramů z daného rozhraní. Hlášené chyby výstupu z příkazu **show interfaces** zahrnují:

- **Kolize** - Kolize v poloduplexních operacích jsou normální. Nikdy byste však neměli vidět kolize na rozhraní nakonfigurovaném pro plně duplexní komunikaci.
- **Pozdní kolize** - Pozdní kolize označuje kolizi, ke které dojde po přenosu 512 bitů rámce. Příliš dlouhé délky kabelů jsou nejčastější příčinou pozdních kolizí. Další častou příčinou je nesprávná konfigurace duplexu.

1.3 Bezpečný vzdálený přístup

Telnet je letitý běžný prostředek, který používá port TCP 23. Jedná se o starší protokol, který používá nezabezpečený prostý přenos jak přihlašovacího ověřování (uživatelské jméno a heslo), tak dat přenášených mezi komunikujícími zařízeními. Je potřeba toto nebrat na lehkou váhu. Narušitel může pomocí Wiresharku monitorovat pakety a zachytit uživatelské jméno **admin** a heslo **ccna** právě z relace Telnetu.

Mnohem bezpečnější a tedy i preferovaná je komunikace pomocí Secure Shell (SSH), což je zabezpečený protokol, který používá TCP port 22. Poskytuje zabezpečené (šifrované) připojení pro správu vzdáleného zařízení. SSH **by měl** nahradit Telnet pro připojení pro správu. SSH poskytuje zabezpečení pro vzdálená připojení tím, že poskytuje silné šifrování, jednak v okamžiku když je zařízení ověřeno a přenáší se uživatelské jméno a heslo, a také když jsou přenášená data mezi komunikujícími zařízeními.

Při zachycení relace SSH programem Wireshark může útočník sice relaci sledovat pomocí IP adresy, ale na rozdíl od Telnetu je v SSH uživatelské jméno a heslo zašifrováno.

Chcete-li povolit SSH na přepínači Catalyst 2960, musí přepínač používat verzi softwaru IOS, který obsahuje kryptografické (šifrovací) funkce a schopnosti. Pomocí příkazu **show version** na přepínači zjistíte, pod jakou verzí IOS přepínač aktuálně běží. Název systému IOS, který obsahuje kombinaci „k9“, podporuje kryptografické (šifrovací) funkce.

Konfigurace SSH

Konfiguračních kroků je více, nejde však o nic složitého a nepochopitelného. Před konfigurací SSH musí být přepínač minimálně nakonfigurován s **jedinečným názvem hostitele** a **správným nastavením síťového připojení**.

Než se pustíte do konfigurace, ověřte podporu SSH daného zařízení pomocí příkazu **show ip ssh**. Pokud na přepínači není spuštěn IOS, který podporuje kryptografické funkce, tento příkaz není rozpoznán.

V dalším kroku je potřeba provést konfiguraci IP domény. provede se to v globálním konfiguračním módu pomocí příkazu **ip domain-name <název-domény>**.

Generování dvojice klíčů RSA se provede jako třetí krok. Generování páru klíčů RSA automaticky umožňuje SSH. Pomocí příkazu globálního konfiguračního módu **crypto key generate rsa** povolíte na přepínači server SSH a vygenerujete pár RSA klíčů.

(V případě potřeby k jejich odstranění, použijte příkaz **crypto key zeroize rsa** v globálním konfiguračním módu. Po odstranění dvojice klíčů RSA je SSH server automaticky deaktivován)

Čtvrtým krokem je konfigurace ověřování uživatelů. SSH server může ověřovat uživatele místně nebo pomocí ověřovacího serveru. Chcete-li použít metodu místního ověřování, vytvořte dvojici uživatelského jména a hesla příkazem globálního konfiguračního módu **username <username> secret password**.

Krok pátý je konfigurace vty linek. Povolte protokol SSH na vty linkách pomocí příkazu **transport input ssh line** v konfiguračním módu. Použijte příkaz **line vty** v globálním konfiguračním módu a poté příkaz **login local line** v konfiguračním módu na získání lokální autentizace pro SSH připojení z lokální databáze uživatelských jmen.

V posledním šestém kroku povolte SSH ve verzi 2. Ve výchozím nastavení podporuje SSH první i druhou verzi, což se ve výstupu příkazu **show ip ssh** zobrazí jako podpůrná verze 2. Povolte vyšší verzi SSH pomocí globálního konfiguračního příkazu **ip ssh version 2**.

Ověření funkčnosti SSH

Na koncových stanicích se pro připojení k SSH serveru používá klientská aplikace, jako je např. již zmíněná PuTTY, která dovoluje zvolit různé režimy připojení.

Pro realizaci připojení předpokládejme například, že je nakonfigurováno následující:

- na přepínači S1 je povolen SSH
- Rozhraní VLAN 99 (SVI) má IPv4 adresu 172.17.99.11 na přepínači S1
- PC1 má IPv4 adresu 172.17.99.21

Pomocí emulátoru terminálu zahajte SSH připojení k IPv4 adrese SVI VLAN S1 z PC1.

Po připojení je uživatel vyzván k zadání uživatelského jména a hesla. Pomocí konfigurace z předchozího příkladu se zadá uživatelské jméno **admin** a heslo, které se na konzoli při zadávání nevypíše. Po zadání správné kombinace je uživatel připojen přes SSH k rozhraní příkazového řádku (CLI).

Chcete-li zobrazit verzi a konfigurační data pro SSH na zařízení, které jste nakonfigurovali jako server SSH, použijte příkaz **show ip ssh**.

1.4 Základní konfigurace směrovače

Další podkapitola se věnuje směrovačům - routerům, kde je konfigurace dílem stejná, avšak jsou zde pochopitelně rozdíly.

Základní nastavení směrovače

Jak jsem již naznačil, mají Cisco směrovače a přepínače mnoho podobného. Podporují podobný modální operační systém, podobné struktury příkazů a mnoho stejných příkazů. Obě zařízení mají navíc podobné počáteční kroky konfigurace. Například by měly být vždy provedeny následující konfigurační úlohy.

Pojmenujte zařízení, abyste jej odlišili od ostatních směrovačů příkazem **hostname R1**

Příklad ukazuje konfiguraci hesla pro přístup z konzole i z terminálu, přičemž se provede zabezpečení, aby se hesla ukládala zašifrovaná příkazem **service password-encryption**.

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# hostname R1
R1(config)# enable secret class
R1(config)# line console 0
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)# service password-encryption
R1(config)#

```

Dle Cisco doporučení je vhodné nastavit banner, který bude osobu přistupující k zařízení informovat o legálnosti přístupu, viz první ukázka.

```
R1(config)# banner motd $ Authorized Access Only! $
R1(config)#

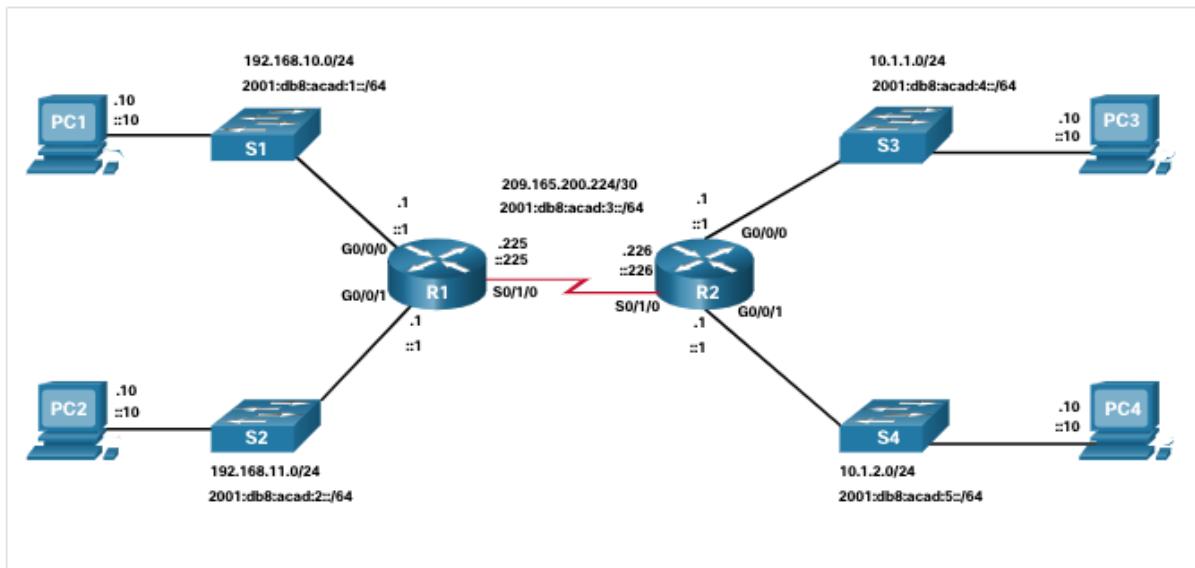
```

Nastavené změny v právě běžící konfiguraci je třeba uložit do souboru startup-config, aby došlo k jejich načtení a prosazení i po restartu zařízení.

```
R1# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]

```

Dual Stack topologie



S prosazováním a masivnějším nasazováním protokolu IPv6 je pravděpodobné, že se setkáte s implementací, která zahrnuje oba protokoly, tedy IPv4 i IPv6. Vnitřní síť dál využívají starší protokol, kdežto na vnějších rozhraních routerů již IPv6. Tak jak to je uvedeno na příkladu topologie dual stack na obrázku, kde je tato použita k předvedení konfigurace rozhraní IPv4 a IPv6 routeru.

Konfigurace rozhraní směrovače

Směrovače podporují síť LAN a WAN, navíc mohou propojovat různé typy sítí, proto podporují mnoho typů rozhraní. Například G2 ISR mají jedno nebo dvě integrovaná rozhraní Gigabit Ethernet a sloty pro vysokorychlostní rozhraní WAN (HWIC) pro přizpůsobení dalším typům síťových rozhraní, včetně sériových, DSL a kabelových rozhraní.

Aby rozhraní bylo k dispozici:

- musí být **nakonfigurováno nejméně jednou IP adresou**. K tomu se použije konfigurační příkaz pro rozhraní **ip address <ip-address> <subnet-mask>** nebo **ipv6 address <ipv6-address/prefix>**.
- musí být **Aktivované**. Ve výchozím stavu nejsou rozhraní routerů aktivována, jsou v režimu **shutdown**. Povolení rozhraní neboli aktivace se provádí pomocí příkazu **no shutdown**. Dále, aby byla fyzická vrstva aktivní, musí být rozhraní připojeno také k jinému zařízení (rozbočovač, přepínač nebo jiný směrovač).
- Musí být popsáno**. Volitelně lze rozhraní nakonfigurovat s popisem až 240 znaků, přičemž preferované jsou popisky krátké a výstižné. Je dobrým zvykem konfigurovat popis na každém rozhraní. V produkčních sítích se výhody popisů rozhraní rychle realizují, protože jsou užitečné při řešení potíží a při identifikaci připojení a kontaktních údajů třetích stran. V Cisco praktických testech je neuvedení popisku kvalifikováno jako chyba a důvod pro nepřidělení bodů za konfiguraci rozhraní.

IPv4 Loopback rozhraní

Dalším běžným krokem konfigurace směrovačů Cisco IOS je povolení rozhraní zpětné smyčky (loopback).

- Rozhraní zpětné smyčky je **logické** rozhraní, které je vnitřní pro směrovač a není přiřazeno k žádnému fyzickému portu, nikdy se jeho prostřednictvím nelze připojit k žádnému jinému zařízení. Považuje se za softwarové rozhraní, které se automaticky uvede do stavu „up“, pokud router funguje.
- Loopback rozhraní je užitečné při testování a správě zařízení Cisco IOS, protože zajišťuje, že alespoň jedno rozhraní bude vždy k dispozici. Lze jej například použít pro účely testování, jako je testování interních směrovacích procesů, použitím emulace sítí za routerem.
- Loopback rozhraní se také běžně používají v laboratorních prostředích k vytváření dalších rozhraní. Můžete například vytvořit více rozhraní zpětné smyčky na routeru a simulovat tak více sítí pro účely konfigurace a testování. Adresa IPv4 pro každé rozhraní zpětné smyčky musí být jedinečná a nepoužívaná jakýmkoli jiným rozhraním.
- V CCNA2 kurzu síťové akademie CISCO se praktických cvičeních často používá loopback rozhraní k simulaci odkazu na internet.
- Povolení a přiřazení adresy loopbacku je jednoduché. V globálním konfiguračním módu se zadáním příkazu **interface loopback <číslo>** rozhraní aktivuje a následujícím příkazem **ip address <ip-adresa> <maska>** již v modu konfigurace rozhraní zadá jeho adresa a maska.

```
Router(config)# interface loopback number
Router(config-if)# ip address ip-address subnet-mask
```

1.5 Ověření přímo propojených sítí

Existuje několik parametrů příkazu **show**, které lze použít k ověření činnosti a konfigurace rozhraní.

Mezi zvlášť užitečné k rychlé identifikaci stavu rozhraní patří tyto:

- **show ip interface brief a show ipv6 interface brief** - Příkazy zobrazují souhrn pro všechna rozhraní, včetně adresy IPv4 nebo IPv6 rozhraní a aktuálního provozního stavu.
- **show running-config interface interface-id** - Zobrazí se příkazy aplikované na zadané rozhraní.
- Pouhý **show running-config** vypíše celou běžící konfiguraci, což nemusí být přehledné, zvlášť pro případ, kdy chceme jen výpis daného rozhraní.
- **show ip route a show ipv6 route** - Příkazy zobrazují obsah směrovací tabulky IPv4 nebo IPv6 uložený v paměti RAM. V systému Cisco IOS 15 by se aktivní rozhraní měla objevit ve směrovací tabulce se dvěma souvisejícími položkami identifikovanými kódem „C“ (připojeno) nebo „L“ (lokálně). V předchozích verzích IOS se objeví pouze jedna položka s kódem „C“. Pokud hledaná přímo připojená síť ve výpisu není, nejspíš je rozhraní neaktivní nebo fyzicky odpojené.

Dále si ještě uvedeme některé detailněji.

Výstup příkazů **show ip interface brief a show ipv6 interface brief** lze použít k rychlému odhalení stavu všech rozhraní na směrovači. Můžete ověřit, zda jsou rozhraní aktivní a funkční, jak je uvedeno v části Status „up“ a Protocol „up“, jak je znázorněno v příkladu. Jiný výstup by znamenal problém s konfigurací.

Výstup příkazu **show ipv6 interface brief** zobrazuje dvě nakonfigurované adresy IPv6 na rozhraních. První, global unicast address, IPv6 adresa byla zadána ručně. Druhá adresa, která začíná znaky FE80, je

link local address rozhraní. Link local address je automaticky přidána do rozhraní, kdykoli je přiřazena global unicast address. Sítové rozhraní IPv6 musí mít místní adresu odkazu, ale nemusí to nutně být global unicast address.

Příkaz **show ipv6 interface gigabitether net 0/0/0** zobrazuje stav rozhraní a všechny adresy IPv6 patřící k rozhraní. Spolu s link local address a global unicast address obsahuje výstup multicast adresy přiřazené k rozhraní, ty začínají znaky FF02, jak je vidět na příkladu.

Výstup příkazu **show running-config interface** zobrazí aktuální příkazy aplikované na zadané rozhraní, tak jak je znázorněno v příkladu.

Následující dva příkazy slouží ke shromažďování podrobnějších informací o rozhraní:

- **show interfaces** zobrazuje informace o rozhraní a počet toků paketů pro všechna rozhraní v zařízení.
- **show ip interface** a **show ipv6 interface** - Zobrazuje informace týkající se IPv4 a IPv6 pro všechna rozhraní směrovače.

Výstup příkazů **show ip route** a **show ipv6 route** odhaluje tři přímo připojené sítě a tři rozhraní z konfigurace routeru.

Jde o tzv routovací tabulku, kde je vidět ke kterému rozhraní je přiřazena jaká IPv4 nebo IPv6 adresa rozhraní. Ta je na něm jednoznačně identifikována pomocí masky /32 pro IPv4 nebo /128 pro IPv6 protokol. Administrativní vzdálenost přímo připojených sítí – směrů je nula.

Tyto routy - směry slouží k tomu, aby směrovač mohl zpracovávat pakety určené pro danou IP adresu.

Písmeno „C“ vedle trasy ve směrovací tabulce označuje, že se jedná o přímo připojenou síť. Pokud je rozhraní směrovače nakonfigurováno s globální adresou unicast a je ve stavu „up/up“, přidá se předpona IPv6 a délka předpony do směrovací tabulky IPv6 jako připojená trasa.

Globální adresa unicast vysílání IPv6 použitá na rozhraní je také nastavena ve směrovací tabulce jako lokální trasa. Lokální trasa má předponu / 128. Lokální trasy používá směrovací tabulka k efektivnímu zpracování paketů s adresou rozhraní směrovače jako cíle.

Filtrování výstupů příkazů show

Příkazy, které generují počtem řádků okno terminálu, neboli více obrazovek výstupu, jsou ve výchozím nastavení pozastaveny po 24 řádcích. Na konci pozastaveného výstupu se zobrazí text - **More** -. Stisknutím klávesy **Enter** se zobrazí další řádek a stisknutím mezerníku se zobrazí další sada řádků. Pomocí příkazu **terminal length** lze definovat počet řádků, které se mají zobrazit. Hodnota 0 (nula) brání směrovači v zastavení mezi obrazovkami výstupu.

Další velmi užitečnou funkcí, která zlepšuje uživatelské prostředí v rozhraní příkazového řádku, je filtrování výstupu **show**. Filtrační příkazy lze použít k zobrazení konkrétních částí výstupu. Povolit příkaz filtrování lze zadáním znaku (| - pipe) za příkazem show a poté zadáním parametru filtru a výrazu filtrování.

Po znaku (|) lze konfigurovat čtyři parametry filtrování:

- **section** - Zobrazí celou část, která začíná výrazem filtrování.
- **include** - Zahrnuje všechny výstupní řádky, které odpovídají filtračnímu výrazu.

- exclude - Vyloučí všechny výstupní řádky, které odpovídají filtrujícímu výrazu.
- begin - Zobrazuje všechny výstupní řádky do určitého bodu, počínaje řádkem, který odpovídá filtračnímu výrazu

Funkce historie příkazů

Funkce historie příkazů je užitečná, protože dočasně ukládá seznam vykonaných příkazů, které tak mohou být znova vyvolány.

Chcete-li vyvolat příkazy ve vyrovnávací paměti historie, stiskněte **Ctrl + P** nebo klávesu **šipka nahoru**. Výstup příkazu začíná nejnovějším příkazem. Opakujte posloupnost kláves, abyste postupně vyvolali starší příkazy. Chcete-li se vrátit k novějším příkazům ve vyrovnávací paměti historie, stiskněte **Ctrl + N** nebo klávesu **šipka dolů**. Opakujte posloupnost kláves, abyste si postupně vyvolali novější příkazy.

Ve výchozím nastavení je historie příkazů povolena a systém zachycuje posledních 10 příkazových řádků ve své vyrovnávací paměti historie. K zobrazení obsahu vyrovnávací paměti použijte příkaz **show history** v privilegovaném EXEC módu.

Jako praktické se může jevit navýšení počtu příkazových řádků, které historie vyrovnávací paměti zaznamenává pouze během aktuální relace terminálu. Pomocí příkazu **terminal history size** v uživatelském EXEC módu můžete zvětšit nebo zmenšit velikost vyrovnávací paměti.

Počítačové sítě 2

2, Koncepce přepínání a VLAN

Název bloku	Náplň
Přenosílání rámciů	Vysvětuje, jak jsou rámce přenosílány v přepínacích sítích.
Přepínací domény	Porovnání kolizní domény s broadcastovou doménou.
VLAN	Vysvětuje princip fungování VLAN
VLAN v rozsáhlých sítích	Vysvětuje koncepci a distribuci VLAN v sítích středních a velkých firem

2.1 Přeposílání rámců

Síťová zařízení k propojení segmentů a koncových zařízení používají porty, jimiž data ve formě rámců protékají oběma směry.

S rámci vstupujícími nebo opouštějícími rozhraní jsou spojeny dva výrazy:

- **Ingress** – vstup do rozhraní
- **Egress** – výstup z rozhraní

Přepínač přeposílá rámce kdo cíle na základě znalosti vstupního rozhraní a cílové MAC adresy.

Svoji tabulku MAC adres využívá k tvorbě rozhodnutí o přeposílání.

Poznámka: Přepínač nikdy nedovolí, aby byl provoz přesměrován na rozhraní, na kterém tento provoz přijal.

Tabulka MAC adres přepínače

Přepínač použije k určení výstupního rozhraní cílovou MAC adresu. Než přepínač může učinit toto rozhodnutí, musí zjistit, na jakém rozhraní se nachází cíl. K tomu vytváří a udržuje tabulku aktivních MAC adres, známou také jako tabulka CAM (Content Addressable Memory). Do tabulky zaznamená zdrojovou MAC adresu spolu s portem, na kterém byla přijata.

Metoda učení a přeposílání přepínače

Pro přeposílání rámců přepínač používá metodu sestávající ze dvou kroků:

Prvním je Učení, kdy v příchozím provozu zkoumá zdrojové adresy a porovnává je se svými záznamy. Pokud zachycená zdrojová MAC adresa v tabulce chybí, přidá ji. Záznamy jsou dynamické. Pokud v tabulce MAC adresa je, obnoví pro ni nastavení časového limitu zpět na dalších 5 minut.

Druhým krokem je přeposílání, v němž se prozkoumají cílové adresy. Pokud je cílová MAC adresa v tabulce MAC adres, rámec je přeposlán na přiřazený port. Pokud cílová MAC adresa v tabulce není, rámec se rozešle na všechna rozhraní kromě toho, kde byl přijat. K danému je možné, pro lepší pochopení, shlédnout video v souvisejícím kurzu na netacad.

Switch Forwarding Methods

Přepínače používají software běžící na integrovaných obvodech specifickým pro aplikaci (ASIC), určeným k velmi rychlému rozhodování.

Poté co přepínač obdrží rámec, použije jednu ze dvou metod k rozhodnutí o přeposílání.

Bud' použije **Store-and-forward switching**, při níž přijme celý rámec a ověří, že je platný. Store-and-forward switching je Ciscem preferovaná metoda přepínání.

Nebo **Cut-through switching**, kde přeposílá rámec okamžitě po spárování zjištění cílové MAC adresy příchozího rámce a výstupního bodu.

Store-and-forward pracuje s kompletním rámcem a vychází z toho její dvě základní charakteristiky:

Error Checking – Přepínač zkontroluje Frame Check Sequence (FCS) zda nemá chyby CRC součtu. Vadné rámců budou zahozeny.

Buffering – Vstupní rozhraní bude ukládat rámec po dobu kontroly FCS. To také umožňuje přepínači přizpůsobit se případnému rozdílu v rychlostech mezi portem vstupu a výstupu rámce.

Metoda Cut-through přeposílá rámec bezprostředně po určení cílové MAC adresy.

Pomocí metody Fragment (Frag) Free zkонтroluje cílovou MAC adresu a zároveň zajistí, aby přeposílaný rámec měl alespoň 64 bajtů. Tento krok eliminuje vznik runts paketů, tedy paketů menších než je uvedená minimální velikost.

Koncept Cut-Through switching je vhodný pro přepínače vyžadující latenci pod 10 mikrosekund. Rámce přeposílá ještě než je celé přijme, proto neprovádí kontrolní součty FCS a může tedy šířit chyby. S tím souvisí, že pokud přepínač šíří příliš mnoho chyb, mohou nastat problémy se šírkou pásma. Omezením také je, že nelze podporovat porty s různou rychlostí na vstupu a na výstupu.

2.2 Přepínání domén

Kolizní domény

Přepínače jsou nezbytné prvky v přepínaných sítích, neboť eliminují kolizní domény a snižují přetížení.

- Pokud je na spojení full duplex, kolizné domény jsou naprostě eliminované.
- Pokud je provoz vedený v half duplexu u jednoho nebo více zařízení, vznikne kolizní doména a nastane problém se šírkou pásma.
- Většina zařízení, včetně těch od společností Cisco a Microsoft, používá ve výchozím nastavení automatické vyjednávání pro duplexní provoz a rychlosť.

Broadcastová doména se rozprostírá přes všechna zařízení L1 nebo L2 vrstvy v síti LAN. Pouze zařízení pracující na L3 vrstvě (především jde směrovače) rozbjí broadcastovou doménu, nazývanou také broadcastová MAC doména. Broadcastová doména sestává ze všech zařízení sítě LAN které přijímají broadcastové zprávy. Vychází to z principu, kdy přepínač na L2 vrstvě přijme broadcastovou zprávu a přepoše ji na všechna rozhraní mimo toho, z něhož zpráva přišla. Příliš mnoho broadcastu může způsobit přetížení a špatný výkon sítě. Příkladem problému může být rozsáhlá lan s jednou broadcastovou doménu ve firemním prostředí, kdy je dán pevný začátek pracovní doby a pracovníci startují své počítače v krátkém časovém intervalu. S rozvojem firem jde často ruku v ruce zvýšení počtu zařízení, což na 1 či 2 vrstvě způsobí rozšíření broadcastové domény.

Ke zmírnění přetížení sítě a k eliminaci kolizí moderní přepínače používají tabulkou MAC adres a full-duplex režim.

Následující protokoly nabízejí funkce, které přetížení sítě zmírňují:

Fast Port Speeds u výkonných HW modelů nabízejí porty s rychlosťí až 100gbps

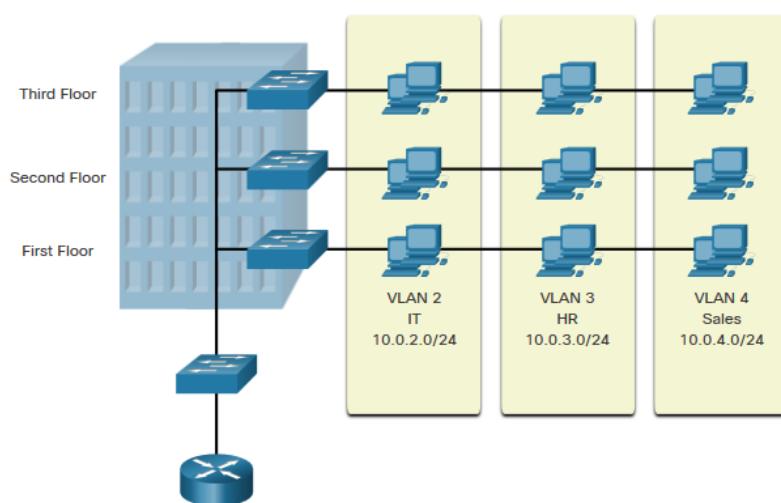
Fast Internal Switching používá ke zlepšení výkonu rychlou vnitřní sběrnici nebo sdílenou paměť

Large Frame Buffers umožňuje při zpracování velkého množství snímků jejich dočasně uložení.

High Port Density znamená vysokou koncentraci portů na jednom přepínači, což zajišťuje větší místní provoz s menším přetížením.

3.1 Přehled VLAN

VLANy jsou logická spojení mezi zařízeními s podobnou funkcí. Umístění zařízení do různých sítí VLAN přináší specifické vlastnosti. Dovoluje segmentaci různých skupin zařízení na stejných přepínačích, jejich nová organizace přináší lepší správu. Broadcast, multicast a unicast jsou pak izolovány v jednotlivých VLANech. Každá VLAN má svůj vlastní jedinečný rozsah IP adres což síť drobí na menší broadcastové domény.



Výhody lze shrnout takto:

Výhoda	Popis
Menší broadcastové domény	Rozdelení sítě LAN snižuje počet broadcastových domén
Zvýšená bezpečnost	Společně mohou komunikovat pouze uživatelé ve stejné síti VLAN
Zvýšená IT efektivita	VLANy mohou seskupovat zařízení s podobnými požadavky, např. učitelé vs. studenti
Snížení nákladů	Jeden HW přepínač může podporovat více skupin nebo VLANů
Lepší výkon	Malé broadcastové domény redukují provoz a zlepšují šířku pásma
Jednodušší správa	Podobné skupiny využívají podobné aplikace a další síťové prostředky

Rozlišujeme několik typů VLAN.

V první řadě jde o Defaultní VLAN neboli VLAN 1. S ní se váže:

- Defaultní VLAN (je předinstalovaná)
- Defaultní nativní VLAN (nepoužívá tagované rámce)
- Defaultní VLAN pro správu (což není nezvratné a z pohledu bezpečnosti může být i kritické)
- Nelze ji smazat ani přejmenovat, avšak, lze ji nepoužívat

Což je zmíněno i jako Cisco doporučení, abychom tyto výchozí funkce přiřadili jiným VLANám.

Mimo defaultní VLAN je tu **Datová VLAN**, určena pro provoz generovaný uživateli (e-mailový a webový provoz, sdílení,...).

VLAN 1 je výchozí datová VLAN, neboť všechna rozhraní jsou přiřazena ve výchozí konfiguraci této VLAN.

Nativní VLAN se používá pouze pro trunková spojení. Všechny rámce jsou definicí 802.1Q tagované pro trunkové spojení, kromě těch, které pocházejí z nativní VLAN – jak jsem již naznačil.

VLAN pro správu se používá pro provoz SSH / Telnet. Neměla by přenášet data společně s provozem koncového uživatele. Typicky je to VLAN, která je SVI pro přepínač vrstvy 2 (neboli má přiřazenou IP adresu).

Hlasová VLAN slouží pro přenos hlasového provozu, pro který je vyžadována samostatná VLAN, protože ten vyžaduje:

- Zaručenou šířku pásma
- Vysokou prioritu QoS
- Schopnost zabránit přetížení
- A má latenci menší než 150 ms od zdroje k cíli.

Pokud je tato funkctionalita ve firemní síti požadována, pak celá síť musí být navržena tak, aby přenos hlasu podporovala.

3.2 VLANy v prostředí mnoha přepínačů

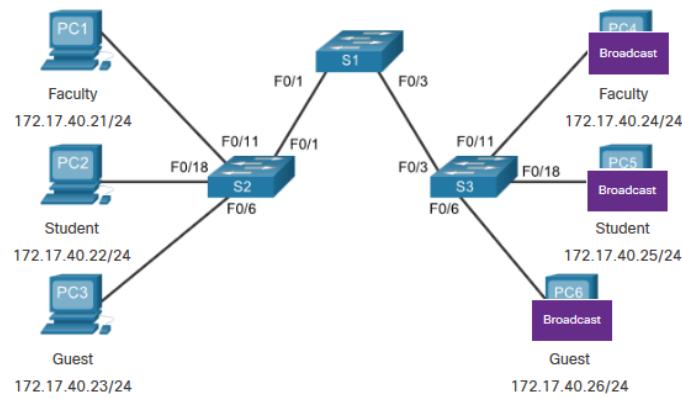
K rozprostření a dostupnosti VLAN nad fyzickou topologií sítě se využívá trunkových spojení.

Trunk znamená realizaci point-to-point spojení mezi dvěma síťovými zařízeními.

Funkcí CISCO trunků je umožnit použití více než jedné VLAN ve firemní síti a hlavně rozšíření těchto VLAN v ní. Realizace je postavena na protokolu 802.1Q, tedy tzv. trunkování.

Pokud bude, jak je na obrázku uvedeno, existovat síť bez VLAN, budou všechna zařízení připojená k přepínačům přijímat veškerý provoz unicast, multicast a broadcast což silně ovlivní šířku pásma a povede k bottleneckům neboli úzkým místům, kde bude provoz nabývat na latenci případně působit výpadky.

V sítích s implementovanými VLAN je provoz unicast, multicast a broadcast omezen na jen danou VLAN. Protože má každá VLAN vlastní adresní prostor, bez zařízení 3 vrstvy nemohou zařízení z různých VLANách spolu komunikovat.



Identifikace VLAN pomocí tagu

Identifikace VLAN se provádí pomocí tagů. Dle protokolu IEEE 802.1Q se vytvářejí se na úrovni rámce vřazením 4 bajtového bloku do hlavičky, jak je znázorněno na obrázku.

Blok obsahuje čtyři části popsané v tabulce.

802.1Q VLAN pole tagu	Funkce
Typ	<ul style="list-style-type: none"> 2-bajtové pole s hexadecimální hodnotou 0x8100 Toto se označuje jako Tag Protocol ID (TPID)
Priorita uživatele	3-bitová hodnota
Canonical Format Identifier (CFI)	1-bitová hodnota, která může podporovat rámce token ring na Ethernetu
VLAN ID (VID)	12-bitový VLAN identifikátor, jenž může adresovat až 4096 VLANů

Když se vytvoří tag, je nutné přepočítat kontrolní součet rámce FCS. Při odeslání rámce z přepínače na koncová zařízení, musí být tag odstraněn a FCS přepočítána zpět na původní hodnotu.

Nativní VLANy a 802.1Q tagování

Tvorbu 802.1Q trunků rozumíme sled úkonů. Tagování se obvykle provádí na všech VLAN. Použití nativní VLAN bylo navrženo pro starší aplikace, jako je hub použitý v příkladu, který tagům nerozumí a není schopen je korektně zpracovat. Provoz mířený na něj musí být od nich oproštěný. Pokud se konfigurace aktivního prvku nezmění, je VLAN1 definována jako nativní VLAN. Oba konec trunkového spojení musí být nakonfigurovány se stejnou nativní sítí VLAN. Každý trunk se konfiguruje samostatně, takže je možné mít různé nativní VLAN na separátních trunciích.

Tagování hlasových VLANů přináší jistá specifika a souvisí s HW, který je jejím prostřednictvím provozovaný. Jde o VoIP telefon, což je přepínač/switch s třemi porty:

P1 – vstupní port pro propojení s přepínačem podporujícím VoIP

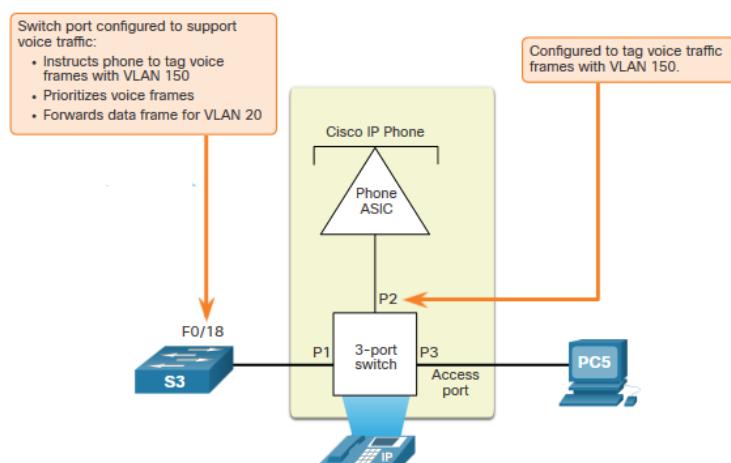
P2 – port s VLAN pro IP telefonii

P3 – port pro připojení např. kancelářského PC do podnikové sítě

V prostředí s omezenou strukturovanou kabeláží je P3 jedinou možností jak provozovat na jedné datové zásuvce současně vybavit pracovníka IP telefonem a PC připojeným do datové sítě.

Přepínač S3 bude pomocí CDP informovat telefon o Voice VLAN. Telefon taguje svůj vlastní provoz (hlas) a může nastavit náklady na službu – cost of service (CoS), což je QoS na druhé vrstvě.

Telefon dále může, ale nemusí tagovat rámce z PC.



3.3 Konfigurace VLAN

V modelové řadě Cisco Catalyst najeznete přepínače Catalyst 2960 a 3650, které podporují přes 4000 VLAN.

Podobně jako u portů TCP a UDP komunikace i zde najdeme rozdělení dostupného rozsahu do sekcí, tak jak je uvedeno v tabulce.

Normální rozsah VLAN 1 – 1005	Rozšířený rozsah VLAN 1006 - 4095
Používá se v malých a středních podnicích	Používají poskytovatelé služeb
1002 - 1005 je vyhrazeno pro starší VLANy	Jsou v Running-Config
1, 1002 – 1005 se vytvářejí automaticky a nelze je smazat	Podporuje méně funkcí VLAN
Uložené v souboru vlan.dat ve flash paměti	Vyžaduje konfigurování VTP
VTP lze synchronizovat mezi přepínače	

VLAN se ukládají mimo starup-configuration soubor načítaný IOS při startu zařízení. VTP – VLAN trunking protocol – automatický systém předávní informací o dostupných VLAN (centrální správa VLAN nad všemi prvky s podporou protokolu v jedné doméně)

Z výpisu show vlan brief je pak vidět, že switch při prvním startu nabízí VLAN 1 a jsou zde zmíněné předdefinované 1002 – 1005, které nelze odstranit.

Switch# show vlan brief			
VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gi0/1, Gi0/2
1002	fddi-default	act/unsup	
1003	token-ring-default	act/unsup	
1004	fdtnet-default	act/unsup	
1005	trnet-default	act/unsup	

Existuje sada příkazů globálního konfiguračního módu pro správu VLAN, zde je uvedeno vytvoření VLAN, která se ukládá, jak již bylo zmíněno, do souboru vlan.dat

Úkol	IOS příkaz
Vstup do globálního konfiguračního módu.	Switch# configure terminal
Vytvoření VLAN s platným ID.	Switch(config)# vlan vlan-id
Jedinečný název pro identifikaci VLAN.	Switch(config-vlan)# name vlan-name
Návrat do privilegovaného EXEC módu.	Switch(config-vlan)# end
Vstup do globálního konfiguračního módu.	Switch# configure terminal

Jakmile je VLAN vytvořena, můžeme ji přiřadit k požadovaným portům. Postup pro přiřazení VLAN k jednomu portu je uvedený v tabulce. Proces pro hromadné přidělení spočívá v doplnění příkazu interface o parametr range a interface id vymezení od-do

Jeden přístupový port lze přiřadit pouze jedné datové VLAN. Může však být také přiřazen k jedné hlasové VLAN, jsou-li telefon a koncové zařízení provozované ze stejného portu přepínače. Pak bude postup konfigurace následující.

Když budeme chtít vytvořit společný port a definovat jak hlasovou, tak i datovou VLAN, kromě přiřazení datové VLAN přiřadíme také hlasovou VLAN a zapneme QoS na rozhraní pro hlasový provoz. Novější přepínače Catalyst vytvoří VLAN automaticky, pokud již neexistuje, když ji přiřazuje k rozhraní. QoS se probírá v navazujícím předmětu sítí. Související příkaz pro nastavení je **mls qos trust [cos | device cisco-phone | dscp | ip-precedence]**.

K ověření nastavení VLAN použijte příkaz **show vlan**. Jeho úplná syntaxe je:

show vlan [brief | id *vlan-id* | name *vlan-name* | summary]

Parametry s popisem jsou v následující tabulce:

Úkol	Možnost příkazu
Zobrazit název VLAN, stav a její porty, jednu VLAN na řádek.	brief
Zobrazit informace o identifikovaném ID VLAN.	id <i>vlan-id</i>
Zobrazit informace o identifikovaném názvu VLAN. <i>vlan-name</i> je ASCII řetězec od 1 do 32 znaků.	name <i>vlan-name</i>
Zobrazit souhrnné informace o VLAN.	summary

Změna přiřazení VLAN portu

Změnit přiřazení VLAN na portech lze provést několika způsoby. Znovu použijte příkaz **switchport access vlan *vlan-id***, kde se novým *vlan-id* přepíše dosavadní platné. Použijte příkaz **no switchport access vlan**. Tím dojde k přeřazení konfigurovaného rozhraní zpět do VLAN 1.

Příkazem **show vlan brief** nebo **show interface fa0/18 switchport** lze ověřit korektní přiřazení VLANů.

3.4 VLAN Trunks

K propagaci VLAN napříč sítí je potřeba definovat trunk porty.

Pro konfiguraci Cisco přepínačů použijeme posloupnost příkazů pro konfiguraci trunk portů, které mají předávat tagované rámce na další aktivní prvky.

Příkazem **conf t** vstoupíme do globálního konfiguračního modu. Následuje výběr rozhraní, které chceme převést do trunk modu, příkazem **interface *interface-id***. Permanentní trunk mod provedeme příkazem **switchport** s parametrem **mode trunk**. Stejným příkazem s jinými parametry lze přenastavit nativní VLAN na jinou než je výchozí VLAN 1 a to **switchport trunk native vlan *vlan-id***. V posledním kroku konfigurace zvolíme VLANy, které se budou do trunku agregovat. Mohu udělat výběr nebo povolit všechny. Je třeba si uvědomit, že nastavení ALL s sebou může nést nadbytečné šíření provozu na prvky, které nevyužívají všechny dostupné VLAN. Z pohledu zabezpečení se tak vytváří potenciálnímu útočníkovi prostor pro průnik do kompletní sítě.

Pro ověření nastavení trunkového módu a nativní VLAN slouží příkaz **sh int <port> switchport**. Je z něj patrné administrativní nastavení na trunk, operativní nastavení trunku (funkční), případně zapouzdření dot1q.

Nativní VLAN by měla být přenastavena, třeba na VLAN 99. Všechny VLANy vytvořené na přepínači, nebo jejich výběr, mohou být posílány přes daný trunk port.

Přestavení trunku do defaultního stavu

Proces přenastavení trunku do výchozího nastavení lze provést příkazem

no switchport trunk allowed vlan

no switchport trunk native vlan

Dále je potřeba resetovat režim trunk na access mode příkazem

switchport mode access

Obnovené nastavení lze následně ověřit příkazem **show interface fa0/1 switchport**.

3.5 Dynamic Trunking Protocol

Dynamic Trunking Protocol (DTP) je proprietární protokol společnosti Cisco. Lze jej využít ve středních a rozlehlých firemních sítích, kde dochází k častým změnám a je třeba na ně reagovat rychle, nejlépe automaticky.

DTP má tyto charakteristiky:

- Defaultně je na přepínačích Catalyst 2960 a 2950 spuštěný
- Funkce Dynamic-auto je výchozím stavem portů na přepínačích 2960 a 2950
- Dá se vypnout příkazem **nonegotiate**
- Může být znova spuštěn nastavením rozhraní na **dynamic-auto**
- Z praktického a bezpečnostního pohledu se lze nastavením přepínače na statický trunk nebo statický přístup pomocí příkazů **switchport mode trunk** nebo **switchport mode access**, vyhnout problémům s vyjednáváním.

Při nastavování rozhraní lze definovat různé vyjednávání. Příkaz **switchport mode** jsme si již představili. Nabízí několik variant použitelných parametrů uvedených v tabulce.

Možnost	Popis
access	Režim trvalého přístupu, a vyjednávání o převedení sousedního spojení na přístupové spojení
dynamic auto	Stane se trunkovým rozhraním, pokud sousedící rozhraní je nastavené na trunk anebo na mód desirable
dynamic desirable	Aktivně hledá možnost přejít na trunk pomocí vyjednávání s dalšími rozhraními v módu auto nebo desirable
trunk	Režim trvalého trunku a vyjednává proměnu sousedícího spojení na trunkové.

Chcete-li dtp vyjednávání zastavit, použijte příkaz **switchport nonegotiate**.

Jak ale porty nastavené na různé možnosti spolu komunikují? Výsledky konfigurace DTP jsou uvedené v následující tabulce. Jak je zřejmé, ne všechny stavby vedou k zabezpečení komunikace.

S1 \ S2	Dynamic Auto	Dynamic Desirable	Trunk	Access
Dynamic Auto	Access	Trunk	Trunk	Access
Dynamic Desirable	Trunk	Trunk	Trunk	Access
Trunk	Trunk	Trunk	Trunk	Limited connectivity
Access	Access	Access	Limited connectivity	Access

Pro ověření nastavení DTP módu je třeba použít příkaz **show dtp interface interface-ID**. Výchozí konfigurace DTP je závislá na verzi a platformě Cisco IOS. Osvědčené postupy doporučují, aby byla rozhraní pevně nastavena buď na access nebo trunk a DTP vypnuto

Počítačové sítě 2

2, Inter-VLAN Routing

Název bloku	Cíle bloku
Inter-VLAN Routing Operation	Popis možnosti konfigurace směrování mezi VLAN.
Router-on-a-Stick Inter-VLAN Routing	Konfigurace směrování mezi VLAN použitím router-on-a-stick.
Inter-VLAN Routing using Layer 3 Switches	Konfigurace směrování mezi VLAN použitím přepínání na vrstvě 3.
Troubleshoot Inter-VLAN Routing	Odstraňování běžných problémů s konfigurací směrování mezi VLAN.

4.1 Inter-VLAN Routing Operation

Co tedy rozumíme routováním nebo též směrováním mezi VLANami.

VLANy se používají k segmentování přepínaných sítí vrstvy 2 z různých důvodů, které jsme si uvedli v předchozí přenášce. Bez ohledu na důvod je jasné, že koncová zařízení v jedné VLAN nemohou komunikovat s koncovými zařízeními v jiné VLAN. Uvedené platí, pokud nemáme k dispozici směrovač nebo přepínač 3 vrstvy, který také poskytuje směrovací služby.

Definici můžeme postavit ve smyslu, že Směrování mezi VLAN je proces přeposílání síťového provozu z jedné VLAN do jiné VLAN.

Máme k dispozici tři možnosti směrování mezi VLAN:

- **Starší způsob směrování mezi VLAN** - Jde o starší řešení, které se nedá dobře škálovat.
- **Router-on-a-Stick** - je časté a pro malou a střední síť přijatelné řešení .
- **Použití L3 přepínače pomocí přepínaných virtuálních rozhraní (SVI)** je nejlépe škálovatelné řešení pro střední a velké organizace.

Tyto možnosti si nyní probereme detailněji.

Starším způsobem směrování mezi VLAN rozumíme způsob, který byl dostupný na počátcích , kdy byla VLAN uvedena.

Spoléhalo se na použití routeru s více ethernetovými rozhraními. Každé rozhraní směrovače bylo připojeno k portu přepínače v různých sítích VLAN. Rozhraní směrovače sloužila jako defaultní gateways k místním hostům v podsíti VLAN. Využívá fyzických rozhraní a má značné omezení. Není přiměřeně škálovatelné, protože směrovače mají omezený počet fyzických rozhraní. Vyžadování jednoho fyzického rozhraní směrovače na každou VLAN rychle vyčerpá jejich kapacitu na směrovači.

Tuto metodu směrování mezi VLAN si zde uvádíme pro úplnost, dnes již v přepínaných sítích není implementována.

Metoda směrování mezi VLANami **Router-on-a-Stick** překonává právě popsaná omezení předchozí metody především v tom, že pro směrování provozu mezi více VLAN potřebuje pouze jedno fyzické ethernetové rozhraní.

Toto rozhraní je nakonfigurováno jako trunkové 802.1Q rozhraní a je připojeno k trunkovému rozhraní na L2 přepínači. Když se věc podíváme detailněji, rozhraní směrovače je nakonfigurováno pomocí dílčích rozhraní k identifikaci směrovaných VLAN sítí.

Tato dílčí rozhraní chápejme jako virtuální rozhraní. Všechna jsou spojená s jediným fyzickým ethernetovým rozhraním směrovače. Jejich konfigurace je otázkou softwarového nastavení. Každé toto podrozhraní je nezávisle konfigurováno s IP adresou a přiřazením k VLAN. Logické směrování usnadňuje, že jsou podrozhraní konfigurována pro různé podsítě s odpovídajícím přiřazením k VLAN. Když tagovaný VLAN provoz ze směrovače vstupuje do rozhraní směrovače, je předán do VLAN podrozhraní. Po rozhodnutí o směrování na základě cílové IP adresy v síti směrovač určí výstupní rozhraní pro přenos. Pokud je výstupní rozhraní nakonfigurováno jako podrozhraní 802.1q, jsou

datové tagované rámce ze zdrojové VLAN otagované pomocí nové VLAN a odeslány zpět z fyzického rozhraní.

Metoda router-on-a-stick má také své limity, nelze ji použít na víc než 50 VLAN.

Směrování mezi VLANy pomocí přepínače 3. vrstvy je moderní metoda realizace směrování mezi VLAN a spočívá ve využití přepínačů na vrstvě 3 a přepínáných virtuálních rozhraní (SVI). SVI je virtuální rozhraní, které je konfigurováno na přepínači vrstvy 3.

Inter-VLAN SVI jsou vytvořeny stejným způsobem, jakým je nakonfigurováno rozhraní VLAN pro správu. SVI je vytvořen pro VLAN, která existuje na přepínači. Ačkoli je SVI rozhraní virtuální, vykonává pro VLAN stejné funkce jako rozhraní směrovače. Konkrétně, umožňuje zpracování paketů na 3 vrstvě, které jsou odesílány na nebo ze všech portů přepínačů přiřazených k této síti VLAN.

Výhody použití L3 přepínačů pro směrování mezi VLAN jsou tyto:

- Jsou mnohem rychlejší než router-on-a-stick, protože vše je přepínáno a směrováno hardwarově.
- Pro směrování není potřeba realizovat externí spoj z přepínače na směrovač.
- Nejsou omezeny jedním spojem, protože EtherChannel druhé vrstvy lze použít jako trunkové spoje mezi přepínači pro zvýšení šířky pásma.
- Latence je mnohem nižší, protože data neopouštějí přepínač, aby byla směrována do jiné sítě.
- V LAN kampusech se nasazují častěji než směrovače.

Jedinou nevýhodou je, že L3 přepínače jsou výrazně dražší.

4.2 Router-on-a-Stick Inter-VLAN Routing

Pro vysvětlení použití metody Router on the stick použijeme následující scénář.

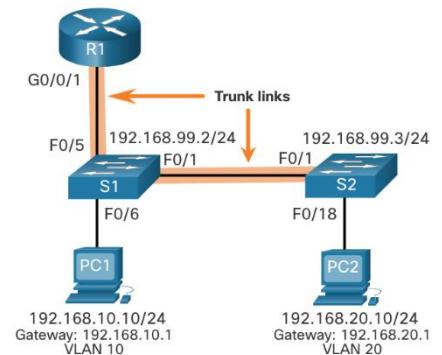
Na obrázku je rozhraní GigabitEthernet 0/0/1 Routeru R1 připojeno k portu FastEthernet 0/5 switche S1. Port FastEthernet 0/1 na switchi S1 je připojen k portu FastEthernet 0/1 switchu S2. Jedná se o trunkové linky, které jsou vyžadovány k předávání provozu uvnitř a mezi VLAN.

Pro směrování mezi VLAN je rozhraní GigabitEthernet 0/0/1 na R1 logicky rozděleno do tří dílčích rozhraní, jak ukazuje tabulka.

Tabulka také zobrazuje tři VLAN, které budou konfigurovány na přepínačích.

Předpokládejme, že R1, S1 a S2 mají počáteční základní konfigurace. V současné době se PC1 a PC2 nemohou navzájem pingnout, protože jsou v samostatných sítích. Pouze S1 a S2 se mohou navzájem pingnout, ale jsou nedostupné pro PC1 nebo PC2, protože jsou také v různých sítích.

Abychom umožnili zařízením prověřit vzájemné propojení, musí být přepínače nakonfigurovány pomocí VLAN a trunkingu a router musí být nakonfigurován pro směrování mezi VLAN.



4.3 Inter-VLAN směrování pomocí přepínačů na vrstvě 3

Směrování Inter-VLAN pomocí metody router-on-a-stick je pro malou až střední organizaci jednoduché. Velký podnik však k zajištění směrování mezi VLAN vyžaduje rychlejší a lépe škálovatelnou metodu.

Směrování Inter-VLAN pomocí metody router-on-a-stick je pro malou až střední organizaci jednoduché. Velký podnik však k zajištění směrování mezi VLAN vyžaduje rychlejší a lépe škálovatelnou metodu.

V LAN v podnikových kampusech se k zajištění směrování mezi VLAN používají L3 přepínače. Ty používají k dosažení vyšší rychlosti, než nabízí směrovače, zpracování paketů pomocí hardwarového přepínání. Přepínače 3 vrstvy jsou také běžně implementovány v podnikových distribučních vrstvách.

Funkcionality L3 přepínače zahrnují schopnost provádět:

- Směrování z jedné VLAN do druhé pomocí více přepínaných virtuálních rozhraní (SVIs).
- Převedení switchportu 2 vrstvy na rozhraní 3 vrstvy, tzv. na směrovaný port. Směrovaný port je podobný fyzickému rozhraní na Cisco IOS směrovači.
- K zajištění směrování mezi VLAN používají přepínače 3 vrstvy SVI. Rozhraní SVI se konfigurují pomocí stejněho příkazu **interface vlan** *vlan-id* používaného k vytvoření SVI pro správu na L2 přepínači. Pro každou směrovanou VLAN musí být vytvořen SVI 3 vrstvy.

K praktickému objasnění problematiky si uvedeme příklad, jehož scénář zahrnuje L3 přepínač a dva uzly.

Na obrázku je L3 přepínač D1 připojený ke dvěma počítačům v různých VLANech. PC1 je ve VLAN 10 a PC2 je ve VLAN 20. Přepínač vrstvy 3 poskytuje zmíněným dvěma hostům službu směrování mezi jejich VLANami.

Konfigurace přepínače na 3 vrstvě spočívá v provedení následujících kroků:

- Vytvoříme VLANy. V našem příkladu jsou použity VLAN 10 a 20.
- Dále vytvoříme SVI VLAN rozhraní. Nakonfigurujeme IP adresu, která poslouží jako default gateway pro hostu v příslušné VLAN.
- Ve třetím kroku nakonfigurujeme přístupové porty, tj přiřadíme příslušný port k požadované VLAN.
- V posledním kroku zapneme IP směrování. Zadáním příkazu **ip routing** v globálním konfiguračním módu povolíme provoz mezi VLAN 10 a 20. Příkaz musí být nakonfigurován tak, aby umožňoval směrování mezi VLAN na přepínači 3 vrstvy pro IPv4.

Pokud mají být sítě VLAN dosažitelné jinými zařízeními vrstvy 3, musí být inzerovány pomocí statického nebo dynamického směrování. Chcete-li povolit směrování na přepínači 3 vrstvy, je potřeba nakonfigurovat směrovaný port. Směrovaný port se vytvoří deaktivací funkce switchportu na portu 2 vrstvy, který je připojen k jinému zařízení 3 vrstvy. Konkrétně, příkazem **no switchport** na rozhraní portu 2 vrstvy jej převede na rozhraní 3 vrstvy. Poté lze rozhraní nakonfigurovat s IPv4 konfigurací pro připojení k routeru nebo jinému přepínači 3 vrstvy.

Počítačové sítě 2

2, Koncepty STP

Název bloku	Cíle bloku
Účel STP	Vysvětlit běžné problémy v redundantní přepínané síti na 2 vrstvě.
Fungování STP	Vysvětlit, jak STP funguje v jednoduché přepínané síti.
Vývoj STP	Vysvětlit, jak funguje Rapid PVST+.

5.1 Účel STP

Základem problému je redundance tras v přepínaných sítích na vrstvě 2.

Téma popisuje příčiny vzniku smyček v síti na vrstvě 2 a stručně vysvětluje, jak funguje spanning tree protokol (STP). Redundance není pochopitelně chybou, ale důležitou součástí hierarchického návrhu pro eliminaci jednotlivých bodů selhání a prevenci narušení síťových služeb. Redundantní sítě vyžadují přidání fyzických cest. Součástí návrhu musí být ale také logická redundancy. Existence alternativní fyzické cesty pro data, která procházejí sítí, umožňuje uživatelům přístup k síťovým prostředkům, a to navzdory případnému narušení cesty. Redundantní cesty v přepínané síti Ethernet však mohou na 2 vrstvě způsobit vznik fyzické i logické smyčky.

Ethernetové lok.sítě vyžadují topologii bez smyček s jedinou cestou mezi libovolnými dvěma zařízeními. Smyčka v ethernetové síti LAN může způsobit trvalé šíření ethernetových rámů, dokud není spojení přerušeno a smyčku nerozpojí. Broadcastové bouře jsme si objasnili již v počítačových sítích 1.

Spanning Tree Protocol (STP) je síťový protokol zabraňující vzniku smyček. Umožnuje zabezpečení redundance spojů mezi přepínači při vytváření bezsmyčkové topologie vrstvy 2.

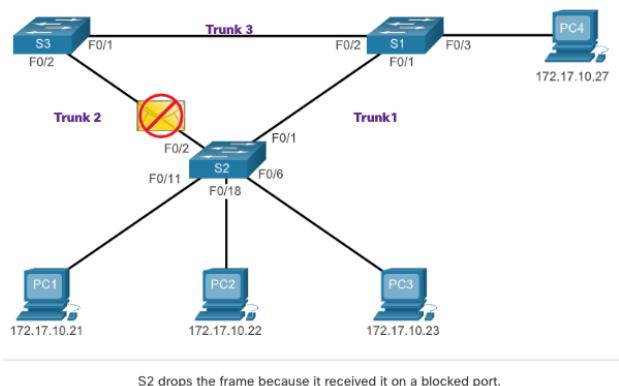
STP logicky/softwarově blokuje fyzické smyčky v síti na vrstvě 2, čímž brání rámům v nekonečném kroužení sítě.

Účelem STP je tedy řešit problémy s redundantními spoji přepínačů. Nelze je chápat jako problém, neboť redundance cest poskytuje více síťových služeb tím, že eliminuje možnost jediného bodu selhání. Problém nastává pokud mezi dvěma zařízeními v síti Ethernet existuje více cest a na přepínačích není implementován spanning tree. Vznikne na druhé vrstvě smyčka. Tato smyčka může mít za následek nestabilitu tabulky MAC adres, přetížení spojení a vysoké využití CPU na přepínačích i koncových zařízeních. Následek je potom nepoužitelnost sítě.

Ethernet na 2 vrstvě nezahrnuje mechanismus pro rozpoznávání a eliminaci nekonečně opakování rámů. Na třetí vrstvě je to již vyřešené. IPv4 i IPv6 obsahují mechanismus, který omezuje počet opakování přenosu paketu síťovým zařízením 3 vrstvy. Směrovač sníží TTL (Time to Live) v každém paketu IPv4 a pole Hop Limit v každém paketu IPv6. Když se tato pole sníží na 0, směrovač paket zahodí. Bohužel, Ethernet a Ethernetové přepínače nemají srovnatelný mechanismus k omezení počtu opakování přenosu na 2 vrstvě. A právě proto byl vyvinut STP speciálně jako mechanismus prevence smyček druhé vrstvy Ethernetu.

Bez povoleného STP se mohou tvořit nekonečné smyčky broadcastu, multicastu a neznámých unicastových rámů. To může rychle poškodit síť, jak už jsem zmínil. Když dojde ke smyčce, tabulka MAC adres na přepínači se bude neustále měnit s aktualizacemi broadcastových rámů, což má za následek nestabilitu databáze MAC adres. Víme, že to může způsobit vysoké nároky na využití CPU. Následkem toho je, že přepínač nebude schopen přeposílat rámce.

Ještě poznámku k termínu. Neznámým rámcem unicast rozumíme, když přepínač nemá v tabulce MAC adres cílovou adresu MAC a musí předat rámec všem portům, kromě příchozího portu.



Broadcastová bouře – broadcast storm je abnormálně vysoký počet broadcastů, které v určitém čase zahltí síť. Bouře mohou během několika sekund deaktivovat síť zahlcením přepínačů a koncových zařízení. Může být způsobena problémem s hardwarem, například vadným síťovým adaptérem nebo smyčkou 2 vrstvy sítě.

Broadcasty na 2. vrstvě v síti, jako jsou například požadavky ARP, jsou velmi běžné. Vícesměrové vysílání 2 vrstvy jsou přepínačem obvykle předávány stejným způsobem jako broadcast. Naproti tomu pakety IPv6 nejsou nikdy předávány jako broadcast 2 vrstvy, ICMPv6 Neighbor Discovery používá na této vrstvě tzv. vícesměrové vysílání.

Host zachycený ve smyčce 2 vrstvy není přístupný jiným hostům v síti. Navíc kvůli neustálým změnám v tabulce MAC adres přepínač neví, na který port má předávat rámce jednosměrového (unicast) vysílání. Aby se předešlo výskytu těchto problémů v redundantní síti, musí být na přepínačích povolen nějaký typ spanning tree. Aby se preventivně vzniku smyček 2 vrstvy zabránilo, je na přepínačích Cisco Spanning Tree Protokol již ve výchozím nastavení povolen.

STP – Spanning tree protocol

STP je založený na algoritmu navrženém paní Radia Perlman, pracující v té době pro Digital Equipment Corporation. Byl publikován v článku "An Algorithm for Distributed Computation of a Spanning Tree in an Extended LAN" v roce 1985. Zmíněný spanning tree algoritmus (STA) vytvoří bezsmyčkovou topologii výběrem jednoho kořenového mostu, z něhož všechny ostatní přepínače určí jednu cestu s nejnižšími náklady. STP brání vzniku smyček konfigurací cesty bez smyčky v síti pomocí strategicky umístěných block-state portů tzv "blokujícího stavu". Přepínače provozující STP jsou schopny kompenzovat poruchy dynamickým odblokováním dříve zablokovaných portů a povolením provozu přes alternativní cesty.

Mechanismus jak STA vytváří bezsmyčkovou topologii spočívá v následujících krocích.

Nejdříve se provede výběr kořenového mostu: Přepínač s funkcí mostu je referenčním bodem pro celou síť, z něhož bude možné sestrojit kostru (spanning tree).

Dále dojde k blokování redundantních cest. STP zajišťuje, že mezi všemi cíli v síti existuje pouze jedna logická cesta záměrným blokováním redundantních cest, které by mohly způsobit smyčku. Když je port zablokován, uživatelská data nemohou vstoupit nebo opustit tento port.

Následkem toho je vytvoření bezsmyčkové topologie. Blokovaný port způsobí, že jím realizovaný spoj je nefunkčním spojením mezi dvěma přepínači. Vytvoří se tím topologie, kde každý přepínač má pouze jednu cestu ke kořenovému mostu, podobně jako větve na stromu, které se připojují ke kořenu stromu. V případě poruchy spojení se provede přepočet. Fyzické cesty, ač jsou ve snaze zabránit výskytu smyček zakázané, stále existují, aby poskytovaly v případě potřeby redundanci. Pokud je jimi realizovaná cesta někdy potřebná k nahradě selhání síťového kabelu nebo přepínače, STP přepočítá cesty a odblokuje nezbytné porty, aby umožnila aktivaci redundantní cesty. Přepočty STP mohou také nastat, kdykoli je do sítě přidán nový přepínač nebo je provedeno nové propojení mezi přepínači.

Poznámka: K tomuto je v materiálech Cisco akademie dostupné video, které ukazuje použití STP.

5.2 Fungování STP

V rámci interní funkce STP nalezneme kroky potřebné pro dosažení bezsmyčkové topologie.

Použitím spanning tree algoritmu STA a protokolu STP k ní dospějeme ve čtyřech krocích, které si dále probereme podrobně:

1. Volba kořenového mostu.
2. Volba kořenových portů.
3. Volba určených portů.
4. Volba alternativních (blokovaných) portů.

Během fungování STA a STP, přepínače používají v rámci komunikace tzv. Bridge Protocol Data Units (BPDU) a to ke sdílení informací o sobě a svých připojených. BPDU se používají k volbě kořenového mostu, kořenových portů, určených portů a alternativních portů. Každá BPDU obsahuje Bridge ID (BID) což by se dalo přeložit jako identifikátor mostu, jenž identifikuje, který přepínač tuto BPDU poslal. BID se podílí na provádění mnoha rozhodnutí algoritmu STA, včetně určování rolí kořenového mostu a portů. BID obsahuje hodnotu priority, MAC adresu přepínače a rozšířené ID systému. Nejnižší hodnota BID je určena kombinací těchto tří polí.

BPDU neboli Bridge Protocol Data Unit má tři hlavní části

položka	protocol ID	protocol version	BPDU type	flags	root BID	root path cost	sender BID	sender port ID	Message Age	Max Age	Hello Time	Forward Delay
velikost [B]	2	1	1	1	8	4	8	2	2	2	2	2

- Globální informace o STP (verze apod.) – bílý peostor
- Informace dané instance STP pro konfiguraci – modrý prostor
- Časové parametry (STP timers) - žlutý

Hello Time je interval, po kterém se zasílají BPDU (default 2s).

Max age (default 20s)

Forward delay (default 15s) jsou doby mezi stavami.

Priorita mostu: Výchozí hodnota priority pro všechny přepínače Cisco je dekadicky 32768. Rozsah hodnot priorit je 0 až 61440 v krocích po 4096. Výhodnější je nižší priorita mostu. Priorita mostu 0 má přednost před všemi ostatními prioritami.

Rozšířené ID systému: Hodnotou rozšířeného ID systému rozumíme dekadickou hodnotu přidanou k hodnotě priority mostu v BID k identifikaci VLAN pro tuto BPDU.

MAC adresa: Pokud jsou dva přepínače nakonfigurovány se stejnou prioritou a mají stejné rozšířené ID systému, bude mít nižší BID ten přepínač, jehož MAC adresa vyjádřená v hexadecimálním tvaru má nejnižší hodnotu.

Volba kořenového mostu

Algoritmus STA označí jeden přepínač jako kořenový most a používá jej jako referenční bod pro všechny výpočty cest. Přepínače si vyměňují BPDU za účelem vytvoření bezsmyčkové topologie počínaje výběrem kořenového mostu.

Všechny přepínače v broadcastové doméně se účastní volebního procesu. Přepínač po spuštění začne odesílat rámce BPDU každé dvě sekundy. Tyto rámce BPDU obsahují BID odesílajícího přepínače a BID kořenového mostu, známý jako kořenový ID.

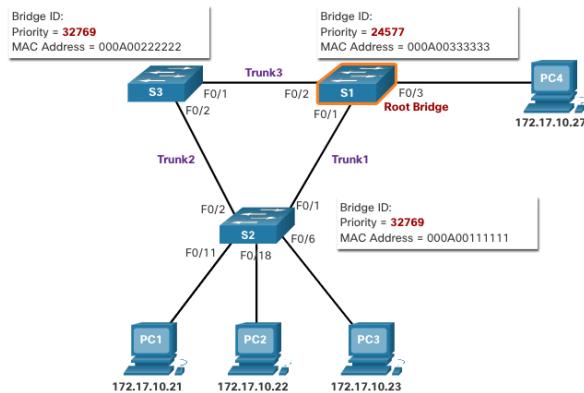
Přepínač s nejnižším BID se stane kořenovým mostem. Nejprve se všechny přepínače prohlásí za kořenové mosty s vlastním BID nastaveným jako kořenový ID. Prostřednictvím výměny BPDU se nakonec přepínače naučí, který z nich má nejnižší BID a shodnou se na jednom kořenovém mostě.

Dopad defaultních BID

Defaultní BID mají pochopitelně dopad na zafungování algoritmu potažmo protokolu STP.

Protože výchozí BID je 32768, je možné, aby dva nebo více přepínačů měly stejnou prioritu. V zobrazené topologii, kde jsou priority stejné, se přepínač s nejnižší adresou MAC stane kořenovým mostem. Správce by měl nakonfigurovat požadovaný přepínač s nižší prioritou, aby se stal kořenovým.

Na obrázku jsou všechny přepínače konfigurovány se stejnou prioritou 32769. MAC adresa se zde stává rozhodujícím faktorem pro volbu, který přepínač se stane kořenovým mostem. Přepínač s nejnižší hexadecimální hodnotou adresy MAC je upřednostňovaný kořenový most. V tomto příkladu má nejnižší hodnotu S2 vzhledem ke své MAC adrese a je zvolen kořenovým mostem pro danou instanci kostry (spanning tree).



Poznámka: Priorita všech uvedených přepínačů je 32769. Tato hodnota vychází ze zmíněné defaultní priority mostu 32768 a rozšířeného ID systému (přiřazení VLAN 1) přidruženého ke každému přepínači. Jednoduchým součtem obou hodnot 32768 a 1 dostaneme zde uvedenou 32769.

1. Stanovení ceny kořenové cesty

Po té co máme zvolený kořenový bridge, je potřeba provést stanovení ceny kořenové cesty.

STA začne určovat nejlepší cesty ke kořenovému mostu ze všech cílů v broadcastové doméně. Informace o cestě, známé jako cena vnitřní kořenové cesty, je určena součtem všech nákladů na jednotlivé porty podél cesty od daného přepínače po kořenový most. Když přepínač obdrží BPDU, přidá cenu vstupního portu segmentu, aby určil své náklady na vnitřní kořenovou cestu.

Defaultní náklady na port jsou definovány jeho rychlostí. Tabulka na snímku zobrazuje defaultní náklady na port navržené organizací IEEE. Přepínače Cisco používají defaultní hodnoty definované v normě IEEE 802.1D, známé také jako náklady na krátkou cestu, pro STP i RSTP. Přestože porty přepínačů mají přidružené defaultní náklady na port, lze je konfigurovat. Schopnost konfigurovat náklady na jednotlivé porty dává správci flexibilitu pro ruční ovládání cest ve spanning tree ke kořenovému mostu.

2. Volba kořenových portů

Poté, co byl určen kořenový most, se algoritmus STA použije k výběru kořenového portu. Každý přepínač, který není kořenový, si vybírá jeden kořenový port. Kořenový port je port nejbliže kořenovému mostu s ohledem na celkové náklady na kořenový most. Tato celková cena se označuje jako cena vnitřní kořenové cesty.

Náklady na vnitřní kořenovou cestu se rovnají součtu všech nákladů na porty podél cesty ke kořenovému mostu, jak je znázorněno na obrázku. Cesty s nejnižšími náklady budou upřednostněny a všechny ostatní redundantní cesty jsou blokovány. V příkladu je cena vnitřní kořenové cesty od S2 po kořenový most S1 po cestě 1 rovna 19, zatímco cena vnitřní kořenové cesty po cestě 2 je 38. Protože cesta 1 má nižší celkové náklady na cestu ke kořenovému mostu, je to preferovaná cesta a F0/1 se stane kořenovým portem na S2.

3. Volba určených portů

Ve volbě určených portů se setkáme s pojmy root port a designated poort, které si samozřejmě vysvětlíme.

Každý segment mezi dvěma přepínači bude mít jeden určený port. Určený port je takový port v segmentu, kterého cena je rovná nákladům na vnitřní kořenovou cestu ke kořenovému mostu. Jinými slovy, určený port je na nejlepší cestě pro příjem provozu vedoucího ke kořenovému mostu.

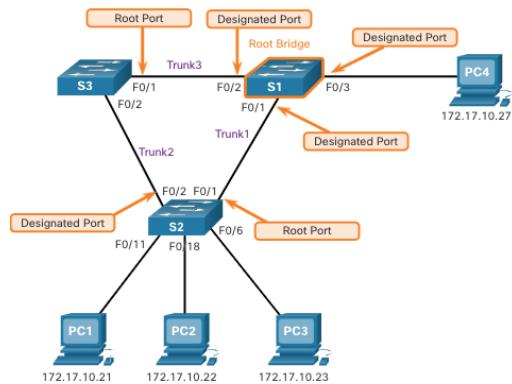
Port, jenž není kořenový nebo určený port, se stane alternativním nebo blokováným portem.

Všechny porty na kořenovém mostě jsou určené. Na obrázku jde o přepínač S1.

Pokud je jeden konec segmentu kořenový port, druhý konec je určený port. Např. na spoji označeném Trunk 3 je na S3 port F0/1 root portem a v souladu s předchozí informací je port f0/2 na S1 portem určeným.

Všechny porty připojené ke koncovým zařízením jsou určené porty. Tedy na S2 jde o porty 6, 11 a 18. Na kořenovém mostu S1 jde o port f0/3.

Na segmentech mezi dvěma přepínači, kde žádný z nich není kořenový most, je port na přepínači, jenž má cestu s nejnižšími náklady na kořenový most, určený port. Což je na obr. spoj mezi přepínači S2 a S3, kde jde o port f0/2 na S2.



4. Volba alternativních (blokováných) portů

Pokud port není kořenový port nebo určený port, stane se alternativním (nebo záložním) portem. Alternativní porty jsou ve stavu vyřazování nebo blokování, aby se zabránilo vzniku smyček. Na obrázku algoritmus STA nakonfiguroval port F0/2 na S3 do role alternativního portu. Port F0/2 na S3 je ve stavu blokování a nebude proto přenosit ethernetové rámce. Všechny ostatní porty nacházející se mezi přepínači, jsou ve stavu přenosu. Toto je ta součást STP, která zabrání vzniku smyček.

Volba kořenového portu z více cest se stejnou cenou

Pokud má totiž přepínač na výběr z více rovnocenných cest ke kořenovému mostu, určí port pomocí následujících kritérií:

- Nejnižší BID odesílatele
- Nejnižší priorita portu odesílatele
- Nejnižší ID portu odesílatele

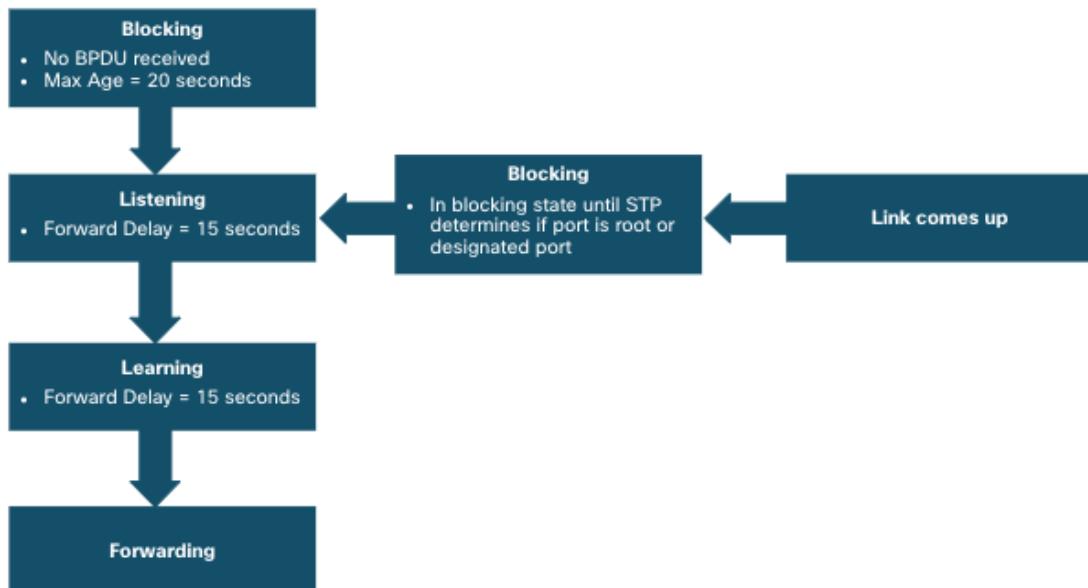
STP časovače a stavy portů

Konvergence STP vyžaduje pro svou správnou funkci následující tři časovače:

- **Hello časovač** - Hello je časový interval mezi zprávami BPDU. Výchozí hodnotou jsou 2 sekundy, ale lze ji upravit v rozmezí 1 až 10 sekund.
- **Časovač zpoždění přeposílání**. - Zpoždění přeposílání je čas strávený ve stavu poslechu a učení. Výchozí hodnota je 15 sekund, ale lze ji upravit na 4 až 30 sekund.
- **Časovač maximálního věku** - Maximální věk je maximální doba, po kterou přepínač čeká, než se pokusí změnit topologii STP. Výchozí hodnota je 20 sekund, ale lze ji upravit na 6 až 40 sekund.

Poznámka: Výchozí časy lze změnit pouze na kořenovém mostu, který určuje hodnotu těchto časovačů pro doménu STP.

Pro lepší pochopení funkce časovačů a s nimi svázanými stavy portů tu máme diagram, který nám zároveň objasňuje prodlevu aktivace portu po zapojení UTP kabelu.



STP umožňuje cestu bez logické smyčky v celé broadcastové doméně. Je určen na základě informací získaných výměnou rámců BPDU mezi propojenými přepínači. Pokud port přepínače přechází přímo ze stavu blokování do stavu přeposílání bez informací o úplné topologii během přechodu, může port dočasně vytvořit datovou smyčku. Z tohoto důvodu má STP pět stavů portů, z nichž čtyři jsou funkční stavů, jak je znázorněno na obrázku. Zakázaný stav je považován za nefunkční.

Při konvergenci (změně topologie, například připojení switche do sítě) prochází jednotlivé porty několika stavů. Mezi každým přechodem je určitý maximální časový interval

Jestliže je port blokovaný, pak pouze naslouchá provozu a přijímá/reaguje jen na BPDU, žádné datové rámce nepřesílá neboť není navázaný na MAC tabulkou. V případě naslouchání se stav změní jen v tom, že BPDU nejen naslouchá, ale též je odesílá. V okamžiku přechodu do režimu učení, již s MAC tabulkou vazbu má a je schopen ji upravovat, avšak, stále provoz nepřepíná. Stav, kdy je port plně funkční nastává až když je Forwarding tedy přesílájící.

Poznánka: Blokovaný Stav nezaměňuje za zakázaný. V tom totiž BPDU nepřijímá a není tedy schopen reagovat na případné změny v topologii. Zakázaný port je pro provoz mrtvý.

Tabulka shrnuje provozní podrobnosti každého stavu portů:

Stav portu	BPDU	Tabulka MAC adres	Přesílání datových rámci
Blokovaný	Pouze přijímá	Žádná aktualizace	Ne
Naslouchání	Přijímá a odesílá	Žádná aktualizace	Ne
Učení	Přijímá a odesílá	Aktualizace tabulky	Ne
Přesílájící	Přijímá a odesílá	Aktualizace tabulky	Ano
Zakázaný	Nepřijímá a nevysílá	Žádná aktualizace	Ne

Per-VLAN Spanning Tree

Za zmínku ještě stojí varianta integrovaná s VLANami. STP lze pochopitelně konfigurovat pro provoz v prostředí s více VLAN. Ne každá VLAN musí být propagována na všechny přepínače v organizaci. Ve verzích STP podle Per-VLAN Spanning Tree (PVST) je pro každou instanci spanning tree zvolen kořenový most. To umožňuje mít různé kořenové mosty pro různé sady VLAN. Jinými slovy, STP provozuje samostatnou instanci STP pro každou jednotlivou VLAN. Pokud jsou všechny porty na všech přepínačích členy VLAN 1, pak existuje pouze jedna instance spanning tree.

Počítačové sítě 2

5, EtherChannel a First Hop Redundancy Protokoly

Název bloku	Náplň
Funkce EtherChannel	Popis EtherChannel technology.
Konfigurace EtherChannel	Configure EtherChannel.
Ověření nastavení a řešení problémů s EtherChannel	Řešení problémů s EtherChannel.
First Hop Redundancy Protocols	Vysvětlení účelu a principu fungování First hop redundancy protocolů.
HSRP	Vysvětlení fungování HSRP.

6.1 EtherChannel – princip

EtherChannel povoluje agregace linek.

Existují scénáře, ve kterých je mezi zařízeními zapotřebí větší šířka pásma nebo redundancy, než kolik může poskytnout jediný fyzický spoj - linka. Ke zvýšení šířky pásma lze zařízení propojit více linkami. To je v rozporu s právě probraným Spanning Tree Protocolem (STP), který je ve výchozím nastavení povolen na zařízeních 2 vrstvy, jako jsou přepínače Cisco, který zablokuje redundantní směry, aby se zabránilo tvorbě smyček. Řešením je technologie agregace portů, která umožňuje redundantní propojení mezi zařízeními, která nebude blokována STP. Tato technologie je známá jako EtherChannel. EtherChannel je technologie agregace linek, která seskupuje více fyzických ethernetových spojů do jednoho logického spoje. Používá se k zajištění odolnosti proti chybám, rozložení zátěže, zvýšení šířky pásma a redundantaci spojů mezi přepínači, směrovači a servery.

Technologie EtherChannel umožňuje kombinovat počet fyzických spojů mezi přepínači a zvýšit tak celkovou rychlosť komunikace mezi přepínači.

Princip EtherChannel

Technologie EtherChannelu byla původně vyvinuta společností Cisco jako technika LAN switch-to-switch seskupení několika portů Fast Ethernet nebo Gigabit Ethernet do jednoho logického kanálu.

Konfigurací EtherChannel, svážeme fyzická rozhraní, čímž získáme virtuální rozhraní, které ozačujeme pojmem portový kanál (port channel).

Princip EtherChannelu lze pochopit i z brázku n snímku. Dvojice L3 switchů je propojena s porty access switche páry kabelů sduzenými do EtherChannelu. Sice přicházíme o dvojnásobek portů, získáváme ale agregované spoje s dvojnásobnou šířku pásma.

Výhody nastavení EtherChannel

Nastavení EtherChannelu je spojeno s řadou výhod. Většinu konfiguračních úkolů lze provádět na rozhraní EtherChannel místo na jednotlivých portech, což zajišťuje konzistenci konfigurace v rámci všech odkazů. EtherChannel spoléhá na stávající porty přepínačů. Není třeba upgradovat na rychlejší a dražší variantu spoje, abychom dosáhli větší šířku pásma.

Loadbalancing probíhá mezi linkami, které jsou součástí stejného EtherChannelu.

EtherChannel vytváří agregaci, která je považována za jeden logický odkaz. Pokud mezi dvěma přepínači existuje několik svazků EtherChannel, může STP blokovat jeden ze svazků, aby se zabránilo přepínání smyček. Když STP blokuje jeden z redundantních odkazů, blokuje celý EtherChannel. Tím se zablokují všechny porty patřící danému propojení EtherChannel. Tam, kde existuje pouze jeden odkaz EtherChannel, jsou aktivní všechny fyzické odkazy v EtherChannel, protože STP vidí pouze jeden (logický) odkaz. EtherChannel poskytuje redundanci, protože seskupené linky jsou považovány za jeden logický spoj. Ztráta jedné fyzické linky v kanálu neznamená změnu topologie.

Implementační omezení

S nasazením Etherchannelu se váží implementační omezení. Typy rozhraní nelze kombinovat. Např. v rámci jednoho EtherChannelu nelze kombinovat Fast Ethernet a Gigabit Ethernet porty. Každý EtherChannel může současně sestávat až z osmi kompatibilních ethernetových portů. EtherChannel

poskytuje plně duplexní šířku pásma až 800 Mbps pro Fast EtherChannel nebo 8 Gbps pro Gigabit EtherChannel vytvořený mezi přepínači nebo s jiným zařízením, např. serverem.

Limit pro přepínač Cisco Catalyst 2960 Layer 2 momentálně je, že podporuje maximálně šest EtherChannelů.

Individuální konfigurace portů začleněných do skupiny EtherChannel musí být na obou zařízeních konzistentní. Pokud jsou fyzické porty jedné strany nakonfigurovány jako trunk, musí být fyzické porty druhé strany také nakonfigurovány jako trunk, přičemž musejí mít shodně nastavenou nativní VLAN. Kromě toho musí být všechny porty v každém EtherChannel spojeny nakonfigurovány jako porty druhé vrstvy. Každý EtherChannel má tzv. logical port channel rozhraní. Konfigurací tohoto rozhraní se ovlivní všechna fyzická rozhraní, která jsou v tomto rozhraní zahrnuta.

Protokoly automatického vyjednávání

Pro správnou funkci využívá EtherChannel protokoly automatického vyjednávání. EtherChannels lze vytvořit pomocí jednoho ze dvou protokolů, Port Aggregation Protocol (PAgP) nebo Link Aggregation Control Protocol (LACP). Protokoly umožňují portům s podobnými charakteristikami vytvořit kanál prostřednictvím dynamického vyjednávání se sousedními přepínači.

Poznámka: Je možné vynechat automatické protokoly a nakonfigurovat statický nebo bezpodmínečný EtherChannel bez PAgP nebo LACP.

Fungování AgP

Jak již bylo naznačeno, je PAgP proprietárním protokolem společnosti Cisco, který pomáhá při automatickém vytváření EtherChannel linek. Když je propojení EtherChannel konfigurováno pomocí PAgP, pakety PAgP jsou odesílány mezi porty podporujícími EtherChannel, aby bylo možné vyjednat vytvoření kanálu. Když PAgP identifikuje spárované ethernetové odkazy, seskupí je do EtherChannelu. Ten je poté přidán do STP jako jeden port. Pokud je konfigurací povoleno, PAgP také spravuje EtherChannel. K tomu každých 30 sekund odesílá PAgP pakety. PAgP dále kontroluje konzistenci konfigurace a spravuje přidání či selhání linek mezi dvěma přepínači. Zajišťuje, že při vytvoření EtherChannel budou mít všechny porty stejný typ konfigurace.

Poznámka: V EtherChannelu je povinné, aby všechny porty měly stejnou rychlosť, duplexní nastavení a informace VLAN. Jakákoli úprava portu po vytvoření kanálu změní také všechny ostatní porty kanálu.

PAgP napomáhá vytvořit EtherChannel spoj s pomocí detekce konfigurace každé strany a zjištěním kompatibility linek. Pokud je vše v pořádku je pak, v případě potřeby, možné propojení EtherChannel realizovat.

PAgP pracuje v následujících módech:

- Zapnuto - Tento mód vynutí zapojení fyzického rozhraní do kanálu bez PAgP. Rozhraní nakonfigurovaná v tomto režimu si nevyměňují PAgP pakety.
- PAgP vyžadované - Tento režim PAgP převede fyz. rozhraní do stavu aktivního vyjednávání, ve kterém rozhraní iniciuje vyjednávání s jinými rozhraními zasláním PAgP paketů.

- PAgP auto - je režim, kdy PAgP převede rozhraní do stavu pasivního vyjednávání, ve kterém rozhraní reaguje na pakety PAgP, které přijímá, ale vyjednávání PAgP neinicializuje.

Režimy na obou stranách (switchích) musí být kompatibilní. Pokud je jedna strana nakonfigurována v automatickém režimu, je v pasivním stavu a čeká, až druhá strana zahájí EtherChannel vyjednávání. Pokud je druhá strana také nastavena na auto, vyjednávání nikdy nezačne a EtherChannel se nevytvoří. Pokud jsou všechny režimy deaktivovány pomocí příkazu no, nebo pokud není nakonfigurován žádný režim, pak je EtherChannel deaktivován. Zapnutý manuální režim převede rozhraní do EtherChannelu bez vyjednávání. Fungování je zajištěno pouze v případě, že je takto nastavena také druhá strana. Pokud je druhá strana nastavena na vyjednávání parametrů prostřednictvím PAgP, žádný EtherChannel se nevytvoří.

Fungování LACP

Fungování LACP je svým způsobem obdobné PAgP. LACP je součástí specifikace IEEE (802.3ad), která umožňuje spojit několik fyzických portů do jednoho logického kanálu. LACP umožňuje přepínače vyjednat automatické sdružení odesláním LACP paketů na druhý přepínač. Provádí funkci podobnou PAgP s Cisco EtherChannel. Protože LACP je standard IEEE, lze jej použít k vytvoření EtherChannelů v prostředích aktivních prvků od různých dodavatelů. Na zařízeních Cisco jsou podporovány oba protokoly. LACP poskytuje stejné vyjednávací výhody jako PAgP. LACP pomáhá vytvořit propojení EtherChannel detekcí konfigurace na obou propojených stranách a zaručením vzájemné kompatibility tak, aby bylo možné v případě potřeby propojení EtherChannel povolit.

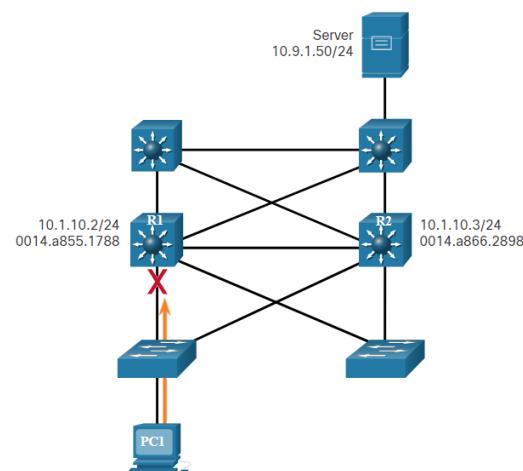
LACP má následující režimy:

- Zapnuto - vynutí zapojení fyzického rozhraní do kanálu bez LACP. Rozhraní nakonfigurovaná v tomto režimu si nevyměňují LACP pakety.
- Aktivní LACP - Tento režim LACP umístí port do aktivního stavu vyjednávání. V tomto stavu zahájí port vyjednávání s ostatními porty zasláním paketů LACP.
- Pasivní LACP - Tento režim LACP umístí port do stavu pasivního vyjednávání. V tomto stavu port reaguje na pakety LACP, které přijímá, ale neiniciuje vyjednávání paketů LACP.

5.2 First Hop Redundancy Protocols

Úzkým místem pro komunikaci koncových zařízení se zařízeními v jiné síti je výchozí brána. Koncová zařízení mají obvykle nakonfigurovanou jednu IPv4 adresu výchozí brány. Na snímku vidíme topologii s příkladem, kde sice redundandní prvek je k dispozici avšak FHRP není implementováno. Jinými slovy:

- Pokud na směrovači havaruje rozhraní definující default gateway vnitřní sítě, koncová zařízení ztratí konektivitu do vnějších sítí.
- K tomu dochází, i když existuje redundantní směrovač nebo přepínač L3 vrstvy, který by



mohl sloužit jako výchozí brána, protože o něm koncová zařízení z konfigurace nevědí a ani nemohou.

Jestliže to nelze řešit na úrovni koncových zařízení, musí být jiná cesta.

First hop redundancy protocols (FHRPs) jsou mechanismy, které poskytují alternativní výchozí brány v přepínaných sítích, kde jsou dva nebo více směrovačů připojeny ke stejným VLAN.

Redundance routerů

Jedním ze způsobů, jak eliminovat problém selhání jedinečného místa výchozí brány, je implementace virtuálního směrovače. Provádí se to tak, že je nakonfigurováno více směrovačů, aby společně představovali iluzi jednoho směrovače hostům v LAN. Sdílením IP adresy a MAC adresy mohou dva nebo více směrovačů fungovat jako jeden virtuální směrovač.

IPv4 adresa virtuálního směrovače je nakonfigurována jako default gateway na pracovních stanicích konkrétního IPv4 segmentu. Když jsou rámce odesílány z hostitelských zařízení na výchozí bránu, hostitelé používají dotaz na ARP k získání MAC adresy, která je přidružena k adrese IPv4 výchozí brány. ARP resolution jim vrátí MAC adresu virtuálního routeru. Rámce, které jsou odesílány na adresu MAC virtuálního směrovače, pak může fyzicky zpracovat aktuálně aktivní směrovač ze skupiny virtuálních směrovačů.

Protokol se používá k identifikaci dvou nebo více směrovačů coby zařízení, která jsou odpovědná za zpracování rámců odesílaných na MAC adresu nebo IP adresu jednoho virtuálního směrovače.

Koncová zařízení odesílají provoz na adresu virtuálního směrovače. Fyzický směrovač, který předává tento provoz, je pro koncová zařízení transparentní.

Protokol redundancy poskytuje mechanismus pro určení, který router by měl převzít aktivní roli při směrování provozu. Určuje také okamžik, kdy musí být role předávání převzata záložním směrovačem. Přechod z jednoho směrovače na druhý je pro koncová zařízení transparentní. Z principu přepínané sítě víme, že neřeší, kudy jsou pakety odesílány a jakou cestou k nim přicházejí. Schopnost sítě dynamicky se zotavovat po selhání zařízení fungujícího jako výchozí brána se nazývá redundancy prvního skoku (First Hop Redundancy).

Kroky pro převzetí služeb při selhání routeru

Převzetí služby směrování při selhání výchozího aktivního routeru je řešeno opět v krocích.

Tedy, když aktivní router selže, protokol redundancy změní záložní router na roli aktivního routeru, jak je znázorněno na obrázku. Proces je následující:

- Na směrovač v pohotovostním režimu (standby router) se přestanou doručovat zprávy Hello od aktivního směrovače.
- Standby router přebírá roli aktivního směrovače.
- Protože nový směrovač přebírá IPv4 i MAC adresu virtuálního směrovače, hostitelská zařízení nevidí žádné přerušení služby.

5.3 HSRP

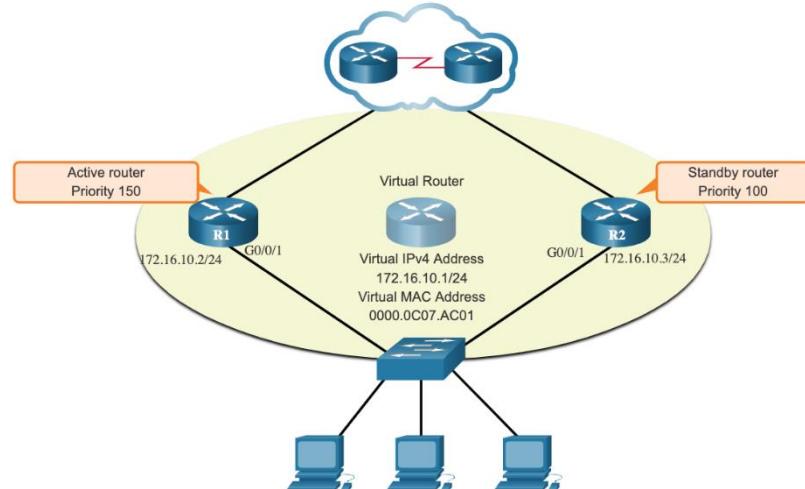
Společnost Cisco poskytuje HSRP a HSRP pro IPv6 jako způsob, jak zabránit ztrátě přístupu za hranice lokální sítě, pokud selže výchozí router. HSRP je Cisco proprietární FHRP řešení a je navrženo tak, aby umožňovalo transparentní převzetí služeb při selhání IP zařízení s prvním skokem.

HSRP zajišťuje vysokou dostupnost sítě poskytováním冗余 redundancy směrování prvního směrování pro IP hosty v sítích nakonfigurovaných s výchozí IP adresou brány. HSRP se používá ve skupině směrovačů pro výběr aktivního zařízení a pohotovostního zařízení. Ve skupině rozhraní zařízení je aktivním zařízení to, které se používá pro směrování paketů. Pohotovostním zařízením je pak zařízení, které převeze funkci při selhání aktivního zařízení nebo při splnění přednastavených podmínek. Funkcí záložního směrovače HSRP je sledovat provozní stav skupiny HSRP a rychle převzít odpovědnost za předávání paketů, pokud aktivní směrovač selže.

Priorita a předpoklad

Nyní k prioritám a předpokladům v HSRP. Role aktivního a pohotovostního směrovače se určuje během procesu volby HSRP. Ve výchozím nastavení je router s číselně nejvyšší adresou IPv4 zvolen jako aktivní router. Je však vždy lepší kontrolovat, jak bude vaše síť fungovat za normálních podmínek, než ji ponechat na náhodě.

- K určení aktivního směrovače lze použít prioritu HSRP a manuálně ji upravit.
- Směrovač s nejvyšší prioritou HSRP se stane aktivním směrovačem.
- Ve výchozím nastavení je HSRP priorita nastavena na hodnotu 100.
- Pokud jsou priority stejné, je jako aktivní router zvolen router s číselně nejvyšší adresou IPv4.
- Chcete-li nakonfigurovat směrovač jako aktivní směrovač, použijte příkaz rozhraní priority pohotovostního režimu. Rozsah priority HSRP je od 0 po 255.
- Na obrázku vidíme topologii, kde virtuální router má svou MAC i IP adresu a zahrnuje dva routery R1 a R2. Ty mají v konfiguraci nastavenou rozdílnou priority R1 150 a R2 výchozích 100.
- Na Routeru R1 je nastavena vyšší než na R2, proto je v aktivním stavu, což znamená, že směrování provozu jde přes jeho G0/0/1 rozhraní G0/0/1 s IP adresou 172.16.10.2. R2 sleduje intervaly Hello paketů a čeká na výpadek.



Priorita a předpoklad

Ve výchozím nastavení poté, co se směrovač stane aktivním směrovačem, zůstane aktivním směrovačem, i když se online připojí jiný směrovač s vyšší prioritou HSRP.

- Chcete-li vynutit nový proces voleb HSRP, když se router s vyšší prioritou připojí k internetu, musí být povolen „**předpoklad**“ příkazem rozhraní **standby preempt**. Preemption (předpoklad) je schopnost routeru HSRP spustit proces znovuzvolení. Se zapnutou předvolbou převeze roli aktivního online směrovače router s vyšší prioritou HSRP.

- Preemption umožňuje routeru, aby se stal aktivním routerem, pouze pokud má vyšší prioritu. Směrovač s povolenou předvolbou, se stejnou prioritou, ale vyšší IPv4 adresou, nepřevezme roli aktivnímu směrovači. Viz topologie na obrázku.

Poznámka: S deaktivovanou preempt předvolbou se router, který se spustí jako první, stane aktivním routerem, pokud během volebního procesu nejsou online žádné další routery.

Stavy a časování

Aktivní a pohotovostní směrovače HSRP ve výchozím nastavení odesílají hello pakety na multicast adresu HSRP skupiny každé 3 sekundy. Pohotovostní směrovač bude aktivní, pokud po 10 sekundách od aktivního směrovače neobdrží Hello zprávu. Tato nastavení časovače lze snížit a urychlit převzetí služeb při selhání nebo jako preventivní opatření. Aby se však zabránilo zvýšenému využití procesoru a zbytečným změnám stavu v pohotovostním režimu, nenastavujte hello časovač pod 1 sekundu nebo časovač hold timer pod 4 sekundy.

Stavy v nichž se může zařízení nacházet jsou Initial, Learn, Listen, Speak a Standby.

Stavem **Initial** označujeme stav, do kterého zařízení vstoupí prostřednictvím změny konfigurace nebo při prvním zpřístupnění rozhraní.

Learn je stav v němž směrovač nejistil virtuální IP adresu a dosud nezaregistroval Hello zprávu od aktivního směrovače. V tomto stavu router čeká, až se ozve aktivní router.

Listen je stavem naslouchání. Směrovač zná virtuální adresu IP, ale není ani aktivním ani záložním směrovačem. Poslouchá Hello zprávy od ostatních routerů.

Speak je stav, kdy směrovač odesílá pravidelné Hello zprávy a aktivně se účastní volby aktivního nebo pohotovostního směrovače.

Standby znamená, že směrovač je kandidátem na to stát se následujícím aktivním směrovačem a odesílá pravidelné Hello zprávy.

Počítačové sítě 2

6, DHCPv4

Název bloku	Náplň
DHCP4 koncept	Vysvětlení funkce DHCPv4 ve firemních sítích malé a střední velikosti.
Konfigurace Cisco IOS DHCP4 Server	Konfigurace Routeru jako DHCPv4 serveru.
Konfigurace DHCP4 klienta	Konfigurace Routeru jako DHCPv4 klienta.

Koncepty DHCPv4

Potřeba bezobslužné automatické konfigurace koncových sítových zařízení je svázána již s počátky datových sítí. První implementací je Bootstrap Protocol (zkratka BOOTP), což je sítový protokol, který byl definován v RFC 951 z roku 1985. Používal se pro nastavení parametrů pro stanice používající TCP/IP. V současné době je plně nahrazen protokolem DHCP, který je s ním zpětně kompatibilní.

Protokol BOOTP umožňoval používání bezdiskových stanic unixových systémů, které pro svůj start a následný běh mohly získat všechny potřebné údaje ze sítě. Iniciační kód stanic byl uložen buď na disketu nebo v BootROM sítové karty. Počítače po zapnutí díky němu zjistily z BOOTP serveru kromě své IP konfigurace také adresu serveru, na kterém byl uložen obraz pro zavedení operačního systému. Obraz byl následně stanicí stažen pomocí TFTP nebo FTP protokolu.

BOOTP je server – client technologie, kde si server uchovává ve své databázi seznam MAC adres sítových rozhraní stanic. Na žádost stanic jim odesílá uložené údaje, k čemuž využívá UDP protokol třetí vrstvy.

Vznik nástupce protokolu BOOTP, nynějšího Standardu DHCP, se pak váže k říjnu 1993, kdy bylo vydáno RFC 1531 obsahující jeho specifikaci.

DHCPv4 Server – klient

Dynamic Host Configuration Protocol v4 (DHCPv4) přiřazuje dynamicky IPv4 adresy a další informace o konfiguraci sítě. Protože klienti stolních počítačů obvykle tvoří většinu sítových uzlů, je DHCPv4 pro správce sítě velmi užitečným a časově úsporným nástrojem.

Dedikovaný DHCPv4 server je škálovatelný a jeho správa je poměrně snadná. V malé pobočce nebo umístění SOHO však lze směrovač Cisco nakonfigurovat tak, aby poskytoval DHCPv4 služby bez nutnosti nasazení dedikovaného serveru. Software Cisco IOS podporuje volitelný, plně vybavený DHCPv4 server.

DHCPv4 server dynamicky přiřazuje nebo zapůjčuje IPv4 adresu z **fondu adres** na omezenou dobu definovanou serverem nebo do té doby, než klient již adresu nebude potřebovat.

Klienti si pronajímají informace ze serveru na administrativně definované období. Správci konfigurují DHCPv4 servery tak, aby nastavili časový limit pronájmu v různých intervalech. Pronájem je obvykle v rozmezí od 24 hodin do týdne, výjimečně i déle. Po vypršení platnosti pronájmu musí klient požádat o adresu znova, obvykle je mu přiřazena stejná adresa.

Roli DHCPv4 serveru může mimo sítových prvků plnit i služba spuštěná na počítačích a serverech připojených do datové sítě.

Na LINUX se DHCP server, lépe řečeno daemon služby, se instaluje jako balíček operačního systému. Konfigurace se jako u všech služeb provádí v souboru.

Microsoft Windows Server nabízí DHCP server jako službu integrovanou v Active Directory, resp s DNS serverem, což je další služba v AD integrovaná. Pro přidělení důvěryhodných adres koncovým stanicím se DHCP server ve struktuře domény musí aktivovat jako důvěryhodný. Při přidělování IP adresy klientovi se předávají i další informace související s AD.

Microsoft Windows klientský OS má také službu DHCP. Je dostupná pro pracovní stanice, které mají dvě a více sítových rozhraní. Sítové připojení je možné na klientském OS MS Windows sdílet a to přes konfiguraci na sítové kartě.

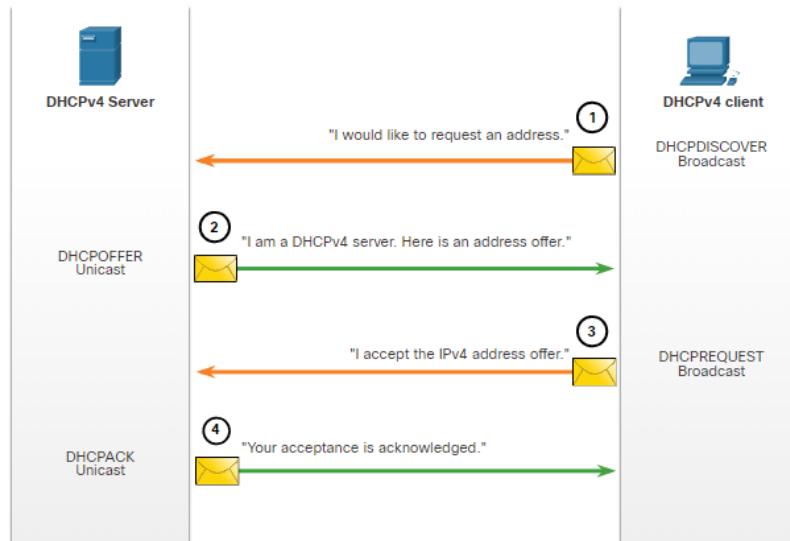
Průběh DHCPv4

Jak bylo naznačeno, DHCP funguje v režimu klient / server. Když klient komunikuje s DHCP v4 serverem, přiřadí tomuto klientovi IPv4 adresu. Klient se připojuje k síti s takto pronajatou IPv4 adresou, dokud nevyprší platnost pronájmu. Klient musí pravidelně kontaktovat server DHCP, aby si prodloužil zápůjčku. Mechanismus zapůjčení zajišťuje, že klienti, kteří se přesunou nebo vypnou, neuchovávají adresy, které již nepotřebují. Po vypršení platnosti pronájmu vrátí server DHCP adresu do fondu, odkud ji lze podle potřeby znova přidělit.

Kroky k získání IPv4 adresy pronájmem

Když klient zavádí OS (nebo se jinak chce připojit k síti), zahájí se proces o čtyřech krocích k získání adresy. Pro lepší pochopení je zde obrázek.

1. Klient neví, kdo je DHCP server, proto odesílá broadcastem DHCP Discover (DHCPDISCOVER), který může přijmou i několik DHCP serverů.
2. DHCP server odpovídá v druhém kroku reaguje a unicastem odesílá na klienta DHCP Offer (DHCPOFFER), tedy nabídku.
3. Klient následně zašle serveru DHCP Request (DHCPREQUEST) v němž akceptuje nabídku. Opět tak činí broadcastem, pro případ, že by v síti byl další DHCP server a čekal na odpověď.
4. V posledním kroku DHCP server unicastem klientovi posílá Acknowledgment (DHCPACK), čímž transakci potvrzuje.



Kroky obnovení pronájmu

Před vypršením platnosti pronájmu klient zahájí dvoustupňový proces obnovy zapůjčení se serverem DHCPv4.

Před vypršením platnosti pronájmu klient odešle zprávu DHCPREQUEST unicastem přímo na DHCPv4 server, který mu původně nabídl IPv4 adresu. Pokud není DHCPACK přijat ve stanoveném čase, klient vysílá další DHCPREQUEST, tentokrát broadcastem, aby získal pronájem od jiného DHCPv4 serveru.

Ve druhém kroku, po přijetí zprávy DHCPREQUEST server ověří informace o zapůjčení vrácením DHCPACK.

Zprávy jsou odesílány v souladu IETF RFC 2131 buď jako broadcast nebo unicast.

Konfigurace DHCPv4 Serveru v Cisco IOS

Funkci DHCP serveru může plnit jednak dedikovaný server, ale také třeba směrovač. Takto lze konfigurovat i Cisco prvky. Přiřazují a spravují IPv4 adresy ze zadaných fondů adres v routeru DHCPv4 klientům. Z obrázku je patrné možné nasazení služby.

Nakonfigurovat server Cisco IOS DHCPv4 lze ve třech krocích.

Krok 1. Vyčleňte IPv4 adresy. Jednu adresu nebo rozsah adres lze vyčlenit zadáním *low-address* a *high-address* z dostupného rozsahu adres. Vyčleněnými adresami by měly být ty adresy, které jsou přiřazeny směrovačům, serverům, tiskárnám a dalším zařízením, která byla nebo budou ručně nakonfigurována. Příkaz můžete také zadat několikrát. Příkaz je **ip dhcp excluded-address *low-address [high-address]***

Krok 2. Definujte jméno pro DHCPv4 pool. Příkaz **ip dhcp pool *pool-name*** vytvoří pool se specifickým jménem a převede směrovač do DHCPv4 konfiguračního režimu, který je identifikovaný promptem **Router(dhcp-config)#**.

Krok 3. Nakonfigurujte DHCPv4 pool. Musí být nakonfigurován fond adres, ze kterého se mají adresy přidělovat a směrovač výchozí brány. Užijte **network** parametr k definici rozsahu dostupných adres. Příkaz **default-router** použijte k definici výchozí brány.

Konfigurací dalších parametrů lze rozšířit množství předaných informací klientovi. V tabulce je výpis nejpoužívanějších z nich:

Úlohy	IOS příkazy
Definujte fond adres.	network network-number[mask /prefix-length]
Definujte výchozí router nebo bránu.	default-router address [address2...address8]
Definujte DNS server.	dns-server address[address2...address8]
Definujte jméno domény.	domain-name domain
Definujte dobu trvání pronájmu z DHCP.	lease {days [hours [minutes]] infinite}
Definujte server NetBIOS WINS.	netbios-name-server address[address2...address8]

Ověření nastavení

Pro ověření provedených nastavení lze provést následující příkazy:

Command	Description
show running-config section dhcp	Zobrazí příkazy DHCPv4 nakonfigurované na routeru.
show ip dhcp binding	Zobrazí seznam všech vazeb IPv4 adresy na MAC adresy poskytnuté službou DHCPv4.
show ip dhcp server statistics	Zobrazí informace týkající se počtu zpráv, které byly odeslány a přijaty DHCPv4 službou.

Ověření funkčnosti DHCPv4

Příkaz pro ověření provedené konfigurace je **show running-config | section dhcp**. Zobrazuje DHCP příkazy použité na R1 ke konfiguraci. Parametr **| section** zobrazí pouze příkazy asociované s konfigurací služby DHCP.

Druhým příkazem, kterým lze ověřit provoz DHCPv4 je **show ip dhcp binding**. Výstupem je seznam všech vazeb mezi IPv4 adresami a MAC adresami, které poskytla služba DHCP.

Výstupem třetího příkazu **show ip dhcp server statistics** je ověření, že zprávy jsou směrovačem přijaty i vyslány. Tento příkaz zobrazuje informace týkající se počtu těchto zpráv.

Ukončení funkce DHCPv4 Serveru

Služba DHCP je ve výchozím nastavení na prvcích povolena. K zakázání služby slouží příkaz **no service dhcp** v globálním konfiguračním modu. Použitím příkazu **service dhcp** v globálním konfiguračním modu lze službu znova povolit, viz příklad. Povolení služby však nemá efekt, pokud nejsou nastaveny její parametry.

Poznámka: Dojde-li k vymazání DHCP vazeb nebo zastavení a restart DHCP služby, může to mít za následek dočasné přiřazení duplicitních IP adres v síti.

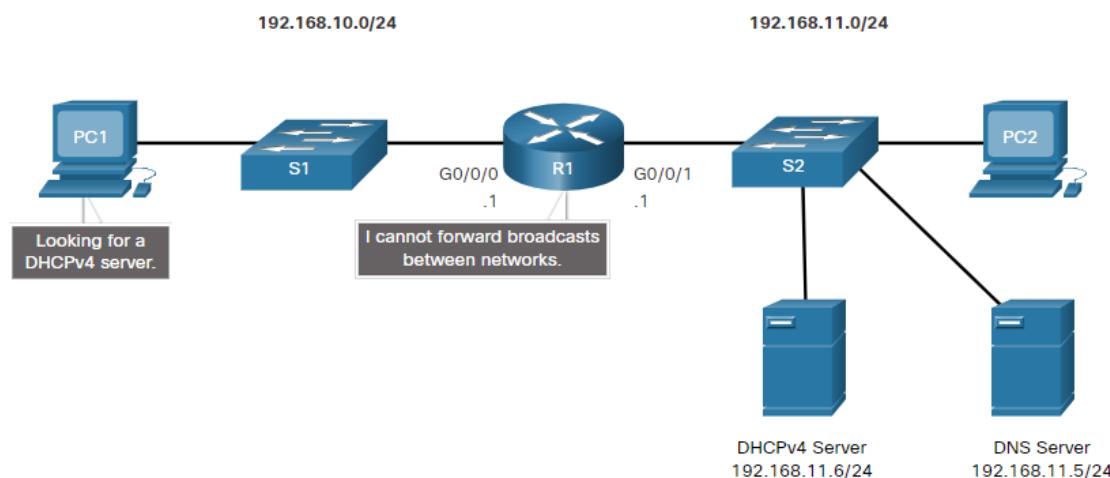
DHCPv4 Relay

Pro rozsáhlější síť, kde je více segmentů LAN, lze využít centrální DHCP prvek

Ve složité síti, kde je více segmentů LAN, jsou podnikové servery obvykle umístěny centrálně. Tyto servery mohou pro síť poskytovat služby DHCP, DNS, TFTP a FTP. Síťoví klienti obvykle nejsou s nimi ve stejném podsíti. Jak bylo uvedeno, za účelem vyhledání serverů a přijímání služeb klienti často používají broadcast.

Na obrázku se PC1 pokouší získat adresu IPv4 ze serveru DHCPv4 pomocí broadcastu. V tomto scénáři R1 není nakonfigurován jako server DHCP a ani broadcastové zprávy nepředává.

Protože je server DHCPv4 umístěn v jiné síti, nemůže PC1 přijímat IP adresu pomocí DHCP. R1 musí být nakonfigurován tak, aby přenášel zprávy DHCPv4 na server DHCPv4. Tomuto nastavení říkáme DHCP relay.



DHCP relay role se nakonfiguruje na R1 v našem případě pomocí konfiguračního příkazu **ip helper-address**. To způsobí, že R1 bude přenášet vysílání DHCPv4 na DHCPv4 server. Jak je znázorněno v příkladu, rozhraní na R1 přijímající vysílání z PC1 je nakonfigurováno tak, aby přenášelo DHCPv4 požadavky na DHCPv4 server na 192.168.11.6.

Když byl R1 nakonfigurován jako DHCPv4 relay agent, přijímá broadcast požadavky pro službu DHCPv4 a poté je předává jako unicast na adresu IPv4 192.168.11.6.

Příkaz **show ip interface** lze použít k ověření konfigurace.

Cisco Router jako DHCPv4 klient

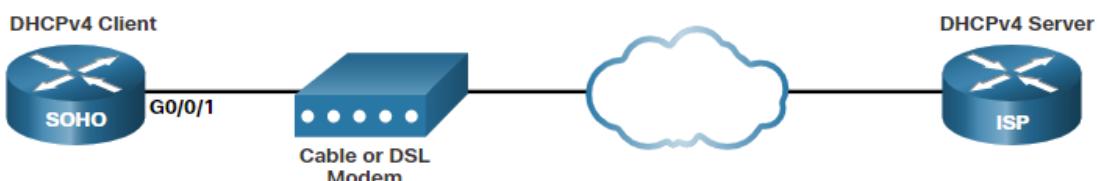
Co se týče DHCP klienta v OS MS Windows, tak jde o výchozí nastavení, proto není třeba nic konfigurovat.

V případě Cisco zařízení s OS IOS je třeba k získání IP adresy z DHCP serveru provést konfigurační kroky. Existují scénáře, kdy můžete mít přístup do datové sítě internet service providera (ISP) prostřednictvím dynamicky přidělené adresy z jeho DHCP serveru.

Napojení na ISP se vždy řídí jeho metodikou a ta může zahrnovat připojení s využitím DHCP na předdefinovanou MAC adresu.

V těchto případech je třeba nakonfigurovat vstupní rozhraní Cisco směrovače jako DHCP klienta.

- Především se jedná o Směrovače Cisco v malé kanceláři nebo domácí kanceláři (SOHO) a na pobočkách. Použitá metoda závisí plně na ISP. V nejběžnější konfiguraci se rozhraní Ethernet používá k připojení ke kabelovému nebo DSL modemu, což je fyzické rozhraní mezi sítěmi.
- Ke konfiguraci Ethernet rozhraní jako DHCP klienta se použije příkaz **ip address dhcp interface**.
- V situaci na obrázku předpokládáme, že ISP byl nakonfigurován tak, aby poskytoval vybraným zákazníkům IP adresy ze síťového rozsahu 209.165.201.0/27. K tomu je třeba nakonfigurovat rozhraní G0 / 0/1 pomocí patřičného příkazu.



Počítačové sítě 2

7, SLAAC a DHCPv6

Název bloku	Obsah
Přidělení IPv6 Global Unicast Adres	Vysvětlení jak IPv6 host může získat svou IPv6 konfiguraci.
SLAAC	Vysvětlení fungování SLAAC.
DHCPv6	Vysvětlení fungování DHCPv6
Konfigurace DHCPv6 Serveru	Konfigurace stavového a bezstavového DHCPv6 serveru.

IPv6 Konfigurace klienta

Na Routeru se IPv6 global unicast adresa (GUA) nastavuje ručně při konfiguraci síťového rozhraní příkazem **ipv6 address** s parametrem *ipv6 adresy a jejího prefixu*

Při konfiguraci klienta OS Windows, lze IPv6 adresu manuálně zadat ve vlastnostech rozhraní.

Nutno podotknout, že manuální vkládání IPv6 adres může být jednak časově náročné a navíc může při něm dojít k chybnému zadání. Koncová zařízení s MS Windows mají ve výchozím nastavení povolené dynamické získání IPv6 GUA konfigurace.

IPv6 Link-Local Adresa (LLA)

Služba DHCPv6 je definovaná v RFC 3315. Pro implementaci IPv6 je nezbytné, aby uzly měly nakonfigurované link-local adresy. Pokud je vybráno automatické adresování IPv6, hostitel použije Router Advertising (RA) zprávu ICMPv6 protokolu, která mu pomůže IPv6 adresu nakonfigurovat automaticky.

```
C:\> ipconfig
Windows IP Configuration
Ethernet adapter Ethernet0:
  Connection-specific DNS Suffix  . :
  IPv6 Address. . . . . : fe80::fb:1d54:839f:f595%21
  Link-local IPv6 Address . . . . . : fe80::fb:1d54:839f:f595%21
  IPv4 Address. . . . . : 169.254.202.140
  Subnet Mask . . . . . : 255.255.0.0
  Default Gateway . . . . . :
C:\>
```

Jak je vidět na výstupu na obrázku, pokud je při zavádění operačního systému Ethernetové rozhraní aktivní, **IPv6 link-local adresa** se hostem vytváří automaticky. Jestliže síťový segment nemá směrovač, který by poskytoval konfiguraci sítě, rozhraní nevytváří IPv6 GUA.

Poznámka: Kombinaci znaku "%" a čísla na konci link-local adresy říkáme Zone ID nebo Scope ID. Operační systém ho používá k asociaci LLA s konkrétním síťovým rozhraním.

Získání IPv6 GUA

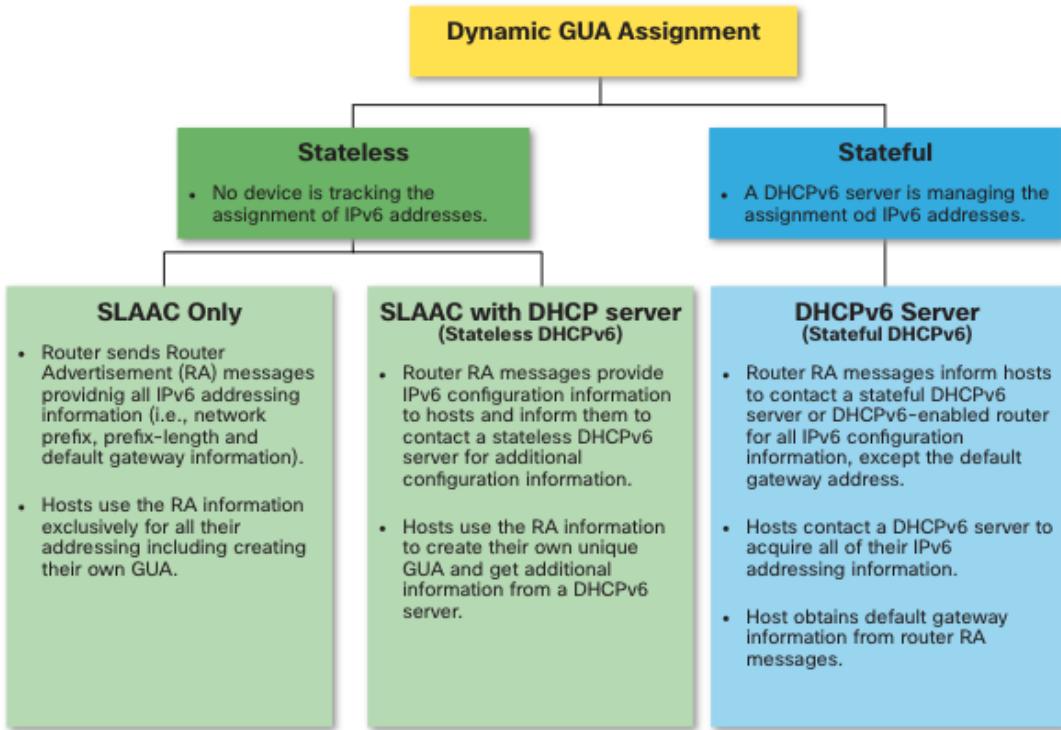
Pro získání IPv6 GUA máme několik možností. Ve výchozím nastavení směrovač s aktivním IPv6 protokolem pravidelně posílá ICMPv6 RA zprávy, což zjednodušuje hostu proces dynamického vytváření nebo získání IPv6 konfigurace. Host může GUA konfiguraci získat dynamicky použitím stavové nebo bezstavové služby GUA. Všechny stavové i bezstavové služby uvedené na schématu používají Router Advert ICMPv6 zprávy k ovlivnění tvorby či získání IPv6 konfigurace.

Základním rysem bezstavové konfigurace je, že žádné zařízení v dané síti nesleduje přidělování IPv6 adres. Můžeme použít dva mechanismy, SLACC a SLAAC s bezstavovým DHCP serverem.

Čistý **SLAAC**, kdy router posílá do segmentu RA zprávy a poskytuje veškeré potřebné informace (např. prefix sítě a výchozí brána) nebo **SLAAC s bezstavovým DHCP serverem**, který také distribuuje RA zprávy, v nich navíc odkazuje na bezstavový DHCPv6 server, pro získání doplňujících informací. Hosté pak používají RA zprávy k vytvoření unikátních globálních adres, z DHCP si doplňují zbývající

informace. Oproti bezstavové je stavová konfigurace podobná již nám známé IPv4 DHCP službě, která má adresní prostor, v mezích možnosti, pod kontrolou.

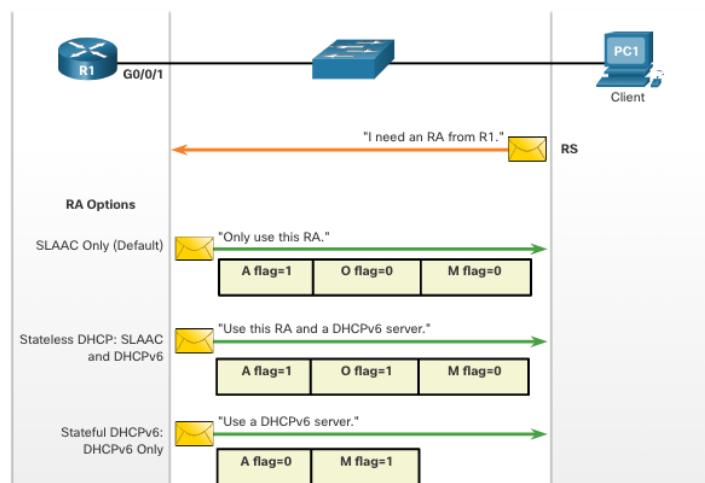
Ačkoli hostitelské operační systémy používají návrh z RA, skutečné rozhodnutí je nakonec na nich.



Trojice příznaků RA zpráv

To, jak klient získá GUV IPv6, závisí na nastavení ve zprávě RA. Zpráva ICMPv6 RA obsahuje následující tři příznaky (A, O, M):

- A flag** - Příznak automatické konfigurace adresy znamená, že k vytvoření GUV IPv6 bude použita bezstavová automatická konfigurace adresy (SLAAC).
- O flag** - Příznak Další konfigurace znamená, že doplňující informace jsou dostupné na bezstavovém DHCIPv6 serveru.
- M flag** - Příznak Konfigurace spravované adresy (Managed Address Configuration) znamená použití stavového serveru DHCIPv6 k získání GUA IPv6.



Pomocí různých kombinací příznaků A, O a M informují RA zprávy hosty o dostupných dynamických možnostech.

SLAAC

EUI – 64

Pro stateless (bezstavovou) autokonfiguraci se používá metoda „Extended Universal Identifier – 64“.

Předpokladem jejího použití je počáteční získání 64-bitového prefixu a to buď z Routeru pomocí NDP, nebo statickou konfigurací prefixu. Pro link-local adresy je prefix **FE80::/10**, zbývajících 54bitů tvoří ve výchozím stavu zpravidla nuly.

64 bitů druhé poloviny adresy je dopočítáno z MAC adresy síťového rozhraní tak, že se mezi její části „**OUI**“ a „**S/N**“ vloží dvojbajt **FF:FE**.

Názorně je možné vidět na příkladu link local adresy:

Prefix: **FE80::/10**

MAC: **0013:D4A5:1D60**

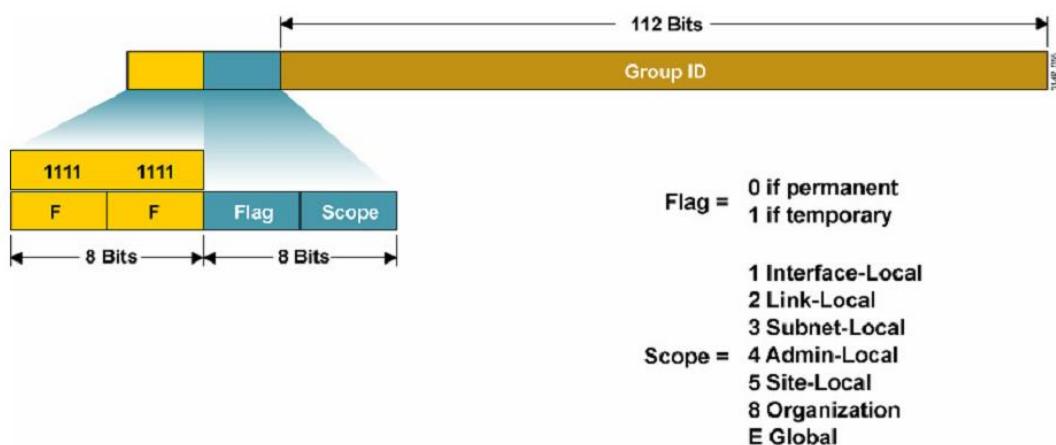
Výsledná link-local IPv6: **FE80:0000:0000:0000:0013:D4FF:FEA5:1D60**

Základní tabulka převodů např. zde <http://easycalculation.com/hex-converter.php>

Multicastové IPv6 adresy

Multicastové IPv6 adresy se v IPv6 používají často, de facto nahradily IPv4 broadcasty. Sama tato adresa je kódem, který v prvním šestnáctibitovém bloku informuje příjemce o svém účelu či obsahu.

První oktet tvoří binární jedničky, v hexa kódu jde o dvojici FF, kterou adresa začíná. Následujících osm bitů je rozdělený na část Flag a Scope. Jejich význam je v legendě obrázku.



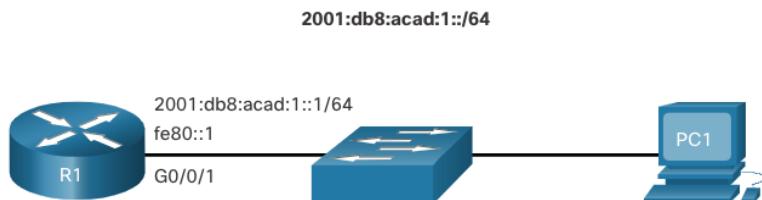
Základ SLAAC

Ne každá síť má přístup k serveru DHCP pro protokol IPv6, ale každé zařízení v síti s alIPv6 adresací potřebuje Globální Unicast Adresu. Stateless address autoconfiguration metoda, neboli SLAAC, umožňuje hostům si vytvořit vlastní unikátní IPv6 global unicast adresu bez služeb DHCPv6 serveru.

Bezstavovost znamená, že v dané síti neexistuje žádný server služby, který by udržoval informace o přidělených a aktivních síťových IPv6 adresách. SLAAC periodicky, každých 200 sekund, zasílá ICMPv6 RA zprávy, které hostům poskytují informaci k adresování a další konfigurační parametry v mechanismu automatické konfigurace. Host mimo to může poslat Router Solicitation (RS) zprávu a RA zprávu si vyžádat. SLAAC může být implementovaný jednak samostatně nebo v kombinaci s DHCPv6 serverem.

Nastavení SLAAC

Nastavení služby SLAAC není složité a vysvětlíme si ho na příkladu uvedeném na obrázku topologie a výpisu.



Na routeru R1 bylo rozhraní G0/0/1 bylo nakonfigurované se zobrazenou Global Unicast Adresou a link-local adresou. Tedy, rozhraní zahrnuje IPv6 adresy:

- **Link-local IPv6 address** - fe80::1
- **GUA s prefixem** - 2001:db8:acad:1::1, 2001:db8:acad:1::/64
- **Multicastovou IPv6 adresu pro all-nodes group** - ff02::1

```
R1# show ipv6 interface G0/0/1
GigabitEthernet0/0/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::1
  No Virtual link-local address(es):
  Description: Link to LAN
  Global unicast address(es):
    2001:DB8:ACAD:1::1, subnet is 2001:DB8:ACAD:1::/64
  Joined group address(es):
    FF02::1
    FF02::1:FF00:1
  (output omitted)
R1#
```

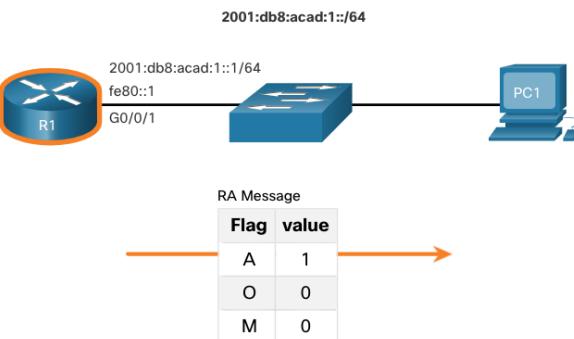
Můžeme říci, že router R1 je nakonfigurován tak, aby se připojil ke všem IPv6 multicast skupinám a posílal RA zprávy obsahující informace o konfiguraci adres hostům pomocí SLAAC.

```
R1(config)# ipv6 unicast-routing
R1(config)# exit
R1#
```

IPv6 all-routers skupina odpovídá na IPv6 multicast adresu ff02 :: 2. Příkazem **show ipv6 interface** lze ověřit, že se R1 připojil k all-routers IPv6 skupině (tj. má adresu ff02::2). R1 posílá RA zprávy každých 200 vteřin na IPv6 all-nodes multicast adresu ff02::1.

RA zprávy z R1 mají následující sadu příznaků:

- A = 1** – Informuje v RA zprávách klienta, aby použil GUI IPv6 předponu a dynamicky vytvořil své vlastní Interface ID.
- O = 0 a M = 0** – Informuje v RA zprávách klienta, aby použil doplňující informace z RA zpráv (DNS server, MTU, and default gateway).



Příkaz **ipconfig** v OS Windows potvrzuje, že PC1 vygeneroval IPv6 GUS s využitím RA zpráv z R1. Adresa výchozí brány je Link Local Adresa rozhraní G0/0/1 routeru R1.

```
C:\> ipconfig
Windows IP Configuration
Ethernet adapter Ethernet0:
  Connection-specific DNS Suffix . :
  IPv6 Address . . . . . : 2001:db8:acad:1:1de9:c69:73ee:ca8c
  Link-local IPv6 Address . . . . : fe80::fb:1d54:839f:f595%21
  IPv4 Address . . . . . : 169.254.202.140
  Subnet Mask . . . . . : 255.255.0.0
  Default Gateway . . . . . : fe80::1%6
C:\>
```

ICMPv6 - RS zprávy

Router rozesílá RA zprávy každých 200 vteřin, nebo když přijme RS zprávu od hostů. Když host s aktivním IPv6 adresováním potřebuje přijmout potřebné informace, pošle RS zprávu na IPv6 all-routers multicast adresu **ff02::2**.

Když PC1 načte OS, zašle RS zprávu na IPv6 all-routers multicast address of ff02::2 s požadavkem na zaslání RA.

R1 vygeneruje a odešle RA zprávu na IPv6 all-nodes multicast adresu ff02::1.

PC1 použije tuto informaci k vytvoření unikátní IPv6 Glob Unic Adr.

Proces hosta při generování ID rozhraní probíhá pomocí SLAAC. Díky němu host získá z RA od Routeru 64 bitovou informaci o své IPv6 podsíti. Zbývá vygenerovat zbývajících 64 bitů identifikátoru rozhraní (ID), k čemuž máme k dispozici dvě metody. Jde buď o náhodné generování nebo EUI-64

- Náhodné generování** 64 bitů pro Interface ID je realizováno **operačním systémem hosta**. Této metody využívá OS Windows 10. Jde o deklarovaný bezpečnější mechanismus než EUI-64.
- EUI-64** - host vytváří Interface ID za pomocí 48 bitů MAC adresy, přičemž do jejího středu vloží hexa hodnotu fffe. Některé operační systémy využívají metodu náhodného generování z důvodu obavy o soukromí. Je to proto, že Ethernetová MAC adresa hostitele je jednoznačným identifikátorem hosta a tedy i jeho majitele.

Zmíněným bezpečnostním faktorem je, že EUI-64 používá část jedinečné MAC adresy zařízení a tím je přesun zařízení napříč sítěmi snadno identifikovatelné, což je na hraně s ochranou osobních údajů. Proto Windows, Linux i Mac OS uživatelům dovolují zvolit si metodu generování Interface ID.

Detekce duplicitních adres

Potenciálním problémem jsou, přes množství adres, které lze vygenerovat, duplicitní adresy. Host používající SLAAC může pro kontrolu využít proces detekce duplicitních adres (DAD). Ten funguje následovně.

Host odešle ICMPv6 Neighbor Solicitation (NS) zprávu se speciálně konstruovanou solicited-node multicast adresou, která obsahuje posledních 24 bitů adresy IPv6 hosta. Pokud žádné jiné zařízení neodpoví zprávou Neighbor Advertising (NA), pak je prakticky zaručeno, že adresa je jedinečná a může ji host použít. Pokud host obdrží NA, znamená to, že adresa není jedinečná a host musí vygenerovat nové Interface ID.

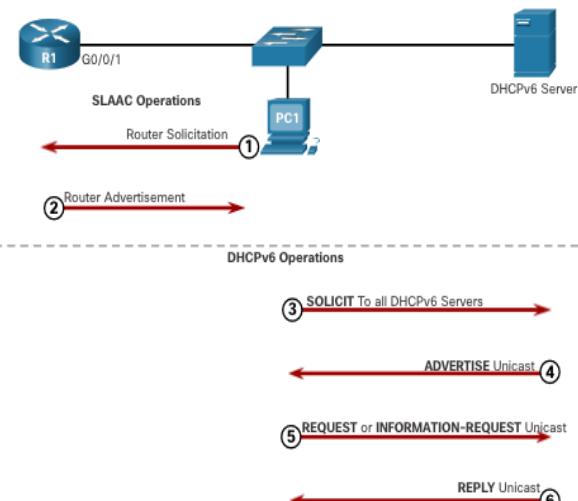
DAD se standardě nevyžaduje, protože 64bitové ID rozhraní poskytuje 18 quintillionů možností. Šance na duplicitní adresu je proto minimální. IETF (Internet Engineering Task Force) však použití DAD doporučuje. Většina operačních systémů proto DAD provádí na všech IPv6 unicast adresách bez ohledu na to, jak je adresa nakonfigurována.

DHCPv6

Existují dvě varianty nastavení a stavový a bezstavový DHCP v6 server. Stavový DHCPv6 nevyžaduje SLAAC zatímco bezstavový DHCPv6 ano. Bez ohledu na to, když RA indikuje použití DHCPv6 nebo stavového DHCPv6 je proces následující:

1. Host posílá Router Solotion zprávu.
2. Router odpovídá Router Advert zprávou.
3. Host posílá DHCPv6 SOLICIT zprávu.
4. DHCPv6 server odpovídá ADVERTISE zprávou.
5. Host odpovídá DHCPv6 serveru.
6. DHCPv6 server odesílá REPLY zprávu.

Pro případ nasazení filtrování provozu na routerech a rozhraní hostů je třeba vědět, že DHCPv6 zprávy ze serveru pro klienta používají UDP cílový port 546, zatímco zprávy od klienta na DHCPv6 server UDP cílový port 547.



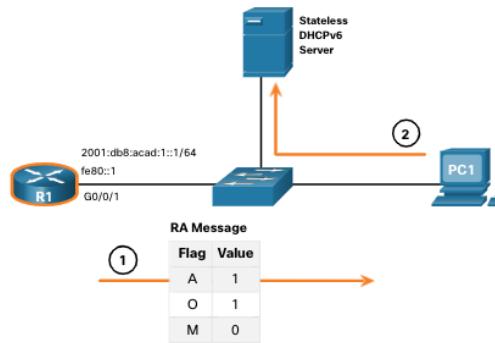
Průběh Stateless DHCPv6

Průběh komunikace pro bezstavové přidělení IPv6 adresy je následující. Pokud RA označuje bezstavovou metodu DHCPv6, host použije informace v RA zprávě k adresování a kontaktuje server DHCPv6 pro další informace.

Poznámka: DHCPv6 server poskytuje klientům pouze konfigurační parametry a nespravuje seznam přiřazených IPv6 adres (proto je označen jako bezstavový).

Na příkladu na obrázku PC1 přijímá bezstavovou RA zprávu obsahující:

- IPv6 GUA síťový prefix a délku prefixu.
- Příznak A nastavený na 1 informuje hosta, aby použil SLAAC.
- Příznak O nastavený na 1 informuje hosta, aby vyhledal další konfigurační informace na DHCPv6 serveru.
- Příznak M je nastavený na výchozí hodnotu 0.



PC1 následně posílá DHCPv6 SOLICIT zprávu k vyhledání doplňujících informací z bezestavového DHCPv6 serveru.

Nastavení bezestavového DHCPv6 na rozhraní

Fyzicky se aktivace bezestavového DHCPv6 na rozhraní provede nastavením Příznaku O na 1 příkazem **ipv6 nd other-config-flag** na daném rozhraní, jak je vidět na výpisu na aktuálním snímku.

Naznačený výstup potvrzuje, že RA řekne přijímajícím hostům, aby použili bezestavovou automatickou konfiguraci (příznak A = 1) a kontaktovali DHCPv6 server, aby získali další informace o konfiguraci (O příznak = 1).

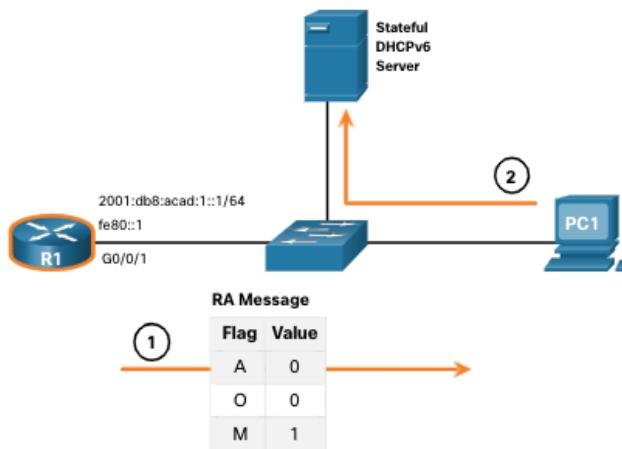
Z výpisu kontrolního příkazu **show ipv6 interface g0/0/1 | begin ND** můžeme zjistit výsledek nastavení nebo stav nastavení, pokud si nejsme jistí nastavením konfigurace u neznámého Routeru.

K resetování rozhraní na výchozí možnost *SLAAC only* (O flag = 0) se použijte příkaz **no ipv6 nd other-config-flag**

Funkce stavového DHCPv6

Pokud RA obsahuje příznak metody stavového DHCPv6, host kontaktuje DHCPv6 server pro získání všech informací o konfiguraci. Stavový DHCPv6 server v tomto případě spravuje seznam přiřazených IPv6 adres. Na obrázku máme uvedený případ, kde jsou šipkami vyjádřené kroky. V prvním PC1 přijme stavovou RA zprávu obsahující:

- IPv6 GUA síťový prefix 2001:db8:acad:1 a délku prefixu 64.
- Příznak A a O (automatické konfigurace adresy a doplňujících informací) jsou nastavené na „0“ a informují hosta, že bude třeba kontaktovat DHCPv6 server, což potvrzuje ...
- Příznak M, který je nastavený na hodnotu 1, která k tomuto kroku vyzývá.



PC1 následně posílá multicast zprávu DHCPv6 SOLICIT na adresu FF02::2, tedy all routers, aby získalo potřebné informace ze stavového DHCPv6 serveru.

Povolení stavové DHCPv6 na rozhraní

Pro povolení služby stavového DHCPv6 serveru na rozhraní je třeba provést zadáním příkazu konfigurace **ipv6 nd managed-config-flag**, čímž dojde k nastavení příznaku M na hodnotu 1.

Příkazem **show ipv6 interface g0/0/1 | begin ND** vypíšeme stav konfigurace ipv6 na daném portu zařízení.

Konfigurace DHCPv6 Serveru

Routery s podporou DHCPv6 mohou vystupovat v různých rolích. Směrovače Cisco IOS jsou výkonná zařízení. V menších sítích není třeba nasazovat DHCPv6 server, přenosového agenta jako samostatné zařízení. Směrovač Cisco IOS lze nakonfigurovat tak, aby poskytoval kompletní služby DHCPv6 serveru. Konkrétně lze na něm nakonfigurovat na jednu z následujících variant:

- **DHCPv6 Server** - Router poskytuje stavové nebo bezstavové DHCPv6 služby.
- **DHCPv6 Client** - Rozhraní směrovače získává konfiguraci IPv6 ze serveru DHCPv6.
- **DHCPv6 Relay Agent** - Router předává požadavky na přidělení adres DHCPv6 službě, když jsou klient a server umístěny v různých sítích.

Konfigurace bezstavového DHCPv6 Serveru

Volba bezstavového DHCPv6 serveru vyžaduje, aby router nabízel klientům informace o síťovém adresování IPv6 jeho prostřednictvím v RA zprávách. Existuje pět kroků, které je v rámci konfigurace a ověření směrovače jako bezstavového DHCPv6 serveru potřeba vykonat:

1. Povolit směrování IPv6 pomocí příkazu **ipv6 unicast-routing**.
2. Definovat jméno DHCPv6 pool použitím příkazu globální konfigurace **ipv6 dhcp pool** s parametrem **POOL-NAME**.
3. Nakonfigurovat parametry pro DHCPv6 pool. Běžné se takto nastavují **dns-server X:X:X:X:X:X:X** a **domain-name name**.
4. Svázat rozhraní a pool pomocí příkazu **ipv6 dhcp server POOL-NAME**.

Ručně se následně změní příznak O z 0 na 1 za pomocí příkazu rozhraní **ipv6 nd other-config-flag**. Odeslané RA zprávy na tomto rozhraní naznačují, že na bezstavovém DHCPv6 serveru jsou k dispozici další informace. Příznak A má ve výchozím nastavení hodnotu 1. Pro klienty znamená, aby si pomocí SLAAC vytvořili vlastní GUA.

K ověření, zda hostu byla přidělena IPv6 adresa slouží příkaz **ipconfig / all**.

Konfigurace DHCPv6 bezstavového klienta

Router může plnit roli DHCPv6 serveru, relay agenta a DHCPv6 klienta a získat IPv6 konfigurační data od DHCPv6 serveru.

1. Je třeba povolit IPv6 směrování pomocí příkazu **ipv6 unicast-routing**.

2. Klientský směrovač nakonfigurovat, aby si vytvořil LLA. Link-local IPv6 adresa je vytvořena na rozhraní routeru, když je nakonfigurována global-unicast adresa, nebo bez GUA pomocí konfiguračního příkazu **ipv6 enable**. Cisco IOS používá k vytvoření ID rozhraní metodu EUI-64.
3. Konfigurace klientského směrovače pomocí příkazu **ipv6 address autoconfig**, aby použil SLAAC.
4. Ověřit pomocí **show ipv6 interface brief** zda je klientskému routeru přiřazen GUA.
5. Ověřit příkazem **show ipv6 dhcp interface g0/0/1**, zda klientský směrovač obdržel od DHCPv6 doplňkové informace, jako je DNS server a domain name.

Konfigurace stavového DHCPv6 Serveru

Druhou variantou je konfigurace stavového DHCPv6 Serveru. Ta vyžaduje, aby router s povoleným IPv6 protokolem předal hostu informaci, aby pro získání IPv6 adresy a doplňujících informací pro adresování kontaktoval DHCPv6 server. Konfiguraci a ověření funkci Routeru jako stavového DHCPv6 serveru lze provést v pěti krocích:

1. Povolte IPv6 směrování příkazem **ipv6 unicast-routing**.
2. Definujte jméno DHCPv6 pool příkazem globální konfigurace **ipv6 dhcp pool POOL-NAME**.
3. Nakonfigurujte volby DHCPv6 pool. Mezi běžné volby patří příkaz **address prefix**, domain name, IP adresa DNS serveru a další.
4. Propojte rozhraní a pool pomocí konfiguračního příkazu rozhraní **ipv6 dhcp server POOL-NAME**.

Ručně změňte příznak M z 0 na 1 pomocí příkazu rozhraní **ipv6 nd managed-config-flag**. Ručně změňte příznak A z 1 na 0 pomocí příkazu rozhraní **ipv6 nd prefix default no-autoconfig**, aby byl klient informovaný, že nemá pro vytvoření GUA použít SLAAC. Směrovač nyní bude na požadavky stavového DHCPv6 odpovídat informacemi obsaženými v poolu.

5. Ověřte pomocí příkazu **ipconfig / all**, zda hosté obdrželi informace o IPv6 adresování.

Konfigurace DHCPv6 stavového klienta

Jak již bylo řečeno, Směrovač může být také DHCPv6 klientem. Aby mohl klient odesílat a přijímat IPv6 zprávy, musí mít klientský směrovač povolený **ipv6 unicast-routing** a IPv6 link-local adresu. Konfiguraci a ověření funkci Routeru coby bezestavového DHCPv6 klienta lze provést opět v pěti krocích:

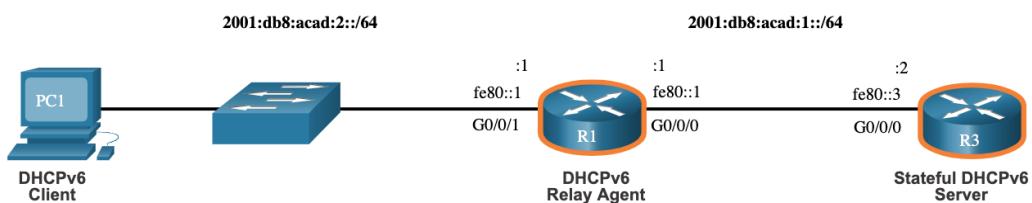
1. Povolit IPv6 směrování příkazem **ipv6 unicast-routing**.
2. Nakonfigurovat klienta Routeru, aby vytvořil LLA. IPv6 link-local adresa je na rozhraní routeru vytvořena vždy, když se konfiguruje *global unicast adresa nebo* bez ní, použitím příkazu rozhraní **ipv6 enable**. Cisco IOS používá k vytvoření Interface ID EUI-64.
3. Konfigurace klientského routeru, aby použil DHCPv6 příkazem na rozhraní **ipv6 address dhcp**.
4. Ověřit, že klientský router má přiřazenu GUA příkazem **show ipv6 interface brief**.
5. Ověřit, že klientský router přijal další nezbytné DHCPv6 informace příkazem **show ipv6 dhcp interface g0/0/1**.

Příkazy ověření funkce DHCPv6 Serveru

Příkazy ověření funkce DHCPv6 Serveru dovolují zjistit konfiguraci a také odhalit případnou neshodu. Například příkazem **show ipv6 dhcp pool**, který je velmi podobný příkazu pro IPv4, lze ověřit jméno DHCPv6 poolu a jeho parametry. Příkaz také vrací počet aktivních klientů. Druhým, také velmi podobným příkazem k IPv4, je **show ipv6 dhcp binding**, který zobrazí IPv6 link-local adresy klientů a GUA přiřazených serverem. Tato informace je udržována stavovým DHCPv6 serverem (bezestavový DHCPv6 server tyto informace neudržuje).

Konfigurace DHCPv6 relay agenta

Jestliže je DHCPv6 server dislokovaný v jiné síti než klient, pak musí být IPv6 router konfigurován jako DHCPv6 relay agent.



Konfigurace DHCPv6 relay agenta je podobná konfiguraci IPv4 routeru coby DHCPv4 relay.

Na obrázku je zvýrazněný Relay agent router a pod ním, v CLI, příkaz, kterým se nastavení řeší na rozhraní směrujícím k DHCPv6 klientům.

```
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# ipv6 dhcp relay destination 2001:db8:acad:1::2 G0/0/0
R1(config-if)# exit
R1(config)#
```

Je v něm specifikována adresa DHCPv6 serveru a adresa, na kterou se má provoz s požadavky směrovat. LLA použít nelze, neboť se neroutují.

Ověření funkce DHCPv6 relay agenta

K ověření, že je DHCPv6 relay agent funkční, se používá příkazů **show ipv6 dhcp interface** a **show ipv6 dhcp binding**.

Na prvním výpisu CLI je zřejmé, že adaptér je v relay módu a jaká má související nastavení.

```
R1# show ipv6 dhcp interface
GigabitEthernet0/0/1 is in relay mode
Relay destinations:
 2001:DB8:ACAD:1::2
 2001:DB8:ACAD:1::2 via GigabitEthernet0/0/0
R1#
```

Druhý výpis obsahuje seznam přidělených adres na R3, tedy DHCPv6 serveru. Je zde jeden klient, daný LLA. Identifikace klienta a serveru.

Každý klient a server DHCPv6 je identifikován tzv. DHCP unique identifier (DUID). Ten se přenáší v identifikátoru klienta a možnostech identifikátoru serveru. DUID je jedinečný pro všechny klienty a servery DHCP a je pevně daný

```
R3# show ipv6 dhcp binding
Client: FE80::5C43:EE7C:2959:DA68
DUID: 0001000124F5CEA2005056B3636D
Username : unassigned
VRF : default
IA NA: IA ID 0x03000C29, T1 43200, T2 69120
Address: 2001:DB8:ACAD:2:9C3C:64DE:AADA:7857
          preferred lifetime 86400, valid lifetime 172800
          expires at Sep 29 2019 08:26 PM (172710 seconds)
R3#
```

pro každého konkrétního klienta nebo server. DHCPv6 používá DUID založené na adresách linkové vrstvy. K vytvoření DUID používá zařízení MAC adresu z rozhraní s nejnižším číslem. Předpokládá se, že toto síťové rozhraní je se zařízením trvale svázáno.

Ověření přidělení IPv6 adres na straně hostů s OS Windows se provádí příkazem **ipconfig /all**.

Počítačové sítě 2

8, Koncept zabezpečení LAN

Název bloku	Obsah
Zabezpečení koncových uzlů	Objasnění, jak použít zabezpečení koncového uzlu ke zmírnění útoků
Řízení přístupu (Access Control)	Vysvětlení, jak se používá AAA a 802.1x k ověřování koncových uzlů a zařízení zapojených v LAN
Bezpečnostní hrozby L2 vrstvy	Identifikace zranitelností na druhé vrstvě (L2)
Útok na tabulku MAC adres	Vysvětlení, jak může útok na tabulku MAC adres narušit zabezpečení LAN
Útoky v LAN	Objasnění ohrožení zabezpečení LAN útoky

Zabezpečení koncových uzlů

Současné síťové útoky

Každá doba byla spojována s různými útoky na počítače, později i datovou síť. Zpravodajská média v poslední době, stále častěji přinášejí informace o útocích na podnikové sítě. Na Internetu lze články v různých jazykových mutacích jednoduše vyhledat. Nalezené útoky budou s největší pravděpodobností zahrnovat jednu nebo více z následujících možností:

- **Distributed Denial of Service (DDoS)** - Distribuované odmítnutí služby je koordinovaný útok z mnoha zařízení zvaných zombie s úmyslem degradovat nebo zastavit veřejný přístup na web a zdroje organizace.
- **Data Breach** - Jde o Porušení dat, což je typ útoku, při kterém jsou napadeny datové servery nebo hosté organizace, vedoucí ke zcizení důvěrné informace.
- **Malware** - Jedná se o útok, při kterém jsou hosté organizace infikováni škodlivým softwarem, který způsobuje řadu problémů. Příkladem je *ransomware*, jehož mutace *WannaCry*, šifruje data na hostech a uzamyká k nim přístup, dokud není zaplacenou výkupné.

Řešení síťové bezpečnosti

Jako prevence proti nim je třeba realizovat zabezpečení sítě, kterým všeobecně říkáme řešení síťové bezpečnosti. K ochraně perimetru sítě před přístupem z vnějšku jsou používána různá zařízení. Patří mezi ně následující:

- **Router s aktivní službou virtuální privátní sítě (VPN)** - poskytuje zabezpečené připojení vzdálených uživatelům přes veřejnou síť do podnikové sítě. Služby VPN lze také integrovat do brány firewall.
- **Next-Generation Firewall (NGFW)** - poskytuje stavovou kontrolu paketů, viditelnost a kontrolu aplikace, a next-generation intrusion prevention system (NGIPS), advanced malware protection (AMP) a filtrování URL.
- **Network Access Control (NAC)** - zahrnuje služby authentication, authorization, a accounting (AAA). Poskytuje stavovou kontrolu paketů, viditelnost aplikací. Ve velkých firmách mohou být tyto služby začleněny do zařízení, která dokáží spravovat zásady přístupu pro velké množství uživatelů a širokou škálu typů zařízení. Příkladem NAC zařízení je Cisco Identity Services Engine (ISE).

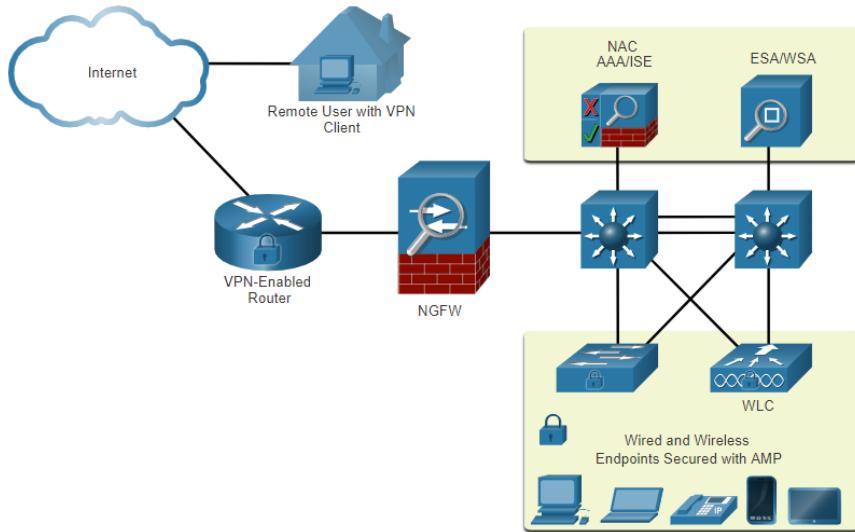
Ochrana koncových uzlů

K sítí jsou připojené koncové body se kterými jsou často v interakci lidé. Koncovými uzly/body jsou méně hosté (notebooky, stolních počítače, servery a IP telefony, a také zařízení vlastněná zaměstnanci). Koncové uzly jsou obzvláště náchylné k útokům souvisejícím s malwarem, které pocházejí z e-mailu nebo procházení webu.

Koncové uzly obvykle používají tradiční funkce zabezpečení jako je antivirus nebo antimalware, lokální firewally a host-based intrusion prevention systémy (HIPS).

V dnešní době, dá se říci, jsou desktopy nejlépe chráněny kombinací Network access control, softwaru advanced malware protection, e-mailového bezpečnostního zařízení (ESA) a webového bezpečnostního zařízení (WSA).

Na obrázku vidíme vcelku typické uspořádání. Firemní síť a koncový bod v domácnosti připojený k organizaci pomocí VPN spoje, což je velice častý případ, zvlášť teď v době preferovaných home office.



Bohužel, některé firmy povolují přímý přístup na vzdálenou plochu RDP, bez výraznějšího zabezpečení.

Cisco bezpečnostní zařízení pro el. poštu (ESA)

Budeme li se bavit o serverové platformě, pak nejpoužívanější službou je jednoznačně email. S ní souvisí také značný počet cílených útoků na ni. Existuje celá řada řešení, mezi které patří i

zařízení Cisco ESA, které je navrženo pro monitorování protokolu SMTP (Simple Mail Transfer Protocol). Cisco ESA je neustále aktualizován zdroji v reálném čase od Cisco Talos, který detekuje a koreluje hrozby a řešení pomocí celosvětového systému monitorování databáze. Tato data zpravodajských informací o hrozbách získává server Cisco ESA každé tři až pět minut.

Zde jsou některé z funkcí Cisco ESA:

- Blokování známých hrozeb
- Oprava proti skrytému malwaru, který se vyhnul počáteční detekci
- Likvidace e-mailů se špatnými odkazy
- Blokování přístupu k nově infikovaným webům.
- Šifrování obsahu odchozích e-mailů zabráňuje ztrátě dat.

Cisco bezpečnostní zařízení pro web (WSA)

Druhou silně zastoupenou službou je web. I zde je v komerčním prostoru dostupné velké množství řešení, která je mají chránit.

Cisco Web Security Appliance (WSA) je technologie zmírňující dopad webových hrozeb. Pomáhá organizacím řešit výzvy zabezpečení a kontroly webového provozu. Kombinuje pokročilou ochranu před malwarem, viditelnost a kontrolu aplikací, přijatelné řízení zásad a reportování. Poskytuje úplnou kontrolu nad tím, jak uživatelé přistupují k internetu. Běžné funkce a aplikace, jako je chat, zasílání zpráv, video a zvuk, lze povolit, omezit limity času a šířky pásma nebo je blokovat podle požadavků organizace.

WSA může využít černé listiny (black list) adres URL k filtrování URL adres, skenování malwaru, kategorizaci adres URL, filtrování webových aplikací a šifrování a dešifrování webového provozu. Některé systémy kontrolují provoz, a pokud dojde k podezřeným aktivitám ve spojení s danou adresou, automaticky ji na black list zařadí.

Pochopitelně, existuje i white list, který obsahuje někdy sporné adresy nebo adresy, které musí být pro organizaci dostupné vždy.

Řízení přístupu (Access Control)

Ověření lokálním heslem

Na síťových zařízeních lze provádět ověřování mnoha způsoby a každá metoda nabízí různé úrovně zabezpečení. Prvotní zabezpečení spočívá v zabránění neautorizovanému přístupu do CLI na konzoli zařízení a následně do konfiguračního režimu. Nejjednodušší metodou ověřování vzdáleného přístupu je konfigurace kombinace přihlašovacího jména a hesla na konzoli, vty linkách a aux portech.

Z první kapitoly víme, že telnet používá otevřený formát přenosu jména a hesla a je tedy de facto nepoužitelný. Naproti tomu je SSH výrazně bezpečnější formou vzdáleného přístupu. Vyžaduje také uživatelské jméno a heslo, která lze ověřit místně, avšak přenos po síti je zašifrovaný. Metoda ověření v místní databázi má určitá omezení:

- Uživatelské účty musí být konfigurovány lokálně na každém zařízení, které není škálovatelné.
- Metoda neposkytuje žádnou záložní metodu ověřování.
- Pro větší množství prvků je vhodné použít ověření vůči jiné centrální autoritě např. prostřednictvím RADIUS serveru.

AAA komponenty

Termín bezpečnost je často spojovaný se zkratkou AAA. Znamená Authentication, Authorization a Accounting. Tato komponenta poskytuje primární rámec pro nastavení řízení přístupu na síťovém zařízení. AAA je způsob jak kontrolovat, kdo smí přistupovat k síti (ověřování), co může dělat, když tam je (autorizace), a auditovat, jaké akce provedly při přístupu k síti (vyúčtování).

Ověření – Authentication

V rámci řízení přístupu tedy využíváme Authentication, které je prvním A z před chvílí uvedené zkratky AAA.

Pro ověřování běžne používáme dvě metody, lokální a server-based. Obě jsou běžné při implementaci AAA.

Lokální AAA ověřování:

Metoda ukládá uživatelská jména a hesla lokálně do síťového zařízení (např. Směrovače Cisco). Uživatelé se autentizují v místní databázi. Lokální AAA je ideální pro malé sítě.

Server-Based AAA ověřování:

Znamená, že router přistupuje k centrální ověřovací autoritě, nějakému AAA serveru. AAA server obsahuje uživatelská jména a hesla všech uživatelů. Router používá ke komunikaci s ním buď

protokol *Remote Authentication Dial-In User Service* (RADIUS) nebo *Terminal Access Controller Access Control System* (TACACS +).

Pokud máme rozsáhlejší síť, kde koexistuje více směrovačů a přepínačů, je vhodnější serverový model AAA ověřování.

Oprávnění – Authorization

Druhým A v AAA je Authorization neboli autorizace. Její provedení je automatické a nevyžaduje, aby uživatelé po ověření prováděli další kroky. Autorizace určuje, co uživatelé mohou a nemohou dělat v síti po ověření. Při autorizaci se využívá sada atributů, které popisují přístup uživatele k síti nebo síťovému prvku nebo také lokální službě.

Atributy tedy používá AAA server k určení oprávnění a omezení pro daného uživatele. Implementaci najdete v access control listech, doménových politikách systémů firmy Microsoft a jinak pojmenovaných bezpečnostních modulů a prvků.

Účtování – Accounting

Poslední A zkratky AAA znamená účtování. Představit si pod ním můžeme shromažďování a hlášení údajů o využití prostředků. Údaje účtování lze použít pro účely auditu nebo fakturace. Shromážděná data mohou zahrnovat časy zahájení a ukončení připojení, provedené příkazy, počet paketů či počet bajtů v návaznosti na pravidla.

Accounting neboli účtování se primárně využívá společně s ověřováním.

Server služby si vede podrobný protokol toho, co ověřený uživatel na zařízení dělá. Může monitorovat všechny uživatelem vykonané příkazy a provedené konfigurace. Výstupní protokol obsahuje celou řadu datových polí, která vyvolaný záznam identifikují, tedy včetně uživatelského jména, data, času a provedené akce. Tyto informace mohou být užitečné při odstraňování problémů se zařízeními. Poskytují také důkazy o tom, kdy byli útočníci aktivní, pokud nám však záznamy logu nesmaží.

Řízení přístupu 802.1X

Pro řízení přístupu lze využít standard IEEE 802.1X. Jde o protokol kontroly přístupu a autentizace na portech aktivních prvků – primárně přepínačů. Tento protokol brání neoprávněným pracovním stanicím v připojení k síti LAN prostřednictvím veřejně přístupných portů přepínačů. Ověřovací server ověří každou pracovní stanici, která je připojena k portu přepínače ještě před zpřístupněním služeb nabízených přepínačem nebo síti LAN.

Zařízení s implementovanou 802.1X, zastávají v síti následující specifické role:

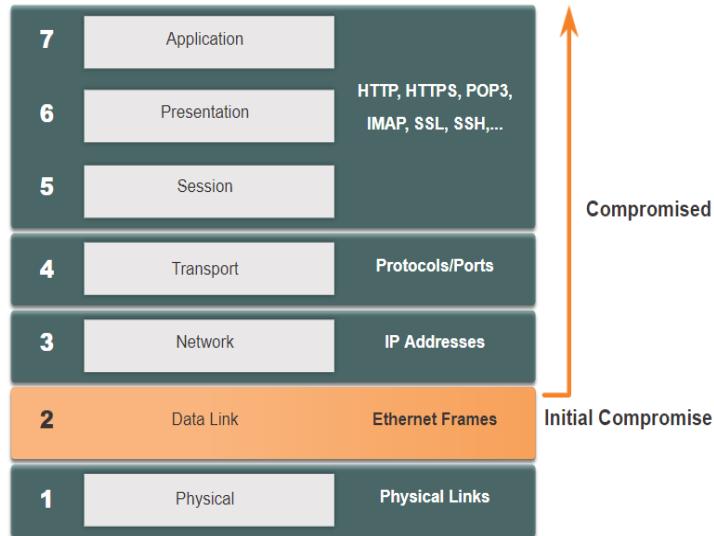
- **Klient (Supplicant)** - je zařízení s klientským softwarem standardu 802.1X, který je k dispozici pro kabelová nebo bezdrátová zařízení.
- **Switch (Authenticator)** – funguje jako prostředník mezi klientem a ověřovacím serverem. Vyžaduje identifikační informace od klienta, které ověří na ověřovacím serveru a výsledek předá klientovi. Dalším zařízením, které může fungovat jako autentizátor, je bezdrátový přístupový bod (access point).
- **Ověřující server** – ověřuje totožnost klienta a oznamuje přepínači nebo bezdrátovému přístupovému bodu, zda má nebo nemá klient oprávnění pro přístup k síti LAN.

Bezpečnostní hrozby L2 vrstvy

Zranitelnosti L2 vrstvy

Připomeňme, že referenční model OSI je rozdělen do sedmi vrstev, které fungují nezávisle na sobě. Obrázek ukazuje funkci každé vrstvy a základní prvky, které lze využít.

Správci sítě běžně implementují bezpečnostní řešení k ochraně prvků od třetí po sedmou vrstvu. K ochraně aktivních prvků používají VPN, firewally a zařízení IPS k ochraně elementů sítě. Pokud je ohrožena vrstva 2, jsou ovlivněny také všechny vrstvy nad ní. Například pokud útočník s přístupem k interní síti zachytí rámce druhé vrstvy, pak by veškeré zabezpečení implementované ve výše uvedených vrstvách bylo zbytečné. Útočník by pak mohl způsobit velké škody na síťové infrastruktuře LAN na druhé vrstvě.



Kategorie útoků na přepínače

Útoky lze kategorizovat. Zabezpečení je jen tak silné jako nejslabší článek v systému. L2 vrstva je za slabý článek považována. Důvodem je, že LAN byly tradičně pod administrativní kontrolou jedné organizace. Ve své podstatě jsme důvěrovali všem osobám a zařízením připojeným k naší síti LAN. Dnes, díky BYOD a sofistikovanějším útokům, se naše LAN staly zranitelnějšími vůči penetraci.

Definujeme šestici kategorií.

Útok na MAC tabulku ... označujeme jako MAC address flooding představuje útok „zaplavení“ MAC adresami.

VLAN útoky ... Zahrnují útoky VLAN hopping a VLAN double-tagging a také útoky mezi zařízeními na společné VLAN.

DHCP útoky ... Zahrnují DHCP starvation a DHCP spoofing.

ARP útoky ... má dva zástupce, ARP spoofing a ARP poisoning.

Address Spoofing útoky ... jsou vedené na MAC a IP adresy.

STP útoky ... jak zkratka napovídá, souvisejí s manipulací Spanning Tree Protocolu.

Techniky zmírňující dopady útoků na přepínače

Odpověď na útoky je logicky obrana. Řešení ochrany L2 vrstvy jsou...

Port Security, které zabraňují mnoha typům útoků, včetně útoků na MAC adresy a DHCP flooding.

DHCP Snooping brání útokům DHCP starvation a DHCP spoofing.

Dynamic ARP Inspection (DAI) - Brání útokům ARP spoofing a ARP poisoning.

IP Source Guard (IPSG) Brání útokům typu MAC nebo IP address spoofing.

Uvedená řešení nebudou účinná, pokud nebudou zabezpečeny protokoly správy. K tomu jsou doporučeny následující strategie:

- Vždy používejte zabezpečené varianty protokolů pro správu, jako je SSH, Secure Copy Protocol (SCP), Secure FTP (SFTP) a Secure Socket Layer / Transport Layer Security (SSL / TLS).
- Zvažte použití out-of-band sítě pro správu zařízení.
- Použijte vyhrazenou správu VLAN, kde se nenachází nic jiného než provoz správy.
- Pomocí ACL seznamů filtrojte nežádoucí přístup.

Útok na MAC tabulku

Provoz přepínače (rekapitulace)

Připomeňme si, že pro rozhodování o přenosu rámců si L2 přepínač na základě zdrojových MAC adres z přijatých rámců vytváří tabulku MAC adres a s nimi spřaženými porty. Tyto tabulky jsou uloženy v paměti a slouží k efektivnějšímu přepínání rámců.

Na výpisu z CLI je výpis přepínače S1, provedený příkazem **show mac address-table dynamic**. Má tvar tabulky, z níž lze zjistit MAC adresu v dané síti LAN, že se jedná o dynamický záznam, který podléhá předdefinované životnosti a s kterým portem je adresa spřažená.

MAC Address Table Flooding

Prvním typem útoku, který si probereme je MAC Address table flooding.

Premisou je, že všechny MAC tabulky mají pevnou velikost. V důsledku toho mohou přepínači docházet zdroje, do kterých se budou ukládat MAC adresy. Útoky MAC address flooding využívají tohoto omezení tak, že bombardují přepínač enormním množstvím falešných zdrojových MAC adres, dokud není tabulka MAC adres přepínače plná. Následné chování všech přepínačů spojené s přetečením je shodné.

Když k tomu dojde, přepínač zachází s rámcem jako s neznámým unicasterem a začne zaplavovat provozem všechny porty na stejně VLAN bez odkazování na MAC tabulku. Na všechny porty jede veškerá komunikace, což je podmínka, která umožňuje útočníkovi zachytit všechny rámce odeslané mezi uzly v místní síti LAN nebo VLAN.

Pro upřesnění, provoz je zaplaven pouze v místní síti LAN nebo VLAN. Útočník tak může zachytit pouze provoz v ní.

Zmírnění MAC Address Table útoků

Ke zmírnění útoků vedených na MAC tabulku je třeba pochopit jejich princip.

Díky nástrojům jako je **macof** může útočník vytvořit útok přetečení MAC tabulky velmi rychle. Například přepínač Catalyst 6500 může do své tabulky MAC adres uložit 132 000 MAC adres. Nástroj jako **macof** dokáže zaplavit přepínač až 8 000 falešných snímků za sekundu. K přetečení tabulky adres tak dojde během několika sekund.

Dalším důvodem, proč jsou tyto útočné nástroje nebezpečné, je to, že ovlivňují nejen lokální přepínač, ale mohou ovlivnit i další připojené L2 přepínače. Když je tabulka MAC adres přepínače plná, začne zaplavovat všechny porty, včetně těch, které jsou připojeny k jiným L2 přepínačům.

Aby se zmírnily popsané útoky, musí správci sítě implementovat zabezpečení portů. To umožní přepínače registrovat na portu pouze definovaný počet zdrojových MAC adres.

Zabezpečení portů bude diskutováno dále.

Útoky na LAN

VLAN Hopping útok

Jako první si uvedeme *VLAN hopping* útok. Ten umožnuje, aby provoz z jedné VLAN byl viditelný jinou VLAN a to bez pomoci routeru. V základu útoku nakonfiguruje útočník hostitelský systém tak, aby fungoval jako přepínač a využil funkci *automatic trunking port*, která je ve výchozím nastavení povolena na většině portů přepínače.

Když útočník nasadí hosta do sítě, dojde k podvržení signalizace 802.1Q a protokolu DTP (Dynamic Trunking Protocol). Tím dosáhne trunk komunikace s připojeným přepínačem.

Situace je vyobrazena na obrázku na snímku, na němž vidíme, že je-li útočník úspěšný, přepínač vytvoří neautorizované trunkové spojení s hostem v jeho správě. Nyní má útočník přístup ke všem VLAN na přepínači a může odesílat a přijímat provoz na jakékoli VLAN a efektivně mezi nimi přecházet.

Útok není vedený na zranitelnost, jen umně využívá funkcionalitu, která ve výchozí konfiguraci aktivního prvku působí automatickou konfiguraci vzájemně propojených přepínačů.

VLAN Double-Tagging útok

Další útok na VLAN je již sofistikovanější. Útočník za jisté situace může do rámce vložit skrytý 802.1Q tag, ačkoli ten již 802.1Q tag obsahuje. Úprava umožňuje rámci přejít na VLAN, kterou původní 802.1Q tag neurčil. Průběh útoku je následující:

Krok 1: Útočník pošle na přepínač dvojitě označený 802.1Q rámec. Vnější záhlaví má tag VLANy útočníka, které je stejné jako nativní VLAN trunk portu.

Krok 2: Rámec dorazí na první přepínač, který nahlédne do prvních 4 B 802.1Q tagu. Přepínač vidí, že rámec je určen pro nativní VLAN. Po odstranění VLAN tagu přepínač paket na všechny porty nativní VLAN. Rámec není znova označen, protože je součástí nativní VLAN. V tomto okamžiku je vnitřní VLAN tag stále neporušený neboť nebyl zkontovalán prvním přepínačem.

Krok 3: Rámec dorazí k druhému přepínači, který předpokládá, že by měl být pro nativní VLAN. Víme, že rámce zasílané mezi zařízeními v nativní VLAN nejsou, v souladu s 802.1Q, odesílacím přepínačem označované. Druhý přepínač, s portem konfigurovaným jako trunk, se dívá na vnitřní 802.1Q tag, který vložil útočník, a vidí, že rámec je určen jisté cílové VLAN. Rámec následně odešle do cíle nebo jím síť zaplaví, v závislosti na tom, zda pro cíl existuje záznam v tabulce MAC adres nebo ne.

Nyní známe princip. Je třeba podotknout, že Útok dvojitého značení VLAN je jednosměrný a funguje pouze v případě, že je útočník připojen k portu umístěnému ve VLAN jako je nativní VLAN trunk portu. Myšlenka spočívá v tom, že dvojitě značení umožňuje útočníkovi odesílat data hostům nebo serverům ve VLAN, které by jinak byly blokovány nějakým typem konfigurace řízení přístupu.

Pravděpodobně bude zpětný provoz povolen, což útočníkovi umožní komunikovat se zařízeními na normálně blokované VLAN.

Zmírnění VLAN útoků VLAN hopping a VLAN double-tagging attacks může být provedeno implementací trunk security postupy, jak již bylo naznačeno v předchozím modulu:

- Zakažte *trunk* na všech přístupových portech.
- Deaktivujte automatické auto trunking, *trunk* tak může být nastavený jen ručně.
- Ujistěte se, že nativní VLAN se používá pouze pro *trunk* linky.

DHCP zprávy

Jako další si uvedeme zranitelnost DHCP zpráv. Víme, že servery DHCP dynamicky poskytují klientům informace o IP konfiguraci, včetně IP adresy, masky podsítě, výchozí brány, DNS serverů a dalších. Klient broadcastem zjišťuje auto konfigurační parametry, DHCP mu unicastem odpovídá.

DHCP útoky

K narušení této vazby existují dva typy útoků a to DHCP starvation a DHCP spoofing. Oba lze eliminovat pomocí DHCP snooping. Uvedené útoky lze popsát následovně.

U DHCP Starvation je cílem vytvořit DoS útok působící zamítnutí připojení klientů. Útoky na „hladovění“ DHCP vyžadují útočný nástroj jakým je např. *Gobbler*. Ten má schopnost zjistit rozsah pronajímatelných IP adres a všechny si je pronajmout. K tomu vytváří DHCP discovery zprávy s falešnými MAC adresami.

Útok DHCP Spoofing je proveden za pomoci podvrženého DHCP serveru připojeného k síti, který poskytuje falešné konfigurační IP parametry legitimním klientům. Podvržený server může poskytnout celou řadu zavádějících informací a to včetně **podvržené default gateway, podvrženého DNS serveru a špatné IP adresy**. Podvržený server poskytuje falešnou výchozí bránu nebo svou IP adresu, aby vytvořil útok typu man-in-the-middle. Je reálné, že může zůstat zcela nezjištěno, že útočník zachytává tok dat v síti. Dále DHCP server poskytuje nesprávnou adresu serveru DNS, která uživatele nasměruje třeba na podvržený web. Taktéž může podvržený DHCP server poskytnout neplatné IP adresy pro danou síť a vytvořit tak DoS útok na DHCP klienta.

DHCP Snooping je ochranou proti uvedeným útokům, což je ve své podstatě ochrana proti spuštění cizího DHCP serveru v síti.

Lze jej zapnout pro na definované VLANy, čímž se porty v nich berou jako nedůvěryhodné. Manuálně se pak nakonfigurují ty, které důvěryhodné jsou. Na těch je služba DHCP dostupná nebo jde o *trunk* porty. Po zapnutí se nedůvěryhodné zprávy zahazují, přičemž záznamy o provedené operaci se logují.

DHCP Snooping Binding Database

Odposlechem DHCP komunikace se vytváří databáze záznamů lokálně nebo mimo aktivní prvek. Obsahuje informace o přidělení IP adres konkrétním MAC adresám a dobou exspirace. Lze ji konfigurovat i manuálně.

ARP útoky

Dále tu máme útoky na ARP protokol. Víme, že Hosté vysílají ARP požadavky k nalezení MAC adresy hosta s cílovou IP adresou. Všichni hosté v podsítích přijímají a zpracovávají ARP request zprávy. Host s IP adresou shodnou v ARP požadavku odešle ARP Reply.

Podstatou útoku je, že útočník může prostřednictvím infikovaného klienta zaslat nevyžádanou ARP odpověď, nazývanou „bez důvodná ARP“. Hosté ve shodné podsítě si z ní ukládají MAC a IP adresu do svých ARP tabulek.

Útočník může přepínač odeslat bezdůvodnou ARP zprávu obsahující falešnou MAC adresu. Přepínač v reakci na ni odpovídajícím způsobem aktualizuje svou MAC tabulku. Při typickém útoku odesílá útočník nevyžádané ARP odpovědi ostatním hostům v podsíti s MAC adresou jeho stroje a IP adresou výchozí brány, čímž efektivně nastavuje útok typu man-in-the-middle.

Na internetu je k dispozici mnoho nástrojů k vytváření ARP útoků typu man-in-the-middle.

IPv6 používá protokol ICMPv6 Neighbor Discovery Protocol pro rozlišení adresy L2 vrstvy. Protokol IPv6 zahrnuje strategie ke zmírnění Neighbor Advertis spoofingu, na podobném principu IPv6 brání i podvrženým ARP Reply.

ARP spoofing a ARP poisoning se eliminují implementací Dynamic ARP Inspection (DAI).

V online materiálech CCNA2 jsou pro lepší pochopení dostupná videa.

Address Spoofing útoky

Útoky vedené podvržením IP adresy se rozumí, že útočník zneužije platnou IP adresu jiného zařízení v podsíti nebo použije náhodnou IP adresu. IP address spoofing je obtížné zmírnit, zvláště když se používá uvnitř podsítě, do které IP adresa patří.

K útokům s podvržením MAC adresy dochází, když útočník změní svou MAC adresu tak, aby odpovídala jiné známé MAC adrese cílového hosta v dané síti. Přepínač přepíše aktuální záznam tabulky MAC a přiřadí MAC adresu novému portu. Poté neúmyslně přepošle rámce určené pro cílového hosta falešnému hostu útočníka.

Když cílový host odešle data, přepínač chybu opraví a MAC adresu převedením na původní port. Chcete-li zabránit tomu, aby přepínač vrátil přiřazení portů do správného stavu, může útočník vytvořit program nebo skript, který bude nepřetržitě odesílat rámce do přepínače, aby přepínač udržoval nesprávné nebo falešné informace.

Na L2 vrstvě neexistuje žádný bezpečnostní mechanismus, který by přepínač umožňoval ověřovat zdroj MAC adres, což ji tak činí velmi zranitelnou vůči spoofingu.

Výskyt falešných IP a MAC adres lze zmírnit implementací IP Source Guard (IPSG), který si ted' popíšeme.

STP útok

Předposledním zde uvedeným útokem na LAN je útok na STP protokol. Útočníci díky němu mohou v síti manipulovat protokolem Spanning Tree Protocol (STP). Cílem je podvržení kořenového mostu (root bridge) a s tím spojená změna topologie sítě. Mohou pak bezprostředně zachytit veškerý provoz v přepínané doméně.

Chcete-li útočník provést manipulační STP útok, vysílá ze své stanice *STP bridge protocol data units* (BPDU) obsahující změny konfigurace a topologie, které vynutí ve spanning-tree hierarchii přepočty. BPDU zaslané útočícím hostem oznamují nižší prioritu mostu ve snaze být zvolen jako kořenový most.

Tento útok STP lze zmírnit implementací *BPDU Guard* na všech přístupových portech pomocí příkazu **spanning-tree portfast bpduguard** jak již bylo popsáno v kapitole věnované STP protokolu.

CDP průzkum

Poslední část se věnuje protokolu CDP. Cisco Discovery Protocol (CDP) je proprietárním L2 linkovým průzkumovým protokolem. Ve výchozím nastavení je povolen na všech Cisco zařízeních. Pomáhá správcům sítí konfigurovat a odstraňovat problémy se síťovými zařízeními. CDP informace jsou odesílány na porty podporující CDP zprávy v pravidelných, nezašifrovaných a neověřených vysíláních. CDP informace zahrnují IP adresu zařízení, verzi softwaru IOS, platformu, funkce a nativní VLAN sousedním zařízením. Zařízení přijímající CDP zprávu aktualizuje z obsahu svou databázi CDP.

Pro zmírnění zneužití funkcí CDP je třeba omezit jeho použití na celých zařízeních nebo definovaných portech. Například, zákazem CDP na access portech, na které se připojují nedůvěryhodná zařízení.

Globální zákaz CDP na zařízení, lze provést příkazem **no cdp run** v globálním konfiguračním modu.

Pro globální povolení CDP, použijte příkaz **cdp run**.

Zakázat CDP na portu lze příkazem konfigurace rozhraní **no cdp enable**. Povolit CDP na portu lze příkazem **cdp enable**.

Obdobně je tomu u Link Layer Discovery Protocolu (LLDP), který je také zranitelný průzkumnými útoky. Konfigurací příkazem **no lldp run** jej lze globálně zakázat. Zakázat jej lze i na rozhraní a to příkazy **no lldp transmit** a **no lldp receive**.

Počítačové sítě 2

9, Konfigurace zabezpečení přepínače

Název bloku	Obsah
Implementace zabezp. portu	Implementace zabezpečení portu ke zmírnění útoků na tabulku MAC adres.
Zmírnění útoků na VLAN	Vysvětlení, jak konfigurovat DTP and nativní VLAN ke zmírnění útoků na VLAN.
Zmírnění útoků na DHCP	Vysvětlení, jak konfigurovat DHCP snooping ke zmírnění útoků na službu DHCP.
Zmírnění útoků na ARP	Vysvětlení, jak konfigurovat inspekci ARP ke zmírnění útoků na službu ARP.
Zmírnění útoků na STP	Vysvětlení, jak konfigurovat PortFast a BPDU Guard ke zmírnění útoků na STP.

Implementace zabezpečení portů

Zabezpečení nepoužívaných portů

Zcela jednoduchým a naprosto logickým krokem je zabezpečení nepoužívaných vstupních portů.

Útoky na L2 vrstvu jsou pro hackery jedny z nejjednodušších. Před nasazením přepínače pro produkční použití by měly být všechny porty přepínače (rozhraní) zabezpečeny. Míra zabezpečení závisí na jeho funkci.

Jednoduchou metodou, kterou mnoho správců používá k zabezpečení sítě před neoprávněným přístupem, je softwarové zakázání všech nepoužívaných portů na přepínači. provede se to tak, že na každém nevyužitém portu je nutné zadat příkaz **shutdown**. Pokud je nutné port znova aktivovat později, lze jej povolit pomocí příkazu **no shutdown**. Chcete-li nakonfigurovat celou řadu portů, použijte parametr **range**.

Alternativně lze k tomu dospět i pomocí fyzického odpojení nepoužívaných spojů na straně rozvaděče, kdy horizontální kabeláz je k přepínači připojena jen když se má datová zásuvka v dané místnosti obsadit legálním zařízením.

Zmírnění útoků na tabulku MAC adres

Dalším krokem při obraně podnikové datové infrastruktury je zmírnění útoků na tabulkou MAC adres. Nejjednodušší a nejúčinnější metodou, jak zabránit konkrétně útokům přetečení tabulky MAC adres, je povolení zabezpečení portu.

Zabezpečení portu omezuje počet platných povolených MAC adres na portu. Umožňuje správci ručně konfigurovat MAC adresy na portu nebo povolit přepínači dynamicky se naučit jen omezený počet MAC adres. Když port nakonfigurovaný se zabezpečením portů přijme rámec, porovná se zdrojová MAC adresa rámce se seznamem zabezpečených zdrojových MAC adres, které byly na portu nakonfigurovány ručně nebo dynamicky.

Omezením počtu povolených MAC adres na daném portu na jednu lze metodu zabezpečení portů použít k řízení neoprávněného přístupu do sítě.

Povolení zabezpečení portů (Port Security)

Zabezpečení portů se realizuje pomocí konfiguračního příkazu na rozhraní - **switchport port-security**.

Všimněte si v příkladu na obrázku, že příkaz **switchport port-security** byl odmítnut. Je to proto, že zabezpečení portů lze konfigurovat pouze na ručně nakonfigurovaných přístupových portech nebo ručně nakonfigurovaných *trunk* portech. Ve výchozím nastavení jsou porty přepínače L2 vrstvy nastaveny na *dynamic auto (trunk on)*. Proto je port v příkladu konfigurován příkazem pro konfiguraci přístupového rozhraní **switchport mode access**.

```
S1(config)# interface f0/1
S1(config-if)# switchport port-security
Command rejected: FastEthernet0/1 is a dynamic port.
S1(config-if)# switchport mode access
S1(config-if)# switchport port-security
S1(config-if)# end
S1#
```

Pokud si nejsme jistí nastavením, můžeme pomocí příkazu **show port-security interface** zobrazit aktuální nastavení zabezpečení portu. Na příkladu výpisu jde o port FastEthernet 0/1.

```
S1# show port-security interface f0/1
Port Security          : Enabled
Port Status             : Secure-shutdown
Violation Mode         : Shutdown
Aging Time              : 0 mins
Aging Type              : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses   : 1
Total MAC Addresses     : 0
Configured MAC Addresses : 0
Sticky MAC Addresses    : 0
Last Source Address:Vlan : 0000.0000.0000:0
Security Violation Count : 0
S1#
```

Všimněte si, jak je zabezpečení portů povoleno. Režim narušení (violation mode) je vypnutý a maximální počet MAC adres je 1.

Pokud je k portu připojeno zařízení, přepínač automaticky přidá MAC adresu zařízení jako zabezpečenou. V tomto příkladu není k portu připojeno žádné zařízení.

Poznámka: Pokud je aktivní port nakonfigurovaný příkazem **switchport port-security** a je k tomuto portu připojeno více než jedno zařízení, port se přepne do stavu **error-disabled state**.

Omezení a učení MAC adres

Chceme-li nastavit maximální počet povolených adres MAC na portu, použijeme příkaz:

Switchport port-security maximum, za nímž uvedeme číslici maximální počet bezpečných MAC adres.

Výchozí hodnota port security je 1. Maximální počet bezpečných MAC adres, které lze nataktovat, závisí na přepínači a IOS. V uvedeném případě je maximum 8192 MAC adres.

Při implementaci omezení a učení MAC adres lze přepínač nakonfigurovat tak, aby se MAC adresy na zabezpečeném portu učil jedním z těchto tří způsobů:

- 1. Manuální konfigurace:** Administrátor konfiguruje manuálně statické MAC adresy na daném portu
- 2. Naučeno dynamicky:** Když je vložen příkaz **switchport port-security**, aktuální zdrojová MAC adresa zařízení připojeného k portu je automaticky zabezpečena, ale není přidána do spuštěné konfigurace. Pokud se přepínač restartuje, port se bude muset znova MAC adresu zařízení naučit.
- 3. Naučeno dynamicky – Sticky:** Správce může povolit přepínači učit se MAC adresy dynamicky a „přilepit“ je do spuštěné konfigurace pomocí příkazu

Switch(config-if)# **switchport port-security mac-address sticky**

V rámci omezení a učení MAC adres je potřeba provést určitou sadu kroků.

Příklad na snímku ukazuje úplnou konfiguraci zabezpečení FastEthernet 0/1 portu.

```

S1(config)# interface fa0/1
S1(config-if)# switchport mode access
S1(config-if)# switchport port-security
S1(config-if)# switchport port-security maximum 4
S1(config-if)# switchport port-security mac-address aaaa.bbbb.1234
S1(config-if)# switchport port-security mac-address sticky
S1(config-if)# end
S1# show port-security interface fa0/1
Port Security          : Enabled
Port Status             : Secure-up
Violation Mode          : Shutdown
Aging Time              : 0 mins
Aging Type              : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses   : 4
Total MAC Addresses     : 1
Configured MAC Addresses: 1
Sticky MAC Addresses    : 0
Last Source Address:Vlan : 0000.0000.0000:0
Security Violation Count: 0
S1# show port-security address
      Secure Mac Address Table
-----
Vlan  Mac Address      Type           Ports  Remaining Age
                   (mins)
---  -----  -----
1    aaaa.bbbb.1234    SecureConfigured  Fa0/1   -
-----
Total Addresses in System (excluding one mac per port) : 0
Max Addresses limit in System (excluding one mac per port) : 8192
S1#

```

Správce určí maximálně 4 MAC adresy. Ručně nakonfiguruje jednu zabezpečenou jednu MAC adresu a poté nakonfiguruje port tak, aby získával další zabezpečené MAC adresy dynamicky až do maxima, čtyř zabezpečených MAC adres. K ověření konfigurace se používají příkazy **show port-security interface** a **show port-security address**.

Stárnutí zabezpečení portů (Port Security Aging)

Mimo samotných MAC adres je možné ovlivnit i čas jejich platnosti pomocí takzvaného Port security aging.

Stárnutí zabezpečení portu lze použít k nastavení doby platnosti pro statické a dynamické zabezpečené adresy na portu, přičemž na jeden port jsou podporovány dva typy stárnutí:

- **Absolute** - Zabezpečené adresy na portu jsou odstraněny po zadané době stárnutí.
- **Inactivity** - Zabezpečené adresy na portu jsou odstraněny, pokud jsou po určitou dobu neaktivní.

Pomocí stárnutí lze odebrat zabezpečené MAC adresy na zabezpečeném portu bez ručního odstranění stávajících zabezpečených adres MAC.

Stárnutí staticky konfigurovaných zabezpečených adres může být povoleno nebo zakázáno na základě jednotlivých portů.

Příkazem **switchport port-security aging** lze na zabezpečeném portu povolit nebo zakázat statické stárnutí nebo nastavit dobu stárnutí nebo typ.

Příkaz switchport port-security aging má parametry: {**static** | **time time** | **type {absolute | inactivity}**}

Static - umožňuje statické stárnutí nakonfigurované zabezpečené adresy na tomto portu.

`aging_time` - určuje dobu stárnutí pro tento port. Platný rozsah je od 0 do 1440minut. Pokud je čas roven 0, stárnutí je na portu deaktivován.

`Type` - nastaví typ stárnutí na absolutní nebo neaktivní. Pro absolutní stárnutí pro všechny zabezpečené adresy vyprší platnost po definované době (v minutách), pak jsou vyjmuty ze seznamu bezpečných adres.

Stárnutí pro neaktivní MAC adresy je spojeno s nulovým datovým provozem ze zdrojové zabezpečené adresy po definovanou dobu.

Režimy narušení zabezpečení portu

Režimy narušení zabezpečení portu nám zabezpečují reakci zařízení na nakonfigurovaném portu na výjimečnou situaci.

Pokud se MAC adresa zařízení připojeného k portu od seznamu zabezpečených adres liší, dojde k narušení stavu portu a port přejde do stavu *error-disabled*.

Nastavit režim *port security violation* lze příkazem:

switchport port-security violation {shutdown | restrict | protect}

Shutdown (je výchozí parametr)

Port okamžitě přejde do *error-disabled* stavu, vypne LED diodu portu a odešle zprávu do syslogu. Proces zvýší hodnotu čítače porušení. Pokud je zabezpečený port v *error-disabled* stavu, musí jej správce znova povolit zadáním sekvence příkazů `shutdown` a `no shutdown`.

Restrict

Port zahazuje pakety s neznámými zdrojovými adresami, dokud neodstraníte dostatečný počet definovaných zabezpečených MAC adres, aby klesl pod maximální hodnotu, nebo se nezvýší jejich maximální hodnota. Tento režim způsobí zvýšení čítače narušení zabezpečení a vygeneruje zprávu do syslogu.

Protect

Z režimů narušení zabezpečení je tento nejméně bezpečný. Port zahazuje pakety s neznámými zdrojovými MAC adresami, dokud není odstraněn dostatečný počet zabezpečených adres MAC, aby klesl pod maximální hodnotu, nebo se nezvýší maximální hodnota. Není však o tom do syslogu odeslána žádná zpráva.

Porty ve stavu *error-disabled*

Když je port vypnutý nebo nastavený do *the error-disabled* stavu, není na něm žádný přijmutý ani odeslaný provoz. Na konzoli se zobrazuje řada zpráv souvisejících se zabezpečením portů, jak ukazuje následující příklad.

Když se stav protokolu portu a spojení změní na *down*, LED dioda portu zhasne. Porty ve stavu *error-disabled* poznáme nejen podle LED diody, ale také z výpisu stavu portu.

Na příkladu příkazu **show interface** je stav portu identifikován jako **err-disabled**. Výstup příkazu **show port-security interface** nyní zobrazuje stav portu jako **secure-shutdown**. Čítač narušení zabezpečení se zvýší o 1.

```

S1# show interface fa0/18
FastEthernet0/18 is down, line protocol is down (err-disabled)
(output omitted)
S1# show port-security interface fa0/18
Port Security          : Enabled
Port Status             : Secure-shutdown
Violation Mode         : Shutdown
Aging Time              : 0 mins
Aging Type              : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses   : 1
Total MAC Addresses     : 1
Configured MAC Addresses : 1
Sticky MAC Addresses    : 0
Last Source Address:Vlan : c025.5cd7.ef01:1
Security Violation Count : 1
S1#

```

Správce by měl určit, co způsobilo narušení zabezpečení. Pokud je k zabezpečenému portu připojeno nepovolené zařízení, je bezpečnostní hrozba před opětovným povolením portu vyloučena.

Chcete-li port znovu povolit, nejprve použijte příkaz **shutdown** a poté použijte příkaz **no shutdown**.

Ověření zabezpečení portu (Port Security)

V provozu nastávají situace, kdy je porty třeba ověřit. Např. po konfiguraci zabezpečení portu na přepínači je vhodné zkontrolovat každé rozhraní a ověřit, zda je zabezpečení portu nastaveno správně a zkontrolovat, zda jsou statické MAC adresy správně nakonfigurovány.

Pro zobrazení nastavení zabezpečení portu přepínače použijeme příkaz **show port-security** s parametrem požadovaného rozhraní.

Pro zobrazení všech zabezpečených MAC adres, které jsou ručně konfigurovány nebo dynamicky naučeny na všech rozhraních přepínačů, lze použít příkaz **show port-security address**. Výpis je řazený podle VLAN.

Zmírnění útoků na VLAN

Cílem útoku DHCP starvation je, aby útočný nástroj, jako je např. Gobbler, vytvořil Denial of Service (DoS) pro připojení klientů.

Připomeňme, že útoky DHCP starvation lze účinně zmírnit pomocí zabezpečení portu, protože Gobbler používá pro každý odeslaný DHCP request jedinečnou zdrojovou MAC adresu. Snižování spoofingových DHCP útoků však vyžaduje větší ochranu.

Gobbler lze nakonfigurovat tak, aby použil pro Ethernetové rozhraní jako zdrojovou adresu skutečnou MAC adresu, ale vybral jinou Ethernetovou adresu z rozsahu DHCP. Tím by se zabezpečení portů stalo neúčinným, protože zdrojová MAC adresa by byla legitimní.

DHCP Snooping

Začneme pojmy, které jsou pro pochopení potřebné.

DHCP snooping filtrouje DHCP zprávy a omezuje rychlosť DHCP provozu na nedůvěryhodných portech.

- Zařízení pod kontrolou administrátora (např., přepínače, směrovače a servery) klasifikujeme jako důvěryhodné zdroje.
- Důvěryhodná rozhraní (např., trunk linky, porty serveru) musí být explicitně nakonfigurované jako důvěryhodné.
- Zařízení mimo síť a všechny přístupové porty jsou obecně považovány za nedůvěryhodné zdroje.

DHCP tabulka je vytvořena tak, že že zahrnuje zdrojovou MAC adresu zařízení na nedůvěryhodném portu a IP adresu přiřazenou DHCP serverem tomuto zařízení.

- MAC adresa a IP adresa jsou spojeny dohromady.
- Proto se tato tabulka nazývá tabulkou vazby DHCP snoopingu.

Kroky implementace DHCP Snoopingu

Následujícími čtyřmi kroky se DHCP snooping aktivuje:

Krok 1. Povolte DHCP snooping pomocí příkazu globální konfigurace **ip dhcp snooping**.

Krok 2. Na důvěryhodném portu použijte příkaz **ip dhcp snooping trust**.

Krok 3: Na nedůvěryhodném rozhraní omezte počet DHCP discovery zpráv, které je možné přijmout příkazem **ip dhcp snooping limit rate packets-per-second** v konfiguraci rozhraní.

Krok 4. Povolte DHCP snooping na VLAN nebo rozsahu VLAN příkazem **ip dhcp snooping vlan** v globální konfiguraci.

Zmírnění útoků na ARP

Inspekce dynamického ARP

Při typickém ARP útoku může útočník odesílat nevyžádané ARP odpovědi ostatním hostům v podsítí s MAC adresou útočníka a IP adresou výchozí brány. Aby se zabránilo ARP spoofingu a výslednému ARP poisoningu, musí přepínač zajistit, aby byly přenášeny pouze platné ARP požadavky a odpovědi.

Dynamic ARP inspection (DAI) pomáhá předcházet útokům ARP k tomu potřebuje spuštěný DHCP snooping. Jak jsme si před chvílí uvedli. Dosahuje toho pomocí následujících postupů a pravidel:

- Nepředává neplatné nebo bezdůvodné ARP Replies na další porty se stejně VLAN.
- Zachytává všechny ARP Requests a Replies na nedůvěryhodných portech.
- Ověřuje platnou vazbu IP-MAC u každého zachyceného paketu.
- Zahazuje a protokoluje ARP Replies přicházející od neplatných prvků, aby zamezil ARP poisoning.
- Jestliže nakonfigurovaný počet DAI u ARP paketů je překročen, rozhraní je nastaveno na Error-disabled.

Pokyny pro implementaci DAI

Ke zmírnění pravděpodobnosti ARP spoofingu a ARP poisoningu je třeba pro implementaci DAI provést následující kroky:

- Povolit globálně DHCP snooping.
- Povolit DHCP snooping na vybraných VLANách.
- Povolit dynamickou ARP inspekci na vybraných VLANách.
- Nakonfigurovat důvěryhodná rozhraní pro DHCP snooping a ARP inspection.

Obecně se doporučuje konfigurovat všechny porty přístupových přepínačů jako nedůvěryhodné a všechny uplinkové porty, které jsou připojeny k jiným přepínačům, konfigurovat jako důvěryhodné.

Příklad konfigurace DAI

- Dynamická ARP Inspekcí bude nakonfigurována tak, aby zmírňovala útoky ARP spoofing a ARP poisoning.
- DHCP snooping musí být povolený, protože DAI ho vyžaduje pro provoz tabulky vazeb DHCP snooping.
- Dále jsou pro počítače na dané VLAN povoleny DHCP snooping a ARP inspection.

```
S1(config)# ip dhcp snooping
S1(config)# ip dhcp snooping vlan 10
S1(config)# ip arp inspection vlan 10
S1(config)# interface fa0/24
S1(config-if)# ip dhcp snooping trust
S1(config-if)# ip arp inspection trust
```

Uplinkový port k routeru je důvěryhodný, a proto je nakonfigurován jako důvěryhodný pro snooping DHCP a ARP inspection.

Dynamickou ARP Inspekci lze také nakonfigurovat pro kontrolu cílových nebo zdrojových MAC a IP adres:

- **Destination MAC** - Zkontroluje cílovou MAC adresu v hlavičce Ethernetu proti cílové MAC adrese v těle ARP.
- **Source MAC** - Zkontroluje zdrojovou MAC adresu v hlavičce Ethernetu proti MAC adrese odesílatele v těle ARP.
- **IP address** - Zkontroluje, zda tělo ARP neobsahuje neplatné a neočekávané adresy IP, včetně adres 0.0.0.0, 255.255.255.255 a všech multicast IP adres.

Dále, příkaz **ip arp inspection validate {[src-mac] [dst-mac] [ip]}** z globální konfigurace je použitý k nastavení DAI, aby zahazoval ARP pakety, když IP adresy nejsou validní.

- Lze jej použít, když adresy MAC v těle paketů ARP neodpovídají adresám, které jsou uvedeny v záhlaví sítě Ethernet.
- Lze nakonfigurovat pouze jeden příkaz, zadání více příkazů **ip arp inspection validate** přepíše příkaz předchozí.
- Pro zahrnutí více než jedny metody ověření, je třeba je zadat na stejném příkazovém řádku.

Zmírnění útoků na STP

PortFast a BPDU Guard

Jedná se v základu o PortFast a BPDU Guard nastavení. Připomeňme, že síťoví útočníci mohou manipulovat protokolem Spanning Tree Protocol (STP), aby provedli útok spoofováním na root bridge zařízení a změnili topologii sítě.

Ke zmírnění útoků na STP se používá PortFast a Guard Protocol Data Unit (BPDU) Guard:

PortFast

PortFast okamžitě přivede port do stavu předávání ze stavu blokování a obejde stavy poslechu a učení. Je třeba ho aplikovat na všechny přístupové porty koncových uživatelů.

BPDU Guard

BPDU guard okamžitě deaktivuje port s chybou, který přijme BPDU. Stejně jako PortFast by měl být BPDU guard konfigurován pouze na rozhraních připojených ke koncovým zařízením.

Konfigurace PortFast

Jak již bylo naznačeno, PortFast obchází stavy naslouchání a učení STP, aby se minimalizoval čas, který musí přístupové porty čekat na konvergování STP. PortFast proto povolíme pouze na přístupových portech.

PortFast může být povolen:

Na rozhraní – příkazem **spanning-tree portfast** při konfiguraci rozhraní.

Globálně – Příkazem **spanning-tree portfast default** na všech přístupových portech.

Je jasné, že na spojích mezi přepínači může vytvořit smyčku spanning-tree loop, proto se pojí s konfigurací BPDU guard.

Pro ověření, že je PortFast globálně povolen lze použít příkazy:

- **show running-config | begin span**
- **show spanning-tree summary**

Pro ověření, zda je PortFast povolený na konkrétním rozhraní použijte příkaz **show running-config interface type/number**.

K ověření lze také použít příkaz **show spanning-tree interface type/number detail**.

Počítačové sítě 2

10, Koncepty WLAN

Název tématu	Cíl
Úvod do bezdrátových sítí	Popsat technologii a standardy WLAN.
Komponenty sítí WLAN	Popsat komponenty infrastruktury WLAN.
Provoz WLAN	Vysvětlit, jak bezdrátová technologie umožňuje provoz WLAN.
Provoz CAPWAP	Vysvětlit, jak WLC používá CAPWAP ke správě více AP.
Správa kanálů	Popsat správu kanálů v síti WLAN.
WLAN hrozby	Popsat hrozby pro WLAN.
Zabezpečení sítí WLAN	Popsat bezpečnostní mechanismy WLAN.

10.1 Úvod do bezdrátových sítí

Bezdrátové sítě mají bezesporu mnoho výhod

- Často je lze nasadit bez provádění náročných stavebních úprav
- Dovolují současné připojení klientů pobývajících v prostoru pokrytí signálem
- Bezdrátové infrastruktury se přizpůsobují rychle se měnícím potřebám a technologiím.
- Bezdrátová síť LAN (WLAN) je typ bezdrátové sítě, která se běžně používá v domácnostech, kancelářích a prostředích kampusu.
- Sítě WLAN umožňují mobilitu v domácím i obchodním prostředí.

Typy bezdrátových sítí

Nemáme jen jeden typ bezdrátových spojů, je potřeba pokrýt různé potřeby, proto máme k dispozici např. tyto varianty:

- **Wireless Personal-Area Network (WPAN)** – Nízký výkon a krátký dosah (do 10 metrů). Založeno na standardu IEEE 802.15 a frekvenci 2,4 GHz. Bluetooth a Zigbee jsou příklady WPAN.
- **Wireless LAN (WLAN) označuje** Středně velké sítě až do vzdálenosti 100 metrů. Založena je na standardu IEEE 802.11 a frekvenci 2,4 nebo 5,0 GHz.
- **Wireless MAN (WMAN) pokrývají** Velkou zeměpisnou oblast, jako je město nebo okres. Používají licencované frekvence.
- **Wireless WAN (WWAN)** – Rozsáhlá geografická oblast pro národní nebo globální komunikaci. Používá taktéž a výhradně licencované frekvence.

Bezdrátové technologie

K provedení bezdrátových spojů se využívají různé bezdrátové technologie, v licencovaných a bezlicenčních pásmech s malým všesměrovým vyzařováním, jako je třeba

Bluetooth, což je Standard IEEE WPAN používaný pro párování zařízení na vzdálenost až 300 stop (100 m).

Setkat se můžeme stále častěji s jeho variantou Bluetooth Low Energy (BLE), která podporuje zařízení ve smíšených (mesh) topologiích i rozsáhlých sítích.

Vedle toho Bluetooth Basic Rate/Enhanced Rate (BR/EDR) – podporuje topologie typu point-to-point a je optimalizován pro streamování zvuku.

Pak tu máme jako protipól

WiMAX (Worldwide Interoperability for Microwave Access), což je alternativní širokopásmové internetové připojení. Jde o standard IEEE 802.16 WLAN pro vzdálenosti až 30 mil (50 km).

Dalším zástupcem bezdrátových technologií je **Cellular Broadband**. Přenáší hlas i data. Používají jej telefony, automobily, tablety a notebooky.

- Global System of Mobile (GSM) – mezinárodně uznávané
- Code Division Multiple Access (CDMA) – primárně se používá v USA.

Satellite Broadband – používá směrovou satelitní anténu zaměřenou na satelit na geostacionární oběžné dráze. Vyžaduje přímou viditelnost. Obvykle se používá v odlehých a nedostupných oblastech, kde nejsou k dispozici kabelové spoje ani terestriální vysílače

802.11 Standardy

Aby bylo možné propojit zařízení různých výrobců a jejich různých technologií, je potřeba mít standardy, vesměs spadají pod IEEE 802.11 WLAN.

IEEE standard	Rádiová frekvence	Popis
802.11	2.4 GHz	Přenosové rychlosti až 2 Mb/s
802.11a	5 GHz	Přenosové rychlosti až 54 Mb/s Nespolupracuje s 802.11b nebo 802.11g
802.11b	2.4 GHz	Přenosové rychlosti až 11 Mb/s Delší dosah než 802.11a a lepší schopnost pronikat do stavebních konstrukcí
802.11g	2.4 GHz	Přenosové rychlosti až 54 Mb/s Zpětně kompatibilní s 802.11b
802.11n	2.4 a 5 GHz	Přenosové rychlosti 150 – 600 Mb/s Vyžadují více antén s technologií MIMO
802.11ac	5 GHz	Přenosové rychlosti 450 Mb/s – 1.3 Gb/s Podporuje až osm antén
802.11ax	2.4 a 5 GHz	High-Efficiency Wireless (HEW) Je schopen používat frekvence 1 GHz a 7 GHz

Rádiové frekvence

Pro lepší představu je zde obrázek elektromagnetického spektra. Ze symbolů je zřejmé, že existuje celá řada implementací. Jejich charakteristiky a tím i možnosti použití se mimo jiné odvíjejí od nosných frekvencí. Jak je z něj patrné, všechna bezdrátová zařízení fungují v rozsahu rádiových vln. Sítě WLAN pracují ve frekvenčních pásmech 2,4 a 5 GHz.

2.4 GHz (UHF – ultra high frequency, pro které máme český ekvivalent ultrakrátké vlny; Jde o frekvence od 300 MHz do 3 GHz, tj. vlnové délky v rozmezí od jednoho do deseti decimetrů). V tomto pásmu najdeme WLAN s normou 802.11b/g/n/ax.

5 GHz (SHF – super high frequency, tedy super krátké vlny; patří sem frekvence v rozsahu 3 až 30 GHz. Patří sem Radiolokace, radioreléové spoje, telekomunikace, satelitní spojení, [satelitní televize](#).) Neboť na 2,4GHz je dosti rušno a signál je značně znehodnocovaný všemožným rušením a interakcí s jinými zdroji jsou normy 802.11a/n/ac/ax postaveny na 5GHz, což s sebou přináší možnost vyšší rychlosti, které je ale vykoupeno horší schopností šíření signálu v členitém prostoru.

Lze říci, že čím vyšší frekvence, tím je možný vyšší objem přenášených dat, avšak menší dosah – pokrytí signálem. Dosah je otázkou vyzářeného výkonu, který nelze donekonečna zvyšovat. Existují technické a hygienické normy, které jsou hlídané úřady.

Mimo WirelesLAN existují bezdrátové spoje, realizované na jiných frekvencích, převážně již v licenčních pásmech, jak jsme si uvedli na přednáškách PSIT1.

Poznámka na okraj, k přenosu dat lze využít oblast infračerveného a viditelného spektra. Důležitou podmínkou je pro úspěšnou realizaci přímá viditelnost mezi komunikujícími body.

Organizace pro bezdrátové standardy

Interoperabilitu mezi zařízeními produkovanými různými výrobci zajišťují normy. Mezinárodně ovlivňují standardy WLAN tyto tři organizace:

- **International Telecommunication Union (ITU)**, ta reguluje přidělování rádiového spektra a satelitních drah.
- **Institute of Electrical and Electronics Engineers (IEEE)**, který určuje, jak je rádiová frekvence pro přenos informace modulována. Udržuje standardy pro místní a metropolitní sítě (MAN) s rodinou standardů IEEE 802 LAN/MAN.
- **Wi-Fi Alliance**. Podporuje růst a přijetí technologií sítí Wireless LAN. Jedná se o sdružení prodejců, jehož cílem je zlepšit interoperabilitu produktů založených na standardu 802.11.

10.2 Komponenty sítí WLAN

Mezi základní komponenty sítí WiFi LAN patří

- Bezdrátové síťové karty (pro koncová zařízení)
- Bezdrátový router (prvek koncentrující provoz bezdrátové sítě)
- Bezdrátový přístupový bod (koncový uzel používající bezdrátovou síťovou kartu)
- Autonomní a na řadiči založené přístupové body (zařízení s lokální nebo centrální správou)
- Antény a kabeláž (pasivní prvky definované fyzickými parametry a vlastnostmi, např. útlumem a ziskem)

Bezdrátové síťové karty

Pro bezdrátovou komunikaci mají notebooky, tablety, chytré telefony a dokonce i nejnovější automobily integrované bezdrátové síťové karty, které obsahují rádiový vysílač a přijímač. Pokud zařízení nemá integrovanou bezdrátovou síťovou kartu, lze použít bezdrátový adaptér napojitelný na sběrnici zařízení, např. na USB. Dnes jsou zařízení standardně WIFI technologií běžně vybavována.

Bezdrátové domácí routery

Domácí uživatel obvykle propojuje bezdrátová zařízení pomocí malého bezdrátového směrovače.

Bezdrátové směrovače slouží jako:

Přístupový bod - poskytuje bezdrátový přístup

Přepínač - slouží k propojení připojených zařízení

Router - poskytuje výchozí bránu do dalších sítí a do Internetu

Jejich výběr je značný. Vybírat lze podle různých parametrů v různých cenových kategoriích. Vybavenost a podpora posledních standardů dovoluje maximální využití standardů implementovaných do zařízení, vyšší rychlosť a menší výpadky, za což si je potřeba připlatit.

Bezdrátový přístupový bod

Bezdrátoví klienti používají své bezdrátové síťové karty k vyhledání blízkých přístupových bodů (AP). Klienti se poté pokusí přidružit a ověřit se na přístupovém bodu. Po ověření mají uživatelé bezdrátové sítě přístup k síťovým prostředkům.

Kategorie AP

Přístupové body lze kategorizovat jako autonomní nebo založené na řadiči.

- **Autonomní AP** jsou samostatná zařízení konfigurovaná prostřednictvím rozhraní příkazového řádku nebo grafického uživatelského rozhraní. Každý autonomní přístupový bod funguje nezávisle na ostatních a je konfigurován a spravován ručně správcem.
- **AP založené na řadiči** se někdy označují jako lehké AP (LAP). Ke komunikaci s řadičem LWAN (WLC) používají Lightweight Access Point protokol. Každý LAP je automaticky konfigurován a spravován právě prostřednictvím řadiče.

Bezdrátové antény

Mobilní zařízení mají antény skryté ve svém těle a tím je limitovaný jejich dosah. Síťové karty a samostatně stojící bezdrátová zařízení jsou vybavena konektory, na které lze připojit různé typy externích antén. Do základní nabídky základní patří:

- **Všesměrové** - poskytují 360 stupňové pokrytí. Ideální do domů a kancelářských prostor.
- **Sektorové** – poskytují pokrytí v prostoru definovaném úhlem, typicky 60 stupňů. Zabezpečí se tak pokrytí signálem jen v požadované oblasti. Soustava sektorových antén může pokrýt 360 stupňů s tím, že obslužení klienti jsou rozděleni do sektorů a je tak rozdělen prostor souboje klientů o připojení, signál a komunikační slot.
- **Směrové** - zaměřují rádiový signál určitým směrem. Příkladem je Yagi a parabolická anténa. Využívají se pro point-to-point spoje.
- **Více vstupů Více výstupů (MIMO)** je systém používající více antén (až osm) ke zvýšení šířky pásma.

10.3 Provoz WLAN

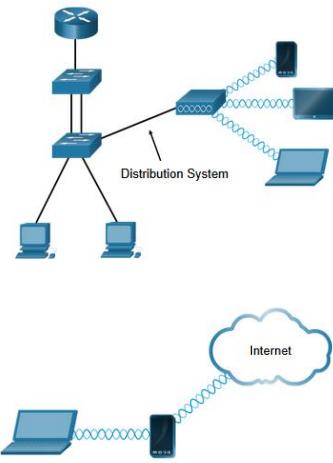
802.11 Režimy bezdrátové topologie

Všeobecně mezi nejpoužívanější režimy bezdrátové technologie patří:

Ad hoc režim - Slouží k připojení klientů způsobem peer-to-peer bez přítomnosti AP.



Infrastrukturní režim - Slouží k připojení klientů k síti pomocí AP.



Tethering je Variací topologie ad hoc, jde o situace, kdy je k vytvoření osobního hotspotu použit chytrý telefon nebo tablet s přístupem k datům. De facto může jím být i notebook připojený k Internetu kabelem nebo UBS/GSM modemem.

S problematikou infrastrukturního režimu se pojí zkratky BSS a ESS. Pro lepší pochopení principů, které popisují, jsou zde obrázky.

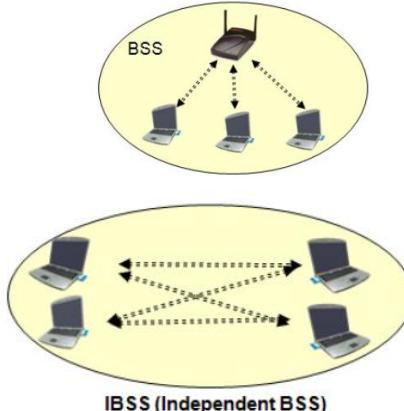
Infrastrukturní režim nám definuje dva topologické bloky:

V Basic Service Set (BSS) se

Používá jeden AP k propojení všech přidružených bezdrátových klientů.

Klienti v různých BSS nemohou spolu navzájem komunikovat.

Variantou je pak **Independed Basic Service Set (IBSS)**, který nepoužívá AP, propojení bezdrátových klientů je realizováno ad-hoc ve dvojicích v daném čase.

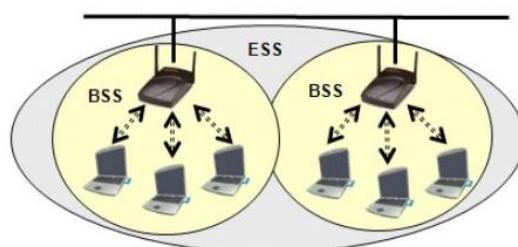


Extended Service Set (ESS)

Spojení dvou nebo více BSS kabelovým distribučním systémem.

Klienti v propojených BSS mohou prostřednictvím ESS komunikovat.

Systém uvedených bloků je postavený na protokolech a principech popsaných dále.



DS, WDS a MESH

Důležitou podmínkou pro propojení více Wi-Fi buněk (BSS) do jedné Wi-Fi sítě (ESS) je jejich vzájemné propojení.

K tomu nám slouží **Distributed system (DS)**, mezi jehož úkoly patří zejména přenos dat mezi jednotlivými buňkami a spolupráce na autentizaci a asociaci koncových stanic s jednotlivými přístupovými body.

Podílí se také na koordinaci přenosu dat mezi sítěmi, k čemuž slouží tzv. portál. K propojení BSS se běžně využívá jedné ze tří variant řešení:

Distributed system (DS), kde je přenos dat mezi jednotlivými buňkami realizovaný prostřednictvím fyzické kabeláže.

Wireless Distributed system (WDS) K propojení BSS využívá bezdrátový přenos a prvky, kterým říkáme opakovače (repeatery). V praxi je např. používáno pro propojení LAN segmentů.

MESH je kombinované propojení více BSS s řetězením opakovačů do velké sítě.

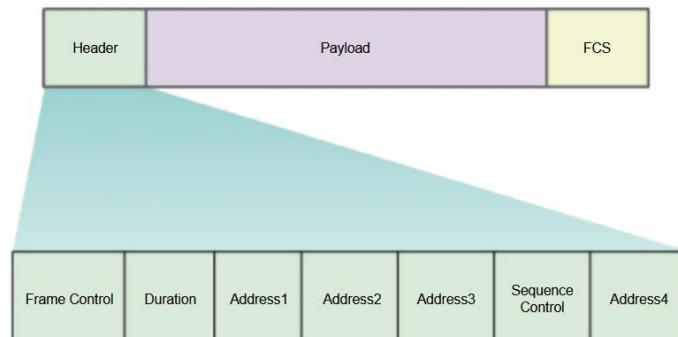
Struktura rámce 802.11

Formát rámce 802.11 je podobný formátu rámce Ethernet, kromě toho, že obsahuje více polí. MAC hlavička + tělo rámce (max. 2312B) + CRC

MAC hlavička obsahuje:

Frame Control (2B)

- typ rámce
(management/control/data) + podtyp (např. RTS, CTS, ...)
- indikace příjmu z/vysílání do distribučního systému
- more fragments flag
- indikace retransmise fragmentu
- power management
- indikace použití WEP



o Duration: pro inzerování doby obsazení média

o Station ID: pro funkci úspory energie

- Adresní pole 1-4: 4 x 6B (zdroj, cíl a další podle hodnoty ve Frame Control)
- Sequence Control (2B) - podpora fragmentace

CSMA/CA

Vzduch je prostředí, v němž nelze jednoduše vytvořit nespočet spojů i zde je třeba řídit, kdo smí v kterém okamžiku komunikovat. Tento rys je společný pro Wi-Fi i sdílený Ethernet.

Prostředí, kde se signál šíří je ale odlišné. Detekce obsazenosti na metalickém kabelu a wifi kanálu je odlišná. Klient zde nemusí komunikaci jiného klienta „slyšet“.

Navíc, síť WLAN jsou polo duplexní. Klient během vysílání nemůže „poslouchat“, což znemožňuje detektovat kolizi.

Síť WLAN používají k určení toho, jak a kdy odeslat data Vícenásobný přístup s detekcí nosiče a zabráněním kolizím (CSMA/CA).

Bezdrátový klient provádí následující kroky:

1. Poslouchá kanál, aby zjistil, zda je nečinný, tj. na kanálu není žádný jiný provoz.
2. Odešle zprávu „ready to send“ (RTS) na AP, čímž požádá o vyhrazený přístup k síti.
3. Přijme zprávu „clear to send“ (CTS) od AP, která přiděluje přístup k odeslání.
4. Pokud není přijata žádná zpráva CTS, počká náhodný časový interval před opakováním procesu.
5. Klient může přenášet data.
6. Po skončení potvrzuje všechny přenosy. Pokud bezdrátový klient neobdrží potvrzení, předpokládá kolizi a opakuje proces.

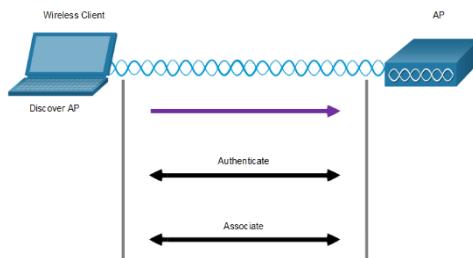
Přidružení bezdrátového klienta a AP

Aby bezdrátová zařízení mohla komunikovat přes síť, musí se nejprve spojit s přístupovým bodem nebo bezdrátovým směrovačem. K tomu bezdrátová zařízení provádějí následující třífázový proces:

Discover – najde bezdrátový AP

Authenticate – ověří se pomocí AP

Associate – přidruží se k AP



Přidružení bezdrátového klienta a AP

K dosažení úspěšného přidružení se musí bezdrátový klient a AP dohodnout na konkrétních parametrech:

SSID (Service Set Identifier) – klient potřebuje znát název sítě pro připojení.

Password – heslo je vyžadováno, aby se klient autentizoval k přístupovému bodu.

Network mode – použitý 802.11 standard.

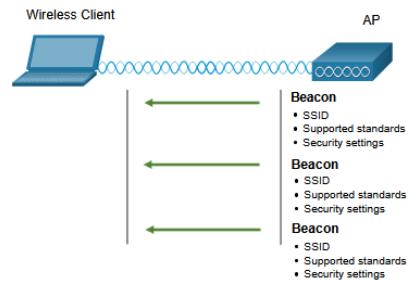
Security mode – nastavení parametrů zabezpečení, tj. WEP, WPA, nebo WPA2.

Channel settings – nastavení kanálu – použitá frekvenční pásma – může být i automat

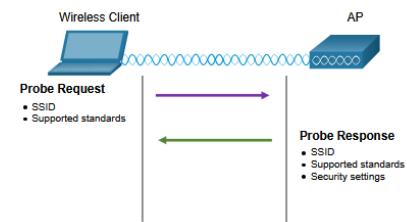
Pasivní a aktivní Discover Mode

Discover Mode je proces, během něhož se Bezdrátoví klienti připojují k AP buď pomocí pasivního, nebo aktivního procesu skenování (sondování).

V Pasivním režimu AP otevřeně inzeruje svou službu pravidelným zasíláním vysílačích rámců obsahujících SSID, podporované standardy a nastavení zabezpečení.



V Aktivním režimu Bezdrátoví klienti musí znát název SSID. Bezdrátový klient proto zahájí proces vysláním sondovacího rámce požadavku na více kanálech (skrýt SSID).

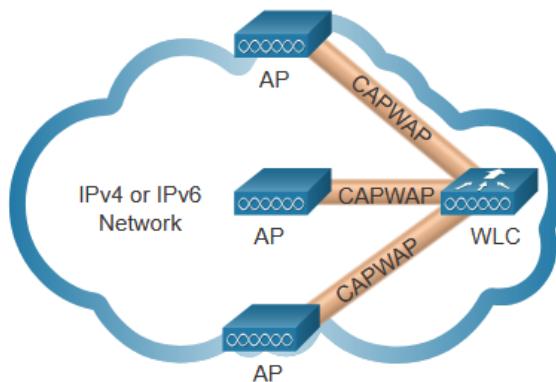


10.4 Provoz CAPWAP

Centrálně spravované bezdrátové sítě v rámci enterprise řešení nejsou ojedinělé, proto se nyní na jeden takový model podíváme.

Úvod do CAPWAP

CAPWAP neboli **Control And Provisioning of Wireless Access Points** je standardní protokol IEEE jehož principy jsou popsány v RFC 5415. Umožňuje použití wireless LAN controlleru – WLC, který je určený ke správě více AP a WLAN. Je založený na Lightweight Access Point Protocolu (LWAPP), ale přidává k němu další zabezpečení a to Datagram Transport Layer Security (DTLS). CAPWAP při své činnosti zapouzdřuje a předává klientský provoz WLAN mezi AP a wireless LAN kontroleru přes tunely pomocí UDP portů 5246 a 5247. Funguje na protokolech IPv4 i IPv6. V případě IPv4 používá UDP protokol a na IPv6 používá UDP lite protokol.



Architektura Split MAC

Koncept CAPWAP split MAC provádí všechny funkce běžně prováděné jednotlivými AP a rozděluje je mezi dvě funkční komponenty:

- AP MAC funkce
- WLC MAC funkce

AP MAC funkce	WLC MAC funkce
Odezvy majáků a sond	Ověřování
Potvrzování paketů a opakování přenos	Přidružení a opětovné přidružení roamingových klientů
Řazení rámců a prioritizace paketů	Překlad rámců do jiných protokolů
Šifrování a dešifrování dat MAC vrstvy	Ukončení provozu 802.11 na kabelovém rozhraní

Šifrování DTLS

Při provozu CAPWAP se používá DTLS šifrování, neboli Datagram Transport Layer Security, které poskytuje zabezpečení mezi AP a Wireless LAN kontrolérem.

- Ve výchozím nastavení je povoleno zabezpečit řídicí kanál CAPWAP a zašifrovat veškerou správu a řízení provozu mezi AP a WLC.
- Šifrování dat je ve výchozím nastavení zakázáno a vyžaduje, aby byla na WLC nainstalována licence DTLS, než jej bude možné povolit na AP.

Flex Connect Apps

FlexConnect je řešení které umožňuje konfigurovat a řídit přístupové body v pobočkách nebo vzdálených kancelářích z podnikové sítě prostřednictvím WAN spoje bez nasazení řadiče v každé kanceláři. Přístupové body s FlexConnect mohou lokálně řídit datový provoz a provádět autentizaci klienta, i když dojde ke ztrátě jejich připojení k řadiči. Když jsou k řadiči připojené, mohou také odesílat data zpět do řadiče. V připojeném režimu může přístupový bod FlexConnect provádět také místní ověřování.

Jinými slovy, FlexConnect je aplikace řízení vzdálených WiFi přístupů a nabízí pro AP dva režimy:

Připojený režim - WLC je dosažitelný. FlexConnect AP má konektivitu s WLC přes CAPWAP tunel. WLC provádí všechny CAPWAP funkce.

Samostatný režim - WLC je nedosažitelný. Jde o situaci, kdy FlexConnect AP ztratil CAPWAP připojení s WLC. FlexConnect AP může převzít některé z funkcí WLC, jako je místní přepínání datového provozu klienta a místní ověřování klienta.

12.5 Správa kanálů

Pro správné a bezchybné fungování WI-FI sítí je správná konfigurace kanálů.

Frequency Channel Saturation

Proč vlastně se využívají kanály. Ze současného rozvoje bezdrátových technologií a jejich masivního nasazení je vcelku jasné, že požadavek na připojení těchto zařízení do datových sítí je enormní, a že nelze všechna zařízení obsloužit na jednom frekvenčním kanále. Z toho důvodu přidělený kmitočet dělíme do tzv. kanálů. Kanály se navzájem překrývají, a proto je třeba volit takovou kombinaci, aby se navzájem nerušily.

V praxi to potom znamená, že aby bylo možné provozovat různé bezdrátové sítě ve vzájemné blízkosti či s jistým překryvem, je nutné toto nějakým způsobem řídit.

Pokud je poptávka po konkrétním bezdrátovém kanálu příliš vysoká, může dojít k jeho přesycení, což zhorší kvalitu komunikace.

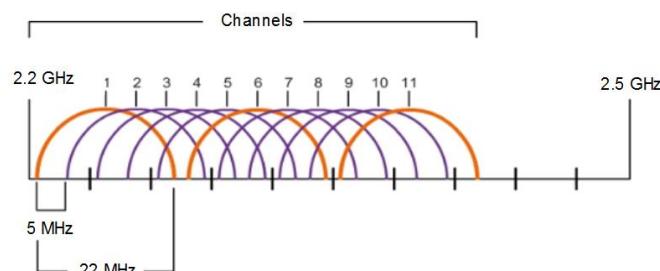
Saturaci kanálu lze zmírnit pomocí technik, které kanály využívají efektivněji.

- **Direct-Sequence Spread Spectrum (DSSS)** - Modulační technika určená k šíření signálu ve větším frekvenčním pásmu. Je používaná zařízeními 802.11b, aby se zabránilo rušení jinými zařízeními používajícími stejnou frekvenci 2,4 GHz.

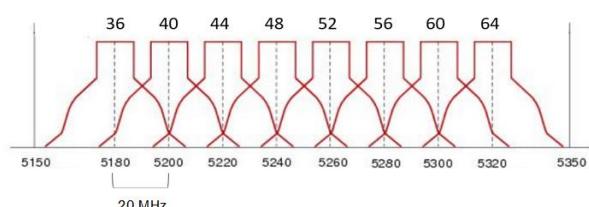
- **Frequency-Hopping Spread Spectrum (FHSS)** - Přenáší rádiové signály rychlým přepínáním nosného signálu mezi mnoha frekvenčními kanály. Odesílatel a příjemce musí být synchronizováni, aby „věděli“, na který kanál skočit. Používá se původním standardem 802.11.
- **Orthogonal Frequency-Division Multiplexing (OFDM)** – je podmnožina multiplexování s frekvenčním dělením, ve kterém jeden kanál používá více subkanálů na sousedních frekvencích. OFDM je používán řadou komunikačních systémů, včetně 802.11a/g/n/ac.

Výběr kanálu

Pásma 2,4 GHz je rozděleno do několika kanálů, z nichž každý má přidělenou šířku pásmá 22 MHz a je oddělen od dalšího kanálu 5 MHz. Nejlepším postupem pro síť WLAN 802.11b/g/n vyžadující více přístupových bodů je použití nepřekrývajících se kanálů, například 1, 6 a 11, jak můžeme vypozorovat na obrázku.



Podobně je to i v případě frekvence 5GHz, kde se používají standardy 802.11a/n/ac, kde existuje 24 kanálů. Každý kanál je od dalšího kanálu oddělen 20 MHz.



Nepřekrývající se kanály jsou 36, 48 a 60.

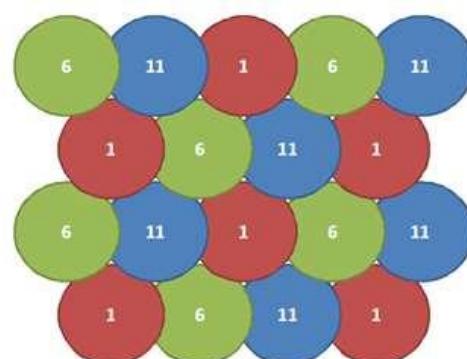
Plán nasazení WLAN

Pro nasazení WLAN se vypracovávají plány, které vycházejí z reálného rozložení staveb a naměřených útlumů.

Důležitý je též parametr počtu uživatelů podporovaných sítí WLAN, který závisí na následujících skutečnostech:

- Prostorové uspořádání budovy
- Počet uživatelů a zařízení, která se vejdu do prostoru
- Datové rychlosti, které uživatelé očekávají
- Použití nepřekrývajících se kanálů více AP a nastavení vysílacího výkonu

Při plánování umístění AP je uvažována přibližně kruhová oblast pokrytí.



10.6 WLAN hrozby

Přehled zabezpečení bezdrátové sítě

Síť WLAN je otevřená komukoli v dosahu přístupového bodu, s příslušnými přihlašovacími údaji a požadovanými k přidružení.

Útoky mohou být generovány cizími osobami, nespokojenými zaměstnanci a dokonce i neúmyslně zaměstnanci. Bezdrátové sítě jsou specificky náchylné k několika hrozbám, včetně následujících:

- Odposlech dat
- Bezdrátoví vetřelci
- Útoky DoS (Denial of Service)
- Cizí (Rogue) AP

Útoky DoS

Bezdrátové útoky DoS mohou být výsledkem následujících akcí:

- Nesprávně nakonfigurovaná zařízení
- Zlomyslný uživatel s nekalými úmysly záměrně zasahující do bezdrátové komunikace
- Náhodné rušení

Chcete-li minimalizovat riziko útoku DoS v důsledku nesprávně nakonfigurovaných zařízení a škodlivých útoků, je třeba zabezpečit všechna zařízení, udržovat bezpečná hesla, vytvářet zálohy a zajistit, aby byly všechny změny konfigurace realizovány mimo pracovní dobu.

Cizí Přístupové body

V celku typickou hrozbou je cizí APOD. Jde o AP nebo bezdrátový router, který byl připojen k podnikové síti bez výslovného povolení a v rozporu s podnikovou politikou. Po připojení může být cizí AP zneužit útočníkem k zachycení MAC adres, zachycení datových paketů, získání přístupu k síťovým prostředkům nebo k zahájení útoku typu man-in-the-middle. Jako podvržený přístupový bod lze také použít osobní síťový hotspot. Například uživatel se zabezpečeným síťovým přístupem umozní, aby se jeho autorizovaný operační systém Windows stal přístupovým bodem Wi-Fi. Aby se zabránilo instalaci podvržených přístupových bodů, musí organizace nakonfigurovat WLC se zásadami proti cizím AP a pomocí monitorovacího softwaru aktivně monitorujícího rádiové spektrum pro detekci neoprávněných přístupových bodů.

Útok typu Man-in-the-Middle

Při útoku typu man-in-the-middle (MITM) je zařízení hackera umístěno mezi dvě legitimní entity, aby mohl číst nebo upravovat data, která procházejí mezi oběma stranami. Populární bezdrátový útok MITM se nazývá útok „zlé dvojče (evil twin) AP“, kdy útočník zapojí podvržený AP a nakonfiguruje jej se stejným SSID jako legitimní AP.

Překažení útoku MITM začíná identifikací legitimních zařízení v síti WLAN. K tomu musí být uživatelé autentizováni. Poté, co jsou známa všechna legitimní zařízení, lze síť monitorovat kvůli abnormálním zařízením nebo provozu.

10.7 Zabezpečení sítí WLAN

V posledním bloku se zaměříme na možnosti maskování SSID, filtrování MAC adres a systémy pro ověřování a šifrování.

K zabránění v přístupu bezdrátovým útočníkům a pro ochranu dat byly využívány a stále jsou k dispozici na většině směrovačů a AP dvě původní bezpečnostní funkce:

- **Maskování SSID** Přístupové body a některé bezdrátové směrovače umožňují zakázat rámec majáku SSID. Bezdrátoví klienti musí být pro připojení k síti ručně konfigurováni pomocí SSID.
- **Filtrování MAC adres** znamená, že Správce může ručně povolit nebo zakázat klientům bezdrátový přístup na základě jejich fyzické hardwarové MAC adresy. V praxi to znamená, že router je nakonfigurován tak, aby umožňoval připojit zařízení s definovanými MAC adresy. Zařízení s odlišnými MAC adresami se nebudou k 2,4 GHz WLAN moci připojit.

Originální metody ověřování ve standardu 802.11

Nejlepší způsob, jak zabezpečit bezdrátovou síť, je používat systémy ověřování a šifrování. Standard 802.11 obsahuje originální metody ověřování. S výchozím standardem byly zavedeny dva typy ověřování:

Otevřené ověřování systému

- Není vyžadováno žádné heslo. Obvykle se používá k poskytování bezplatného přístupu k internetu ve veřejných prostorách, jako jsou kavárny, letiště a hotely.
- Klient je odpovědný za zajištění bezpečnosti, například prostřednictvím VPN.

Ověření sdíleným klíčem

- Poskytuje mechanismy, jako jsou WEP, WPA, WPA2 a WPA3 k ověřování a šifrování dat mezi bezdrátovým klientem a přístupovým bodem. Pro připojení však musí být heslo předem sdíleno mezi oběma stranami. Možnosti si popíšeme na následujícím snímku.

Metody ověřování sdíleným klíčem

V současné době jsou k dispozici čtyři techniky ověřování pomocí sdíleného klíče, jak je uvedeno v tabulce:

Metoda ověřování	Popis
Wired Equivalent Privacy (WEP)	Originální specifikace 802.11 navržená k zabezpečení dat pomocí šifrovací metody Rivest Cipher 4 (RC4) se statickým klíčem. WEP se již nedoporučuje a nikdy by se neměl používat.
Wi-Fi Protected Access (WPA)	Standard Wi-Fi Alliance, který používá WEP, zajišťuje data pomocí mnohem silnějšího šifrovacího algoritmu tzv. Temporal Key Integrity Protocol (TKIP). TKIP mění klíč pro každý paket, takže je mnohem obtížnější jej hackovat.
WPA2	K šifrování používá Advanced Encryption Standard (AES). AES je v současné době považován za nejsilnější šifrovací protokol.
WPA3	Jde o novou generaci zabezpečení Wi-Fi. Všechna zařízení s povoleným WPA3 používají nejnovější metody zabezpečení, zakazují zastaralé starší protokoly a vyžadují použití chráněných rámců správy (Protected Management Frames - PMF).

Ověření domácího uživatele

Pro objasnění metody lze uvést příklad nastavení domácího AP/wifi Routeru.

SOHO směrovače mají pro ověřování obvykle dvě možnosti: WPA a WPA2, přičemž WPA 2 má dvě metody ověřování.

- **Osobní** - Určeno pro domácí nebo malé kancelářské sítě, uživatelé se autentizují pomocí předem sdíleného klíče (PSK). Bezdrátoví klienti se ověřují pomocí bezdrátového směrovače pomocí předem sdíleného hesla. Není vyžadován žádný speciální ověřovací server.
- **Firemní** - Určeno pro podnikové sítě. Vyžaduje ověřovací server RADIUS (Remote Authentication Dial-In User Service). Zařízení musí být ověřeno serverem RADIUS a uživatelé se pak musí ověřit pomocí standardu 802.1X, který k ověřování používá protokol EAP (Extensible Authentication Protocol).

Metody šifrování

O možnostech šifrování jsme již hovořili. V běžné situaci máme k dispozici WPA a WPA2. Ty zahrnují dva šifrovací protokoly:

- **Temporal Key Integrity Protocol (TKIP)** – Používán WPA a poskytuje podporu pro starší zařízení WLAN. Využívá WEP, ale šifruje užitečné zatížení 2. vrstvy pomocí TKIP.
- **Advanced Encryption Standard (AES)** – Používán WPA2 a používá režim Counter Cipher Mode s Block Chaining Message Authentication Code Protocol (CCMP), který umožňuje cílovým hostitelům rozpoznat, zda byly šifrované a nešifrované bity změněny.

Autentizace ve firemním prostředí

Režim firemního (Enterprise) zabezpečení vyžaduje nasazení RADIUS serveru pro jednotné, centrální ověřování, autorizaci a účtování (AAA).

K tomu jsou vyžadovány určité informace pro provedení nastavení AP:

- **RADIUS server IP address** – IP adresa serveru.
- **UDP port numbers** – typicky se používají UDP porty 1812 pro RADIUS Authentication a 1813 pro RADIUS Accounting. Mohou být ale použity i UDP porty 1645 a 1646.
- **Sdílený klíč** je znakový řetězec, který se používá k ověření AP u serveru RADIUS.

WPA3

Protože WPA2 již není považován za bezpečný, doporučuje se implementovat tak, kde je to technicky možné WPA3 a pokud je k dispozici. WPA3 zahrnuje čtyři funkce:

- **WPA3 – Personal** : Zabraňuje útokům hrubou silou pomocí Simultaneous Authentication of Equals (SAE).
- **WPA3 – Enterprise** : Používá ověřování 802.1X / EAP. Vyžaduje však použití 192bitové kryptografické sady a eliminuje míchání bezpečnostních protokolů pro předchozí standardy 802.11.
- **Open Networks** : Nepoužívá žádné ověřování. K šifrování veškerého bezdrátového provozu však používá oportunistické bezdrátové šifrování (OWE).
- **IoT Onboarding** : Používá protokol Device Provisioning Protocol (DPP) k rychlému připojení zařízení IoT.

Počítačové sítě 2

11, Konfigurace WLAN

Název tématu	Cíl
Konfigurace vzdálené WLAN	Nakonfigurovat síť WLAN tak, aby podporovala vzdálené místo.
Konfigurace základní WLAN na WLC	Nakonfigurovat WLC WLAN za použití rozhraní pro správu a ověřování WPA2 PSK.
Konfigurace WPA2 podnikové WLAN na WLC	Nakonfigurovat WLC WLAN za použití VLAN DHCP serveru a firemní WPA2 ověřování.
Odstraňování problémů s WLAN	Odstranit běžné problémy s konfigurací bezdrátové sítě.

Blok je zaměřený na konfiguraci bezdrátových LAN v SOHO prostředí i v podnikové infrastruktuře.

11.1 Konfigurace vzdálené WLAN

Problematikou diskutovanou v této části je:

- Použití webové stránky bezdrátového směrovače ke konfiguraci
- Provedeme Změnu hesla
- Změny v nastavení WAN a LAN
- Připojení bezdrátové sítě

Bezdrátový směrovač

Vzdálení pracovníci, malé pobočky a domácí sítě často používají směrovač pro malou kancelář či domácnost což bývá často označováno jako SOHO (Small Office Home Office) zařízení.

- „Integrované“ směrovače obvykle zahrnují přepínač pro kabelové klienty, port pro připojení k internetu (někdy označovaný jako „WAN“) a bezdrátové komponenty pro bezdrátový přístup klientů.
- Zmíněné bezdrátové směrovače obvykle poskytují zabezpečení WLAN, služby DHCP, integrovaný překlad adres (NAT), kvalitu služeb (QoS) a celou řadu dalších funkcí.
- Sada nabízených funkcí se bude lišit podle modelu směrovače.

Pokud jde o konfiguraci kabelového nebo DSL modemu, obvykle se provádí zástupce poskytovatele služeb na místě, na dálku nebo je zařízení dodané již předkonfigurované a stačí jej jen připojit.

Přihlášení k bezdrátovému směrovači

Prvním krokem po zapojení bezdrátového směrovače je přihlášení k jeho konfiguračnímu uživatelskému rozhraní.

Většina bezdrátových směrovačů je předem nakonfigurována pro připojení k dané síti a poskytování služeb.

- Výchozí IP adresy, uživatelská jména a hesla bezdrátových směrovačů lze snadno najít na internetu.
- Proto by, z bezpečnostních důvodů, měla být vaší prioritou změna těchto výchozích hodnot.

Chcete-li získat přístup ke konfiguračnímu grafickému uživatelskému rozhraní bezdrátového směrovače

- Otevřete webový prohlížeč a zadejte výchozí IP adresu bezdrátového směrovače.
- Výchozí IP adresu lze nalézt v dokumentaci dodané s bezdrátovým směrovačem alternativně ji lze vyhledat na Internetu.
- Slovo **admin** se velice často používá jako výchozí uživatelské jméno a heslo.

Základní nastavení sítě

Následným krokem při konfiguraci je provedení základního nastavení, které zahrnuje tyto kroky:

- Přihlaste se ke směrovači z webového prohlížeče.
- Změňte výchozí heslo pro správu.

- Přihlaste se pomocí nového správcovského hesla.
- Změňte výchozí IPv4 adresy pro DHCP pool.
- Obnovte IP adresu PC, ze kterého provádít konfiguraci a...
- Přihlaste se ke směrovači pomocí nové IP adresy.

Základní nastavení bezdrátové sítě

Základní nastavení bezdrátové sítě pak obnáší následující kroky:

- Zobrazte výchozí nastavení sítě WLAN.
- Změňte režim sítě - určete, který standard 802.11 má být implementován.
- Nakonfigurujte SSID.
- Nakonfigurujte kanál a ujistěte se, že se nepoužívají žádné překrývající se kanály.
- Nakonfigurujte režim zabezpečení výběrem z možností Otevřít (Open), WPA, WPA2 Osobní (Personal), WPA2 Enterprise atd.
- Nakonfigurujte přístupovou frázi, jak je požadováno pro vybraný režim zabezpečení.

Nastavení bezdrátové mesh sítě

Pokud chcete rozšířit dosah na více než přibližně 45 metrů v interiéru a 90 metrů venku, vytvořte bezdrátovou síť typu MESH. Vytvoříte ji přidáním přístupových bodů se stejným nastavením, avšak s výjimkou použití různých kanálů, dle tabulky překrytí, abyste zabránili rušení. Technologie WDS je stále na zařízeních k dispozici, ale jejich implementací ubývá na úkor MESH, neboť rozšíření bezdrátových LAN v malé kanceláři nebo domácnosti je velmi snadné. Výrobci zjednodušili (zaautomatizovali) vytváření bezdrátové mesh sítě (WMN) prostřednictvím aplikací pro smartphony.

NAT pro IPv4

Bezdrátovému směrovači může být poskytovatelem internetu přiřazena veřejná adresa, ale není to pravidlem. Pro adresování v síti LAN se standardně používají soukromé síťové adresy.

Aby bylo zařízením v síti LAN umožněno komunikovat s vnějším světem, použije směrovač proces zvaný Překlad síťových adres (NAT). NAT překládá soukromou (místní) zdrojovou adresu IPv4 na veřejnou (globální) adresu (proces je u příchozích paketů obrácen). NAT umožňuje sdílení jedné veřejné adresy IPv4 sledováním čísel zdrojových portů pro každou relaci vytvořenou zařízením. Pokud má váš ISP povolený IPv6 protokol, uvidíte pro každé zařízení jedinečnou adresu IPv6. Na vnějším rozhraní můžete mít od providera přidělenou i privátní IP adresu. V popsaném schématu pak půjde o několikanásobný NAT.

Kvalita služeb - Quality of Service

Většina dnešních bezdrátových směrovačů má možnost konfigurace Kvality služeb (QoS).

Toto konfigurací můžete zaručit, že určité typy provozu, například hlas a video, budou mít přednost před provozem, který není tak citlivý na čas, jako je e-mail či procházení webu. U některých bezdrátových směrovačů lze definovat prioritu provozu na jeho konkrétních portech. U některých můžete i definovat šířku pásma a zredukovat tak tok dat a to oddělenou konfigurací pro download či upload.

Přesměrování portů - Port Forwarding

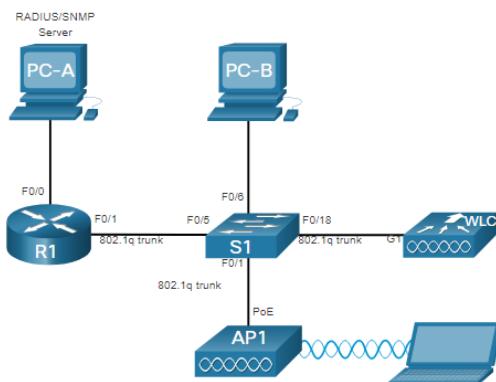
Bezdrátové směrovače dokáží blokovat definované TCP a UDP porty, aby tak zabránily neoprávněnému přístupu do LAN.

Existují však situace, kdy je nutné otevřít konkrétní porty, aby určité programy a aplikace mohly komunikovat se zařízeními v různých sítích. Přesměrování portů je metoda směrování provozu mezi zařízeními v samostatných sítích založená na pravidlech. Spouštění portů (Port triggering) umožnuje směrovači dočasně předávat data přes příchozí porty do konkrétního zařízení. Spouštění portů můžete použít k předávání dat do počítače pouze v případě, že se k provedení odchozího požadavku použije určený rozsah portů.

11.2 Konfigurace základní WLAN na WLC

Topologie WLAN

Topologie a schéma adresování použité v následujícím příkladu jsou zobrazeny na obrázku a v tabulce.



Device	Interface	IP Address	Subnet Mask	
R1	F0/0	172.16.1.1	255.255.255.0	
R1	F0/1.1	192.168.200.1	255.255.255.0	
S1	VLAN 1	DHCP		
WLC	Management	192.168.200.254	255.255.255.0	
AP1	Wired 0	192.168.200.3	255.255.255.0	
PC-A	NIC	172.16.1.254	255.255.255.0	
PC-B	NIC	DHCP		
Wireless Laptop	NIC	DHCP		

Zde použitý přístupový bod je řízený řadičem WLC a na rozdíl od autonomního AP nevyžaduje žádnou počáteční konfiguraci. Bývá nazýván lehkým AP (LAP). Ten používají ke komunikaci s řadičem WLAN (WLC) protokol Lightweight Access Point Protocol (LWAPP). AP založené na řadiči jsou výhodné v situacích, kdy je v síti vyžadováno použití mnoha AP. Jak jsou do sítě připojovány další AP, jsou automaticky konfigurované a spravované z řadiče.

Přihlášení k WLC

Konfigurace řadiče bezdrátové sítě LAN (WLC) se příliš neliší od konfigurace samostatného bezdrátového směrovače. WLC řídí přístupové body, poskytuje více služeb a možností správy.

Uživatel se přistupuje k řadiči pomocí pověření, která byla nakonfigurována během počátečního nastavení. Webová stránka **Souhrn sítě (Network Summary)** je řídicí panel, který poskytuje rychlý přehled nakonfigurovaných bezdrátových sítí, přidružených přístupových bodů (AP) a aktivních klientů. Je zde také možno vidět počet podvržených přístupových bodů a klientů.

Zobrazení informací o AP

Kliknutím na **Přístupové body** (Access Points) v levé nabídce se zobrazí celkový přehled systémových informací a výkonu AP. Když je v síti aktivní protokol Cisco Discovery Protocol (CDP), WLC ví, že přístupový bod je připojen ke konkrétnímu portu na přepínači.

Mezi AP této kategorie patří např. Cisco Aironet 1815i, pro jehož konfiguraci lze použít příkazový řádek a omezenou sadu známých příkazů IOS.

Pokročilé nastavení

Většina řadičů je vybavena základními nastaveními a nabídkami, ke kterým mohou uživatelé rychle přistupovat a realizovat řadu běžných konfigurací. Správce sítě však často potřebuje přístup k pokročilým nastavením. U AP Cisco 3504 se lze dostat na stránku Souhrn (Summary) kliknutím na **Pokročilé (Advanced)** v pravém horním rohu.

Odtud je pak přístup ke všem funkcím WLC.

Konfigurace WLAN

Řadiče bezdrátové sítě LAN mají porty přepínače vrstvy 2 a virtuální rozhraní, která jsou vytvořena softwarově a jsou velmi podobná rozhraním VLAN. Každý fyzický port může podporovat mnoho AP a bezdrátových LAN. Porty na řadiči jsou v podstatě trunk porty, které mohou přenášet provoz z více VLAN do přepínače pro distribuci do více AP. Každý přístupový bod může podporovat více sítí WLAN.



11.3 Konfigurace WPA2 podnikové WLAN na WLC

V následujícím bloku bude probrána konfigurace řadiče WLAN tak, aby odesílal SNMP trapy na externí server, dále aby využil externí RADIUS serveru k ověření připojovaných uživatelů a realizace ověření připojení s RADIUS serverem.

SNMP a RADIUS

Na obrázku vidíme příklad topologie, na níž budeme konfiguraci provádět. Na PC-A běží Simple Network Management Protocol (SNMP) a serverový software Remote Authentication Dial-In User Service (RADIUS).

- Správce sítě chce, aby řadič předal všechny zprávy protokolu SNMP (tj. depeše) na server SNMP.
- Dále chce použít server RADIUS pro služby ověřování, autorizace a účtování
- Pro připojení musí Uživatelé zadat své přihlašovací jméno a heslo, které ověří server RADIUS.
- Server RADIUS je vyžadován pro bezdrátové sítě, které používají ověřování WPA2 Enterprise.

Poznámka: Detailní konfigurace serveru SNMP a serveru RADIUS je nad rámec učiva tohoto modulu a nebude zde podrobně probírána.

Konfigurace SNMP serveru spočívá v povolení protokolu SNMP a následující konfiguraci nastavení:

1. Kliknutím na kartu Management hlavní nabídky získáme přístup k řadě funkcí pro správu.
2. Volbou **SNMP** se rozbalí dílčí nabídka.
3. Kliknutím do levého menu na položku Trap Receivers se zobrazí okno, kde zvolením položky **Nový (New...)** nakonfigurujeme nový přijímač SNMP trapů.
 - Je třeba zadat název SNMP komunity a IP adresu (IPv4 nebo IPv6) pro SNMP server. Kliknutím na **Použít (Apply)** zadání potvrďme.
 - Řadič nyní začne předávat zprávy protokolu SNMP na nastavený SNMP server.

RADIUS server je zařízení sítě, které dovoluje oddělit autorizační autoritu jako je např. LDAP nebo AD od zařízení vykonávající ověřování identity pro poskytovanou službu. V malých sítích si mnohdy vystačíme s ověřováním identit přímo na vstupních zařízeních. V enterprise prostředí je ale třeba centralizovat služby ověřování pro udržení konzistence prostředí. Není bezpečné ověřovat identity přímo proti aktivní databázi účtů, proto je zde nasazen prostředník – RADIUS server.

Pro konfiguraci řadiče s informacemi o RADIUS serveru je třeba vykonat toto:

1. Klikneme na **záložku SECURITY**
2. Klikneme v levém sloupci na nabídku **RADIUS** a vybereme Authentication
3. Vybereme New... a přidáme PC-A následujícím postupem jako RADIUS server.

Zadáme IPv4 adresu počítače PC-A, na kterém je nainstalovaný RADIUS server. Dále zadáme sdílené heslo (Shared Secret), které bude použito pro ověření mezi bezdrátovým řadičem a RADIUS serverem. Konfiguraci potvrďme kliknutím na tlačítko Apply. Z uvedeného vyplývá, že potřebujeme-li bezpečně propojit RADIUS server se zařízením, je třeba na obou zařízeních nastavit IP adresy pro navázání komunikace a zadat stejnou frázi sdíleného tajemství.

Po kliknutí na tlačítko Apply se seznam nakonfigurovaných ověřovacích RADIUS serverů (RADIUS Authentication Servers) aktualizuje o nový uvedený server.

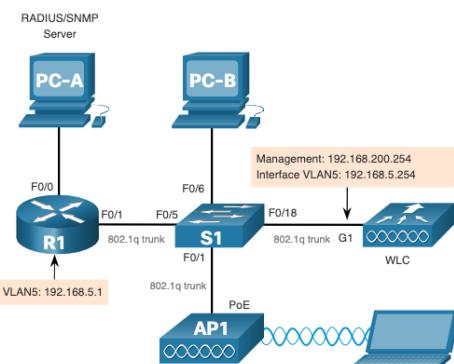
V dalším kroku se provede konfigurace Virtuální LAN pro novou bezdrátovou LAN, což zahrnuje:

- Ověření topologie
- Nasazení nového rozhraní virtuální sítě
- Přiřazení nového rozhraní virtuální sítě k bezdrátové síti

Topologie s adresováním VLAN 5

Každá WLAN nakonfigurovaná na WLC potřebuje své vlastní virtuální rozhraní. WLC má pět fyzických datových portů, které lze nakonfigurovat tak, aby podporovaly více sítí WLAN a virtuální rozhraní.

Na příkladu vidíme, že nová WLAN bude používat rozhraní VLAN 5 a síť 192.168.5.0/24. Proto byl R1 nakonfigurován pro VLAN 5, jak je znázorněno v topologii, a ve výstupu příkazu **show ip interface brief**.



```
R1# show ip interface brief
Interface          IP-Address      OK? Method Status      Protocol
FastEthernet0/0    172.16.1.1     YES manual up       up
FastEthernet0/1    unassigned      YES unset  up       up
FastEthernet0/1.1   192.168.200.1  YES manual up       up
FastEthernet0/1.5   192.168.5.254 YES manual up       up
(output omitted)
R1#
```

11.4 Odstraňování problémů s WLAN

Problémy se sítí mohou být jednoduché nebo složité a mohou být výsledkem kombinace problémů s hardwarem, softwarem a připojením. Než budou technici schopni vyřešit problém se sítí, musí analyzovat problém a určit příčinu chyby. Tento proces je podstatou řešení problémů.

Při řešení jakéhokoliv problému se sítí je třeba postupovat systematicky. Běžná a efektivní metodika řešení problémů je založena na vědecké metodě a lze ji rozdělit do šesti hlavních kroků uvedených v následující tabulce.

Krok	Název	Popis
1	Zjistěte problém	Prvním krokem v procesu odstraňování problémů je identifikace problému. I když v tomto kroku lze použít nástroje, konverzace s uživatelem je často velmi užitečná.
2	Stanovte teorii pravděpodobných příčin	Poté, co jste mluvili s uživatelem a identifikovali problém, můžete zkoušet vytvořit teorii pravděpodobných příčin. Tento krok často přináší více než několik pravděpodobných příčin problému.
3	Vyzkoušejte teorii a určete příčinu	Na základě pravděpodobných příčin otestujte své teorie a určete, která z nich je příčinou problému. Technik často použije rychlý postup k testování a zjistí, zda problém vyřeší. Pokud rychlý postup problém nevyřeší, možná budete muset problém dál prozkoumat a zjistit přesnou příčinu.
4	Vytvořte akční plán k vyřešení problému a implementujte řešení	Poté, co určíte přesnou příčinu problému, vytvořte akční plán k vyřešení problému a implementujte řešení.
5	Ověřte funkčnost celého systému a implementujte preventivní opatření	Po odstranění problému ověřte plnou funkčnost a případně proveděte preventivní opatření.
6	Dokumentujte zjištění, akce a výsledky	V posledním kroku procesu řešení potíží zdokumentujte svá zjištění, akce a výsledky. To je velmi důležité pro budoucí použití.

Bezdrátový klient se nepřipojuje

Pokud není připojení funkční, zkонтrolujeme následující:

- Ověříme síťovou konfiguraci v počítači pomocí příkazu **ipconfig**.
- Ověříme, že se zařízení může připojit ke kabelové síti. Ping na známou IP adresu.
- V případě potřeby znova načteme ovladače klienta nebo vyzkoušíme jiný bezdrátový síťový adaptér.
- Pokud bezdrátová síťová karta klienta funguje, zkонтrolujeme režim zabezpečení a nastavení šifrování na klientovi.

Pokud je počítač funkční, ale bezdrátové připojení funguje špatně, zkонтrolujeme následující:

- Je počítač mimo plánovanou oblast pokrytí (BSA)?
- Zkontrolujeme nastavení kanálu na bezdrátovém klientovi.
- Zkontrolujeme rušení v pásmu 2,4 GHz.

Dále zkonzolujte, zda jsou všechna zařízení skutečně na svém místě.

- Zvažte možný problém s fyzickým zabezpečením.
- Jsou všechna zařízení napájena a jsou zapnuta?

Nakonec zkonzolujte propojení mezi kabelovými zařízeními a hledejte špatné konektory nebo poškozené nebo chybějící kably.

- Pokud je fyzické zařízení na místě, ověřte kabelovou LAN pomocí příkazu **ping** na jednotlivá zařízení, včetně AP.

- Pokud připojení v tomto okamžiku stále nefunguje, možná něco není v pořádku s AP nebo jeho konfigurací.
- Když je uživatelský počítač vyloučen jako zdroj problému a fyzický stav zařízení je v pořádku, přezkoumejte výkon AP.
- Zkontrolujte stav napájení AP.

Odstraňování problémů, když je síť pomalá

Chceme-li optimalizovat a zvýšit šířku pásma dvoupásmových wi-fi směrovačů a přístupových bodů, je třeba postupovat takto:

- **Upgradovat bezdrátové klienty** - Starší zařízení 802.11b, 802.11g a dokonce i 802.11n mohou zpomalit celou WLAN. Pro nejlepší výkon by všechna bezdrátová zařízení měla podporovat stejný nejvyšší možný standard.
- **Rozdělit provoz (split-the-traffic)** - Nejjednodušší způsob, jak zlepšit bezdrátový výkon, je rozdělit bezdrátový provoz mezi pásmo 2,4 GHz a pásmo 5 GHz. Standard 802.11n (nebo lepší) může používat tato dvě pásmata jako dvě samostatné bezdrátové sítě, což pomáhá řídit provoz.

Existuje několik důvodů pro použití přístupu split-the-traffic:

- Pásmo 2,4 GHz může být vhodné pro základní internetový provoz, který není časově citlivý.
- Šířku pásmata lze i nadále sdílet s dalšími blízkými sítěmi WLAN.
- Pásmo 5 GHz je mnohem méně přeplňené než pásmo 2,4 GHz; je ideální pro streamování multimédií.
- Navíc má toto pásmo více kanálů, proto bude pravděpodobně zvolený kanál bez interference.

Ve výchozím nastavení používají dvoupásmové směrovače a přístupové body stejný název sítě v pásmu 2,4 GHz i 5 GHz.

- Může být užitečné segmentovat provoz.
- Nejjednodušší způsob segmentace provozu je přejmenování jedné z bezdrátových sítí.

Chcete-li zlepšit dosah bezdrátové sítě, zajistěte, aby umístění bezdrátového směrovače nebo přístupového bodu neblokovaly žádné překážky, jako je nábytek, příslušenství a vysoká zařízení.

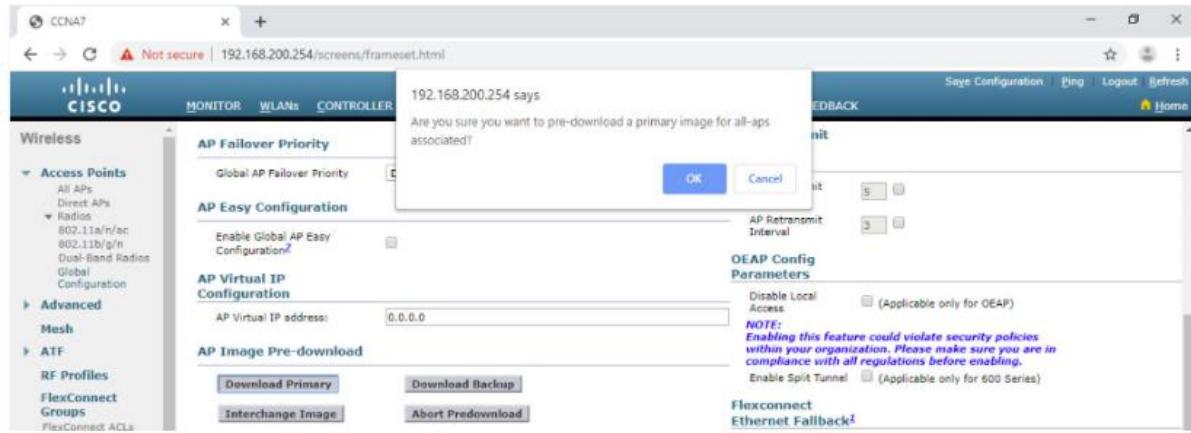
- Blokují signál, což zkracuje dosah WLAN.
- Pokud to problém stále nevyřeší, můžete použít prodlužovač dosahu Wi-Fi (extender) nebo nasadit bezdrátovou technologii Powerline.

Aktualizace Firmware

Poslední metodou vedoucí k odstranění problémů je aktualizace firmwaru. Většina bezdrátových směrovačů a přístupových bodů nabízí upgradovatelný firmware, který by měl být pravidelně ověřován.

Na řadičích bezdrátových sítí bude s největší pravděpodobností možnost upgradovat firmware na všech AP, které řídí.

- Na obrázku je příklad, kde je stažen obraz firmwaru, který bude použitý k aktualizaci všech přístupových bodů.



- Na bezdrátovém řadiči Cisco 3504 se dosáhne distribuovaného upgradu přes nabídku **WIRELESS > Access Points > Global Configuration**. Poté se zvolí v dolní části stránky sekce AP Image Pre-download.

Distribuce je pak věcí komunikace řadiče s AP.

Počítačové sítě 2

12, Základy směrování

Název tématu	Cíl
Určení cesty	Vysvětlit, jak směrovače určují nejlepší cestu.
Přeposílání paketů	Vysvětlit, jak směrovače předávají pakety na místo určení.
Přehled základní konfigurace směrovače	Nakonfigurovat základní nastavení na routeru.
Směrovací tabulka IP	Popsat strukturu směrovací tabulky.
Statické a dynamické směrování	Porovnat principy statického a dynamického směrování.

12.1 Určení cesty

Dvě funkce směrovače

Když směrovač obdrží IP paket na jednom rozhraní, určí, které rozhraní se použije k předání paketu na místo určení. Tomu se říká směrování. Rozhraní, které směrovač používá k předávání paketů, může být konečným cílem nebo to může být síť připojená k jinému směrovači, který se používá k dosažení cílové sítě. Každá síť, ke které se router připojuje, obvykle vyžaduje samostatné rozhraní, ale nemusí tomu tak vždy být.

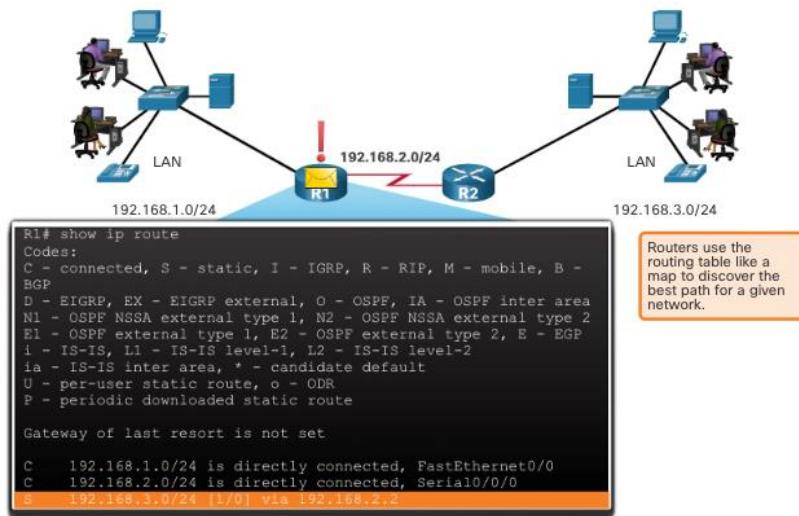
Primární funkce routeru jsou určit nejlepší cestu k předávání paketů na základě informací v jeho směrovací tabulce a předávání paketů směrem k jejich cíli.

Příklad funkcí směrovače

Směrovač používá svou směrovací tabulku k určení, kterou cestu (trasu) použít k předání paketu.

Na snímku vidíme konfiguraci se dvěma routery a výpis z Routeru R1.

Routery R1 a R2 použijí své příslušné směrovací tabulky k určení nejlepší cesty a následnému předání paketu. Na výpisu R1 dole vidíme zvýrazněný statický záznam, jehož syntaxi si objasníme dále.



Nejlepší cesta se rovná nejdelší shodě

Při určování cesty platí, že Nejlepší cesta ve směrovací tabulce je ta s nejdelší shodou.

Směrovací tabulka obsahuje položky směrování skládající se z (předpony síťové adresy - prefix) a délky předpony. Aby mezi cílovou IP adresou paketu a trasou ve směrovací tabulce byla shoda, musí se mezi IP adresou paketu a trasou ve směrovací tabulce shodovat minimální počet krajních levých bitů. Délka předpony trasy ve směrovací tabulce se používá k určení minimálního počtu krajních levých bitů, které se musí shodovat. Nejdelší shodou je trasa ve směrovací tabulce, která má největší počet levých shodných bitů s cílovou IP adresou paketu. Nejdelší shoda je vždy preferovaná trasa.

Termín délka předpony (nebo také prefix) bude dále použitý k označení síťové části adres IPv4 i IPv6.

Příklad IPv4 nejdelší shody

IPv4 paket v tabulce má cílovou adresu 172.16.0.10. Směrovač má ve směrovací tabulce IPv4 tři položky trasy, které odpovídají tomuto paketu: 172.16.0.0/12, 172.16.0.0/18 a 172.16.0.0/26. Ze tří tras má 172.16.0.0/26 nejdelší shodu a byla by tedy vybrána k předání paketu. Aby kterákoli z těchto tras byla považována za shodu, musí mezi ní a cílovou adresou korespondovat počet souhlasných bitů naznačený maskou podsítě trasy.

Destination IPv4 Address		Address in Binary
172.16.0.10		10101100.00010000.00000000.00001010
Route Entry	Prefix/Prefix Length	Address in Binary
1	172.16.0.0/12	10101100.00010000.00000000.00001010
2	172.16.0.0/18	10101100.00010000.00000000.00001010
3	172.16.0.0/26	10101100.00010000.00000000.00001010

Příklad IPv6 nejdelší shody

Pro IPv6 toto pak platí následovně. Paket IPv6 má cílovou adresu IPv6 2001:db8:c000::99. Tento příklad ukazuje tři možné trasy, ale pouze dvě z nich jsou platnou shodou, přičemž jedna z nich je pak nejdelší shodou. První dvě položky trasy mají délky předpony, které mají požadovaný počet odpovídajících bitů, jak je indikováno délkou předpony. Třetí položka trasy není shoda, protože její předpona /64 vyžaduje 64 odpovídajících bitů, což ve čtvrtém bloku nesedí.

Destinatio n	2001:db8:c000::99/48	
Route Entry	Prefix/Prefix Length	Does it match?
1	2001:db8:c000:/40	Match of 40 bits
2	2001:db8:c000:/48	Match of 48 bits (longest match)
3	2001:db8:c000:5555:/64	Does not match 64 bits

Sestavení směrovací tabulky

Jak se tedy sestavuje routovací / směrovací tabulka. Rozeznáváme:

- Přímo připojené sítě
- Vzdálené sítě

Přímo připojené sítě jsou do směrovací tabulky přidány, když je místní rozhraní nakonfigurováno pomocí IP adresy a masky podsítě (délka předpony) a je aktivní (up - up).

Vzdálené sítě jsou pak ty, které nejsou přímo připojeny k routeru. Směrovače získají informaci o vzdálených sítích dvěma způsoby:

- **Statické trasy** jsou přidány do směrovací tabulky, když je trasa nakonfigurována ručně.
- **Dynamické trasy vznikají na základě dynamických směrovacích protokolů**. Jsou přidány do směrovací tabulky, když se směrovací protokoly dynamicky učí od připojených routerů o vzdálené síti.

Mimo to je zde pro routování důležitý pojem - Výchozí trasa: Určuje směrovač dalšího skoku (next-hop), který se použije, když směrovací tabulka neobsahuje konkrétní směrování, které odpovídá

cílové IP adresy. Výchozí trasu lze zadat ručně jako statickou trasu nebo se ji lze naučit automaticky z protokolu dynamického směrování.

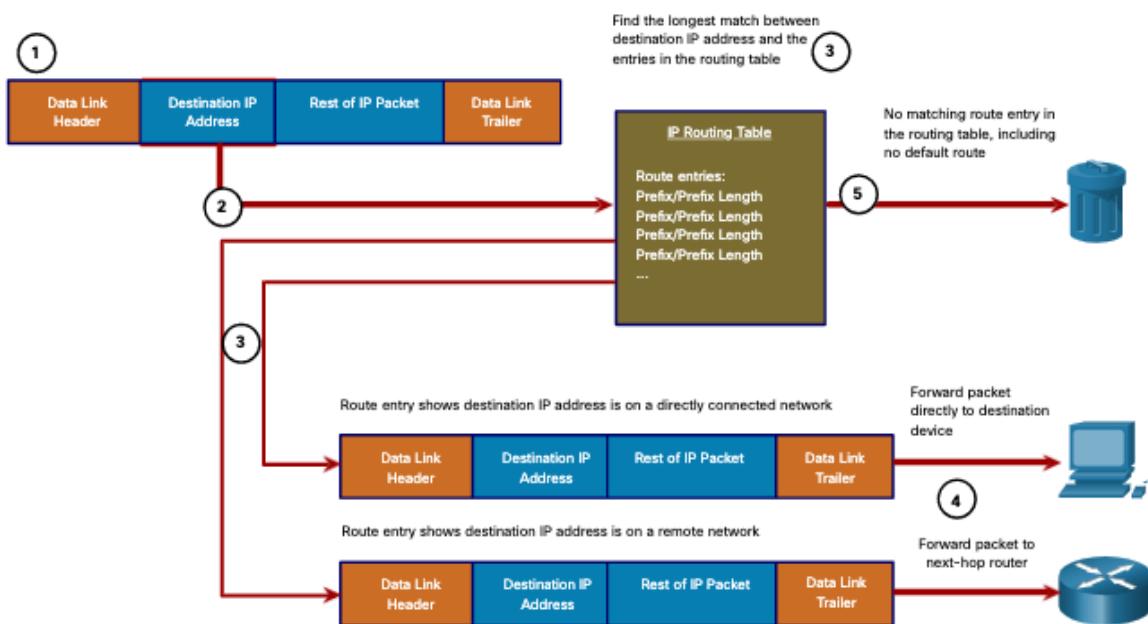
Výchozí trasa má délku předpony /0. To znamená, že pro použití této trasy se žádné bity nemusí shodovat s cílovou IP adresou. Pokud neexistují žádné cesty se shodou delší než 0 bitů, použije se k předání paketu výchozí trasa. Výchozí trasa se někdy označuje jako brána poslední instance (gateway of last resort).

12.2 Přeposílání paketů

Proces rozhodování o předávání paketů

O předávání paketů se stará rozhodovací proces, který můžeme rozložit na pět částí, které jsou vyznačeny v obrázku na snímku.

1. Rámeček datového spojení se zapouzdřeným IP paketem dorazí na vstupní rozhraní.
2. Směrovač prozkoumá cílovou IP adresu v záhlaví paketu a konzultuje svoji směrovací tabulkou.
3. Směrovač najde nejdelší shodnou předponu ve směrovací tabulce.
4. Směrovač zapouzdří paket v rámci datového spojení a předává jej ven z výstupního rozhraní. Cílem může být zařízení připojené k síti nebo další směrovač (next-hop).
5. Pokud neexistuje žádný odpovídající záznam trasy, paket je zahozen.



Poté, co směrovač určí nejlepší cestu, může provést **přeposlání paketu na zařízení v přímo připojené síti**. Pokud položka trasy naznačuje, že výstupním rozhraním je přímo připojená síť, paket lze předat přímo do cílového zařízení. Obvykle se jedná o Ethernet LAN.

Pro zapouzdření paketu v ethernetovém rámcu musí směrovač určit cílovou MAC adresu spojenou s cílovou IP adresou paketu. Proces se liší podle toho, zda je paket typu IPv4 nebo IPv6.

Nebo je zde druhá možnost. Poté, co směrovač určí nejlepší cestu, může **přeposlat paket na další (next-hop) směrovač a to za podmínky, že** položka trasy naznačuje, že cílová IP adresa je ve vzdálené síti. Znamená to, že se zařízení nachází v síti, která není přímo připojena. Paket musí být proto předán dalšímu (next-hop) směrovači. Adresa dalšího skoku je uvedena v záznamu trasy.

Pokud jsou oba směrovače v síti Ethernet, dojde k podobnému procesu (ARP a ICMPv6 Neighbour Discovery) pro určení cílové MAC adresy paketu, jak bylo uvedeno v předchozím kroku. Rozdíl je v tom, že směrovač bude hledat IP adresu dalšího (next-hop) směrovače ve své ARP tabulce nebo v mezipaměti sousedů namísto cílové IP adresy paketu.

Tento proces se bude pochopitelně lišit pro ostatní typy sítí vrstvy 2.

Jako poslední může nastat situace, kdy se směrovač snaží určit nejlepší cestu, avšak **ve směrovací tabulce není žádná shoda, proto Zahodí paket. Jinými slovy..**

Pokud neexistuje žádná shoda mezi cílovou IP adresou a předponou ve směrovací tabulce a pokud neexistuje žádná výchozí trasa, paket bude zahozen.

End-to-end předávání paketů

Jak již bylo naznačeno, proces se bude lišit pro ostatní typy sítí druhé vrstvy.

Primární odpovědností funkce předávání paketů je zapouzdření paketů do příslušného typu rámce datového spojení pro odchozí rozhraní. Například formát rámce datového spoje pro sériovou linku může být protokol PPP (Point-to-Point), protokol HDLC (High-Level Data Link Control) nebo jiný protokol vrstvy 2.

Mechanismy předávání paketů

Primární odpovědností funkce předávání paketů je zapouzdření paketů do příslušného typu rámce datového spojení pro odchozí rozhraní. Čím efektivněji může směrovač provádět tento úkol, tím rychleji může směrovač pakety předávat.

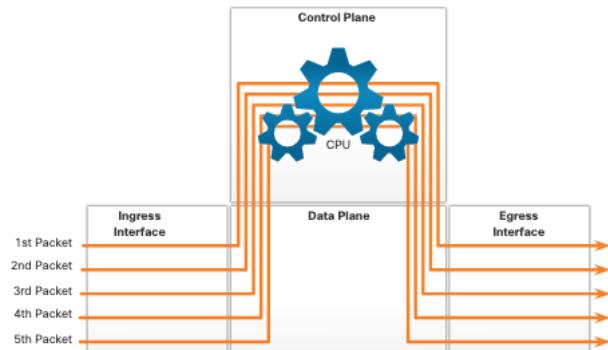
Směrovače podporují následující tři mechanismy předávání paketů:

- Přepínání se zpracováním (Process switching)
- Rychlé přepínání (Fast switching)
- Cisco Express Forwarding (CEF)

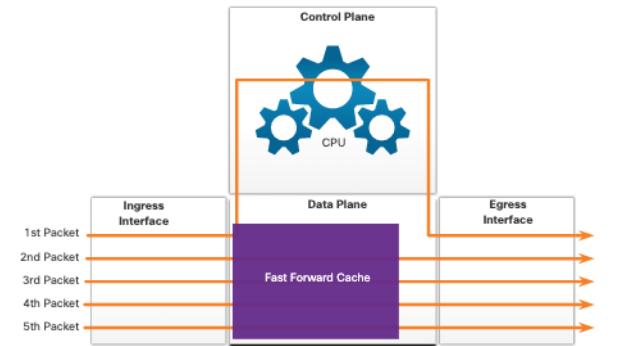
Nyní k jednotlivým variantám podrobněji.

Přepínání se zpracováním (Process switching):

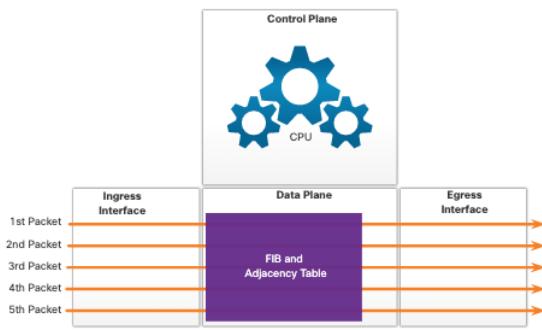
Jde o starší mechanismus předávání paketů, který je pro směrovače Cisco stále k dispozici. Když paket dorazí na rozhraní, je předán na řídící jednotku, kde procesor srovnává cílovou adresu se záznamem ve své směrovací tabulce, a poté určí výstupní rozhraní a předá paket. Je důležité si uvědomit, že směrovač toto dělá pro každý paket, i když je cíl stejný pro proud paketů.



Rychlé přepínání (Fast switching): Další, starší mechanismus předávání paketů, který byl nástupcem Přepínání se zpracováním. Rychlé přepínání používá k ukládání informací o dalším skoku (next-hop) mezipaměť rychlého přepínání. Když paket dorazí na rozhraní, je předán do řídicí úrovně, kde CPU hledá shodu v mezipaměti rychlého přepínání. Pokud tam shoda není, je paket zpracován metodou Přepínání se zpracováním a přesměruje se na výstupní rozhraní. Informace o způsobu zpracování paketu se poté uloží do mezipaměti rychlého přepínání. Pokud na rozhraní dorazí další paket směrující do stejného cíle, informace o dalším skoku v mezipaměti se znova použijí bez zásahu CPU.



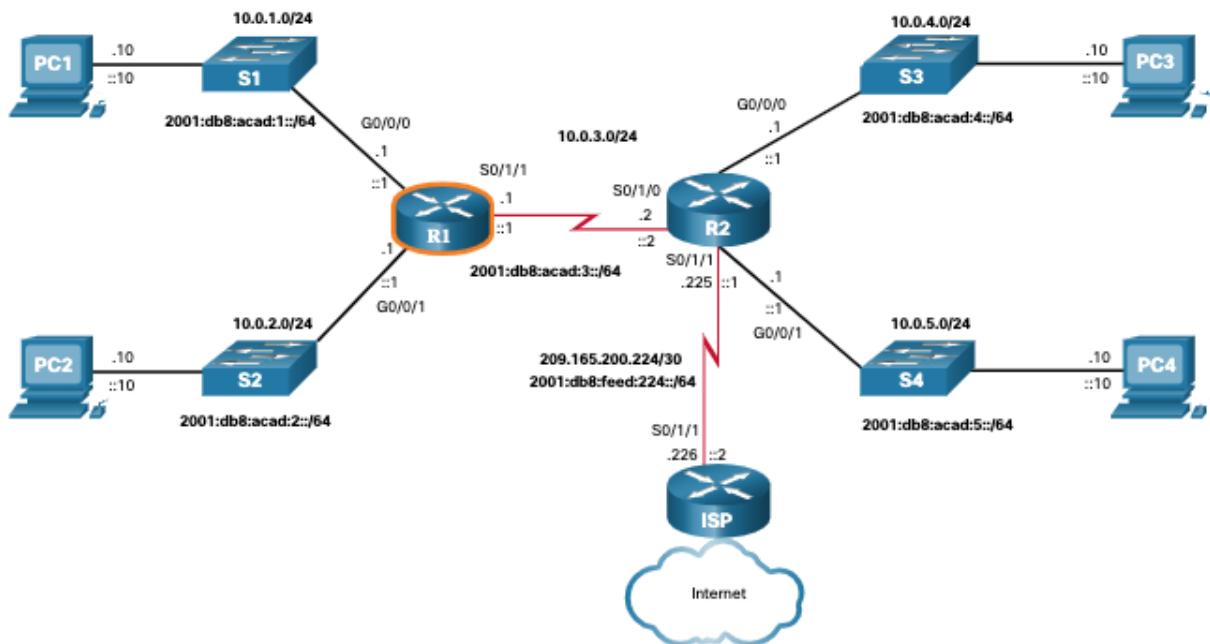
Cisco Express Forwarding (CEF): Nejnovější a výchozí mechanismus předávání paketů Cisco IOS. CEF vytváří Forwarding Information Base (FIB) a tabulku sousedství (Adjacency table). Položky tabulky se nespouštějí pakety jako u rychlého přepínání, ale spouštějí se změnami, například když se něco změní v topologii sítě. Když síť zkonzervovala, FIB a tabulky sousedství obsahují všechny informace, které by směrovač měl brát v potaz při předávání paketu.



12.3 Přehled základní konfigurace směrovače

Topologie

Topologie uvedená na obrázku bude použita pro příklady konfigurace a ověření. Bude také použita v dalším tématu k diskusi o IP směrovací tabulce.



Příkazy konfigurace

V první části je třeba vytvořit základní konfiguraci a zabezpečit vzdálený přístup k zařízení:

```
Router> enable
Router# configure terminal
Enter configuration commands, one per line. End with
CRTL/Z.
Router(config)# hostname R1
R1(config)# enable secret class
R1(config)# line console 0
R1(config-line)# logging synchronous
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# transport input ssh telnet
R1(config-line)# exit
R1(config)# service password-encryption R1(config)#
banner motd #
Enter TEXT message. End with a new line and the #
*****
WARNING: Unauthorized access is prohibited!
*****
#
```

V druhém kroku se provádí konfigurace rozhraní pro IPv4 a IPv6 protokoly:

```
R1(config)# ipv6 unicast-routing
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# description Link to LAN 1
R1(config-if)# ip address 10.0.1.1 255.255.255.0
R1(config-if)# ipv6 address 2001:db8:acad:1::1/64
R1(config-if)# ipv6 address fe80::1:a link-local
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# description Link to LAN 2
R1(config-if)# ip address 10.0.2.1 255.255.255.0
R1(config-if)# ipv6 address 2001:db8:acad:2::1/64
R1(config-if)# ipv6 address fe80::1:b link-local
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface serial 0/1/1
R1(config-if)# description Link to R2
R1(config-if)# ip address 10.0.3.1 255.255.255.0
R1(config-if)# ipv6 address 2001:db8:acad:3::1/64
R1(config-if)# ipv6 address fe80::1:c link-local
R1(config-if)# no shutdown
R1(config-if)# exit
R1# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
```

Ověřovací příkazy

Abychom si mohli ověřit provedenou konfiguraci použijeme běžné ověřovací příkazy, mezi které patří:

- **show ip interface brief**
- **show running-config interface *interface-type number***
- **show interfaces**
- **show ip interface**
- **show ip route**
- **ping**

Pro kontrolu ipv6 ve všech uvedených případech nahraďte **ip** za **ipv6**.

12.4 Směrovací tabulka IP

Zdroje tras

Směrovací tabulka obsahuje seznam cest do známých sítí daný předponami a délkom předpon. Zdroj těchto informací je odvozen z informací o:

- Přímo připojených sítích
- Statických trasách
- Sítích zpropagovaných přes Dynamické směrovací protokoly

Zdroj pro každou trasu ve směrovací tabulce je identifikován kódem. Mezi nejčastější kódy patří:

- L ... local - Určuje adresu přiřazenou rozhraní routeru.
- C ... connected - Určuje přímo připojenou síť.
- S ... static - Určuje statickou cestu vytvořenou k dosažení konkrétní sítě.
- O ... OSPF - Identifikuje dynamicky naučenou síť z jiného routeru pomocí směrovacího protokolu OSPF.
- * ... Tato trasa je kandidátem na výchozí trasu.

Zásady směrovací tabulky

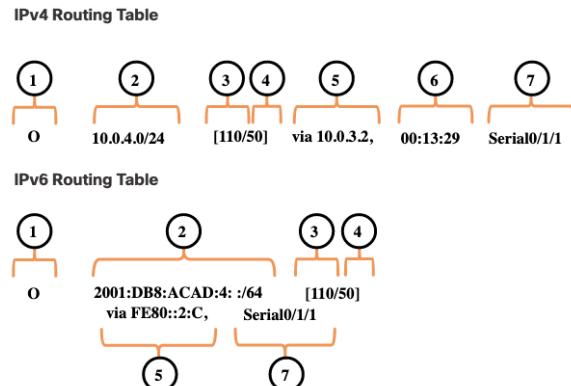
V následující tabulce jsou popsány tři principy směrovací tabulky. Jedná se o definici problémů, které řeší správná konfigurace protokolů dynamického směrování nebo statických cest na všech směrovačích mezi zdrojovým a cílovým zařízením.

Zásada směrovací tabulky	Příklad
Každý směrovač se rozhoduje sám na základě informací, které má ve své vlastní směrovací tabulce.	<ul style="list-style-type: none">směrovač může přepošílat pakety pouze pomocí vlastní směrovací tabulky.Směrovač neví, jaké cesty jsou ve směrovacích tabulkách jiných směrovačů (např. R2).
Informace ve směrovací tabulce jednoho směrovače nemusí nutně odpovídat směrovací tabulce jiného směrovače.	Jen to, že R1 má ve své směrovací tabulce cestu k síti na internetu přes R2, ještě neznamená, že R2 ví o stejně síti.
Informace o směrování cesty neposkytuje informace o zpětném směrování.	R1 přijímá paket s cílovou IP adresou PC1 (přímo připojená síť) a zdrojovou IP adresou PC3 (ze vzdálené sítě přes R2). Jen proto, že R1 ví, že má přepošílat paket ze svého rozhraní G0/0/0, nemusí nutně znamenat, že ví, jak předávat pakety pocházející z PC1 zpět do vzdálené sítě PC3.

Položky směrovací tabulky

Při zobrazení výpisu routovací tabulky pro IPv4 a IPv6 je zde jistý rozdíl, spíše však v uspořádání položek, jak je patrné z obrázku. Položky jsou očíslované a obsahují následující informace:

1. **Zdroj trasy** Určuje, jakým způsobem se trasa naučila.
2. **Cílová síť (je dána předponou a délkom předpony)** a identifikuje adresu vzdálené sítě.
3. **Administrativní vzdálenost** identifikuje důvěryhodnost zdroje trasy. Nižší hodnoty označují preferovaný zdroj trasy.
4. **Metrika** identifikuje hodnotu přiřazenou k dosažení vzdálené sítě. Nižší hodnoty označují preferované trasy.
5. **Next-hop** Identifikuje IP adresu dalšího routeru, na který bude případně paket předán.
6. **Časové razítko trasy** Určuje u IPv4, kolik času uplynulo od načtení trasy.
7. **Výchozí rozhraní** identifikuje výstupní rozhraní, které se použije pro odchozí pakety k dosažení jejich konečného cíle.



Poznámka: Délka předpony cílové sítě určuje minimální počet krajních levých bitů, které se musí shodovat mezi IP adresou paketu a cílovou sítí (předponou) pro použití této trasy.

Přímo připojené sítě

Aby se směrovač dozvěděl o vzdálených sítích, musí mít alespoň jedno aktivní rozhraní nakonfigurované s IP adresou a maskou podsítě (tedy délkou předpony). Toto se nazývá přímo připojená síť nebo přímo připojená trasa. Směrovače přidávají přímo připojenou trasu do své směrovací tabulky, když je rozhraní nakonfigurováno s IP adresou a je aktivováno.

- Přímo připojená síť je označena stavovým kódem **C** ve směrovací tabulce. Trasa obsahuje síťovou předponu a délku předpony.
- Směrovací tabulka obsahuje také lokální trasu pro každou z jejích přímo připojených sítí, označenou stavovým kódem **L**.
- U lokálních tras IPv4 je délka předpony /32 a u lokálních tras IPv6 je délka předpony /128. To znamená, že cílová IP adresa paketu musí odpovídat všem bitům v místní trase, aby se tato trasa shodovala. Účelem lokální trasy je efektivně určit, když směrovač obdrží paket pro rozhraní namísto paketu, který je třeba předat dál.

Statické cesty

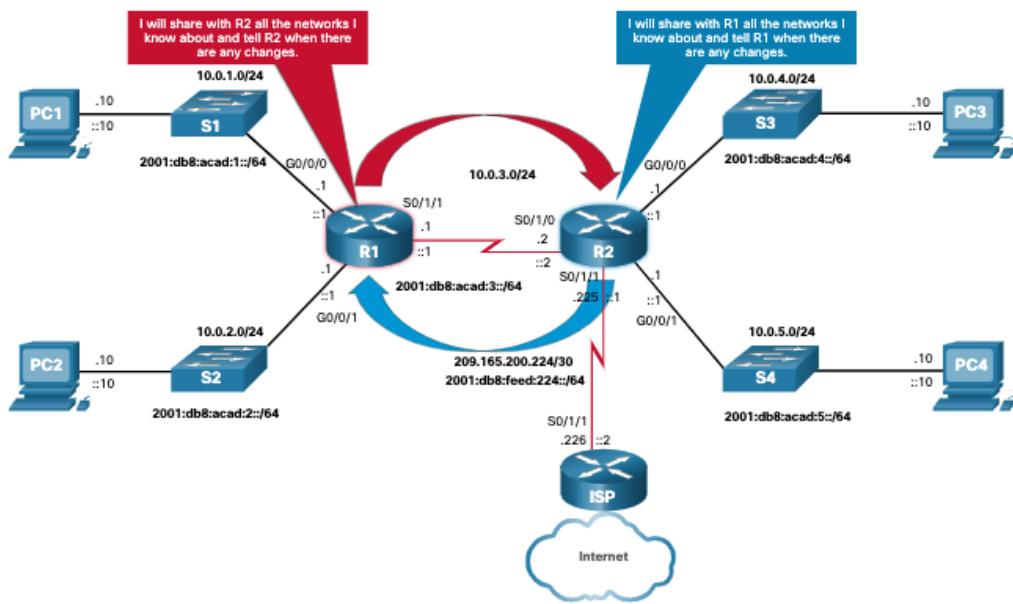
Poté, co jsou přímo připojená rozhraní nakonfigurována a přidána do směrovací tabulky, lze pro přístup ke vzdáleným sítím implementovat statické nebo dynamické směrování. Statické trasy se konfigurují ručně. Definují explicitní cestu mezi dvěma síťovými zařízeními. Nejsou automaticky aktualizovány a musí se ručně překonfigurovat, pokud se změní topologie sítě.

Statické směrování má tři hlavní důvody použití:

- Poskytuje snadnou údržbu směrovací tabulky v menších sítích, u nichž se neočekává výrazný nárůst.
- Používá jedinou výchozí trasu představující cestu k jakékoli síti, která nemá konkrétnější shodu s jinou trasou ve směrovací tabulce. Výchozí trasy se používají k odesílání provozu do libovolného cíle mimo další předcházející směrovač.
- Je směrován do a z krajních (stub) sítí. Krajní síť je síť přístupná jedinou cestou a směrovač má pouze jednoho souseda.

Dynamické směrovací protokoly

Na směrovačích se výrazně častěji v enterprise prostředí používají dynamické směrovací protokoly, k automatickému sdílení informací o dosažitelnosti a stavu vzdálených sítí. Dynamické směrovací protokoly provádějí několik činností, včetně zjišťování sítí a údržby směrovacích tabulek.



Dynamické trasy ve směrovací tabulce

Jako příklad si uvedeme Protokol OSPF. V naší ukázkové topologii se používá k dynamickému učení všech sítí připojených k R1 a R2. Položky směrovací tabulky obsahují stavový kód „O“ k označení trasy, kterou se směrovač pomocí OSPF naučil. Obě položky také obsahují IP adresu next-hop směrovače resp. IP adresu jeho vstupního rozhraní.

Směrovací protokoly IPv6 používají místní adresu next-hop směrovače.

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP D - EIGRP,
EX - EIGRP external, O - OSPF, IA - OSPF inter area
(output omitted for brevity)
O 10.0.4.0/24 [110/50] via 10.0.3.2, 00:24:22, Serial0/1/1
O 10.0.5.0/24 [110/50] via 10.0.3.2, 00:24:15, Serial0/1/1
R1# show ipv6 route
IPv6 Routing Table - default - 10 entries
(Output omitted)
NDR - Redirect, RL - RPL, O - OSPF Intra, OI - OSPF Inter
O 2001:DB8:ACAD:4::/64 [110/50]
  via FE80::2:C, Serial0/1/1
O 2001:DB8:ACAD:5::/64 [110/50]
  via FE80::2:C, Serial0/1/1
```

Konfigurací směrování pomocí OSPF protokolu se detailněji zabývat nebudeme, problematika je nad rámec předmětu PSIT2.

Výchozí trasa

Již jsme si zmiňovali pojem výchozí trasy. Ta nám určuje next-hop směrovač, který se má použít, když směrovací tabulka neobsahuje konkrétní trasu, která odpovídá cílové IP adresy. Výchozí trasa může být buď statická trasa, nebo ji lze naučit automaticky z protokolu dynamického směrování. Výchozí trasa má IPv4 záznam trasy 0.0.0.0/0 nebo IPv6 záznam trasy ::/0. To znamená, že mezi cílovou IP adresou a výchozí trasou nemusí odpovídat žádné bity. Pakety s adresou neznámé sítě nebudou na takto konfigurovaném směrovači zahozené, ale předány k realizaci na router definovaný rozhraním či adresou.

Struktura IPv4 směrovací tabulky

Protokol IPv4 byl standardizován pomocí dnes již zastaralé architektury klasického (classful) adresování. Směrovací tabulka IPv4 je organizována pomocí stejné klasické struktury. Ačkoli proces vyhledávání již nepoužívá třídy, struktura IPv4 směrovací tabulky se v tomto formátu stále zachovává.

Na obrázku s výpisem vidíme, že byl použitý příkaz show ip route. Pohled na výpis není kompletní, vidíme zde jen popisovanou část.

- Odsazený záznam je známý jako **podřízená cesta** (child route). Položka trasy je odsazena, pokud se jedná o podsíť klasické adresy (sítě třídy A, B nebo C).
- Přímo připojené sítě budou vždy odsazeny (podřízené trasy), protože místní adresa rozhraní je vždy zadána ve směrovací tabulce jako /32.
- Podřízená trasa bude zahrnovat zdroj trasy a všechny informace o přenosu, jako je adresa dalšího směrování (next-hop).
- Klasická síťová adresa této podsítě se zobrazí nad položkou trasy, méně odsazená a bez zdrojového kódu. Tato trasa je známá jako **nadřazená trasa** (parent route).

```
Router# show ip route
(Output omitted)
  192.168.1.0/24 is variably..
C   192.168.1.0/24 is direct..
L   192.168.1.1/32 is direct..
O   192.168.2.0/24 [110/65]..
O   192.168.3.0/24 [110/65]..
  192.168.12.0/24 is variab..
C   192.168.12.0/30 is direct..
L   192.168.12.1/32 is direct..
  192.168.13.0/24 is variably..
C   192.168.13.0/30 is direct..
L   192.168.13.1/32 is direct..
  192.168.23.0/30 is subnette..
O   192.168.23.0/30 [110/128]..
Router#
```

Struktura IPv6 směrovací tabulky

Výpis pro protokol IPv6 na obrázku má v porovnání s IPv4 pozměněnou strukturu. K jeho provedení byl použitý příkaz show ipv6 route.

Koncept klasického adresování nikdy nebyl součástí IPv6, takže struktura IPv6 směrovací tabulky je velmi přímočará. Každý záznam IPv6 trasy je formátován a zarovnán stejným způsobem.

```
R1# show ipv6 route
(output omitted for brevity)
OE2 ::/0 [110/1], tag 2
    via FE80::2:C, Serial0/0/1
C 2001:DB8:ACAD:1::/64 [0/0]
    via GigabitEthernet0/0/0, directly connected
L 2001:DB8:ACAD:1::1/128 [0/0]
    via GigabitEthernet0/0/0, receive
C 2001:DB8:ACAD:2::/64 [0/0]
    via GigabitEthernet0/0/1, directly connected
L 2001:DB8:ACAD:2::1/128 [0/0]
    via GigabitEthernet0/0/1, receive
C 2001:DB8:ACAD:3::/64 [0/0]
    via Serial0/1/1, directly connected
L 2001:DB8:ACAD:3::1/128 [0/0]
    via Serial0/1/1, receive
O 2001:DB8:ACAD:4::/64 [110/50]
    via FE80::2:C, Serial0/1/1
O 2001:DB8:ACAD:5::/64 [110/50]
    via FE80::2:C, Serial0/1/1
L FF00::/8 [0/0]
    via Null0, receive
R1#
```

Administrativní vzdálenost

Položka tras pro konkrétní síťovou adresu (jak již víme - předpona a délka předpony) se ve směrovací tabulce může zobrazit pouze jednou. Je však možné, že se směrovací tabulka dozví o stejné síťové adrese z více než jednoho zdroje směrování. S výjimkou velmi specifických okolností by měl být na směrovači implementován pouze jeden protokol dynamického směrování. Každý směrovací protokol se může rozhodnout pro jinou cestu k dosažení cíle na základě metriky tohoto směrovacího protokolu.

To vyvolává několik otázek, například následující:

- Jak směrovač ví, který zdroj použít?
- Jakou trasu má přidat do směrovací tabulky?

Systém Cisco IOS používá k určení cesty, která bude přidána do směrovací tabulky takzvanou administrativní vzdálenost (AD). Ta představuje „důvěryhodnost“ tras. Platí, že čím nižší je hodnota administrativní vzdálenosti, tím důvěryhodnější je zdroj tras.

Vychází se z pevně daných hodnot přiřazených různým směrovacím protokolům, které vidíme uspořádané na snímku v tabulce.

Zdroj trasy	Administrativní vzdálenost
Directly connected	0
Static route	1
EIGRP summary route	5
External BGP	20
Internal EIGRP	90
OSPF	110
IS-IS	115
RIP	120
External EIGRP	170
Internal BGP	200

Nejnižší hodnotu nula mají přímo připojené sítě, jedničkou jsou ohodnocené staticky definované směry.

12.5 Statické a dynamické směrování

Statické nebo dynamické?

Statické a dynamické směrování se navzájem nevylučují. Většina sítí spíše používá kombinaci dynamických směrovacích protokolů a statických cest.

Statické trasy se běžně používají v následujících scénářích:

- Jako výchozí směrování paketů k poskytovateli služeb
- Pro trasy mimo směrovací doménu a nenaučené protokolem dynamického směrování
- Když chce správce sítě explicitně definovat cestu pro konkrétní síť
- Pro směrování mezi pahýlovými (stub) sítěmi

Statické trasy jsou užitečné pro menší sítě s pouze jednou cestou k vnější síti. Poskytují také zabezpečení ve větší síti pro určité typy provozu nebo odkazy na jiné sítě, které vyžadují větší kontrolu.

Protokoly dynamického směrování jsou implementovány v jakémkoli typu sítě sestávající z více než jen několika směrovačů. Dynamické směrovací protokoly jsou škálovatelné a automaticky určují lepší trasy, pokud dojde ke změně v topologii.

Protokoly dynamického směrování se běžně používají v následujících scénářích:

- V sítích sestávajících z více než jen několika směrovačů
- Když změna topologie sítě vyžaduje, aby síť automaticky určila jinou cestu
- Pro škálovatelnost. Jak síť roste, protokol dynamického směrování se automaticky učí o všech nových sítích.

Na závěr polemiky nad volbou statického a dynamického směrování si udělejme souhrnné porovnání v tabulce.

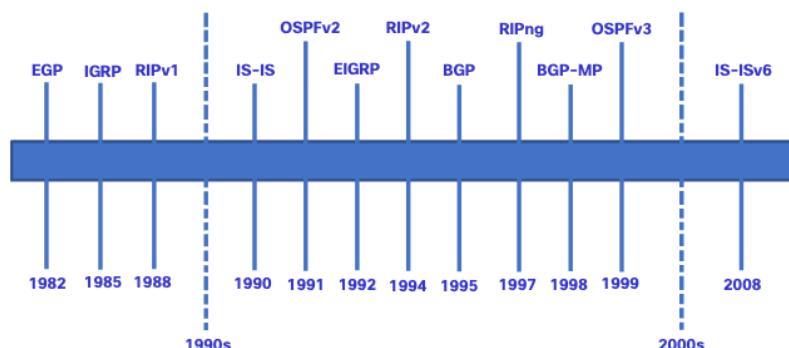
Vlastnosti	Dynamické směrování	Statické směrování
Složitost konfigurace	Nezávislé na velikosti sítě	Zvyšuje se s velikostí sítě
Změny topologie	Automaticky se přizpůsobuje změnám topologie	Je nutný zásah správce
Škálovatelnost	Vhodné od jednoduchých až po složité topologie sítě	Vhodné pro jednoduché topologie
Bezpečnost	Musí být nakonfigurováno zabezpečení	Zabezpečení je inherentní
Využití zdrojů	Využívá CPU, paměť a šířku pásmo linky	Nejsou potřeba žádné další zdroje
Předvídatelnost cesty	Trasa závisí na topologii a použitém směrovacím protokolu	Explicitně definováno správcem

Evoluce dynamického směrování

Statické nastavení cest je výchozí variantou, základem. Dynamické protokoly se s časem vyvíjely a nejen proto, že k IPv4 přibyl IPv6 protokol. Protokoly dynamického směrování se v sítích používají od konce 80. let. Jedním z prvních směrovacích protokolů byl RIP.

RIPv1 byl oficiálně vydán v roce 1988, ale některé jeho základní algoritmy byly použity již v síti Advanced Research Projects Agency Network (ARPANET) již v roce 1969.

Jak se síť vyvíjely a staly se složitějšími, objevily se nové směrovací protokoly nebo jejich verze, jak konečně můžeme vidět na časové ose na snímku.



K evoluci dynamických směrovacích protokolů se váže i následující tabulka. Klasifikuje aktuální směrovací protokoly.

	Interior Gateway Protocols				Exterior Gateway Protocols
	Distance Vector		Link-State		Path Vector
IPv4	RIPv2	EIGRP	OSPFv2	IS-IS	BGP-4
IPv6	RIPng	EIGRP for IPv6	OSPFv3	IS-IS for IPv6	BGP-MP

Protokoly vnitřní brány (Interior Gateway Protocols - IGP) jsou směrovací protokoly používané k výměně směrovacích informací v rámci směrovací domény spravované jedinou organizací.

Existuje pouze jeden Exterior Gateway Protocol (EGP), a to BGP. Používá se k výměně směrovacích informací mezi různými organizacemi, známými jako autonomní systémy (AS). BGP používají ISP k směrování paketů přes internet. Směrovací protokoly vektoru vzdálenosti (Distance Vector), stavu spojení (Link-State) a vektoru cesty (Path Vector) odkazují na typ směrovacího algoritmu použitého k určení nejlepší cesty.

Koncepty protokolu dynamického směrování

Směrovací protokol lze definovat jako sadu procesů, algoritmů a zpráv, které se používají k výměně směrovacích informací a k naplnění směrovací tabulky výběrem nejlepších cest. Účel protokolů dynamického směrování zahrnuje následující:

- Objev vzdálených sítí
- Údržba aktuálních informací o směrování
- Výběr nejlepší cesty k cílovým sítím
- Možnost najít novou nejlepší cestu, pokud aktuální cesta již není k dispozici

Mezi hlavní součásti protokolů dynamického směrování patří následující trojice:

- **Datové struktury** - Směrovací protokoly obvykle používají pro své operace tabulky nebo databáze. Tyto informace se uchovávají v paměti RAM.
- **Zprávy směrovacího protokolu** - Směrovací protokoly používají různé typy zpráv k objevování sousedních směrovačů, výměně směrovacích informací a dalších úkolů k učení a udržování přesných informací o síti.
- **Algoritmus** - Algoritmus je konečný seznam kroků použitých k provedení úkolu. Směrovací protokoly používají algoritmy pro usnadnění směrovacích informací a pro nejlepší určení cesty.

Směrovací protokoly určují nejlepší cestu nebo trasu do každé sítě. Tato trasa je poté nabídnuta směrovací tabulce. Trasa bude nainstalována do směrovací tabulky, pokud není k dispozici jiný zdroj směrování s nižší administrativní vzdáleností.

Nejlepší cesta

Nejlepší cesta je vybrána směrovacím protokolem na základě ohodnocení - hodnoty administrativní vzdálenosti, nebo metriky, kterou používá k určení vzdálenosti k dosažení sítě. Metrika je kvantitativní hodnota používaná k měření vzdálenosti k dané síti. Nejlepší cesta k síti je cesta s nejnižší metrikou.

Dynamické směrovací protokoly obvykle používají svá vlastní pravidla a metriky k vytváření a aktualizaci směrovacích tabulek. V následující tabulce jsou uvedeny běžné dynamické protokoly a jejich metriky.

Směrovací protokol	Metrika
Routing Information Protocol (RIP)	<ul style="list-style-type: none">• Metrika je "počet skoků (hop count)".• Každý směrovač na cestě přidává k počítadlu jeden skok.• Povolen je maximálně 15 skoků.
Open Shortest Path First (OSPF)	<ul style="list-style-type: none">• Metrika jsou "náklady", které jsou založeny na kumulativní šířce pásma od zdroje k cíli.• Rychlejší spojení mají nižší náklady než pomalejší spojení.
Enhanced Interior Gateway Routing Protocol (EIGRP)	<ul style="list-style-type: none">• Vypočítá metriku na základě nejpomalejší šířky pásma a zpoždění.• Může také zahrnovat zatížení a spolehlivost do výpočtu metriky.

Podrobnější rozbor a implementace dynamických routovacích protokolů Je náplní předmětu PSIT3.

Vyrovnávání zatížení

Pro předávání paketů může být využito více cest současně, což nazýváme loadbalancingem neboli vyrovnáváním zátěže. Jde o to, že pokud má směrovač dvě nebo více cest k cíli se stejnými metrikami nákladů, přeposílá pakety pomocí obou cest stejně. Tomu se říká vyrovnávání zátěže se stejnými náklady.

- Směrovací tabulka obsahuje jednu cílovou síť, ale má více výstupních rozhraní, jedno pro každou cestu se stejnými náklady. Směrovač předává pakety pomocí více výstupních rozhraní uvedených ve směrovací tabulce.
- Pokud je správně nakonfigurováno, vyrovnávání zatížení může zvýšit efektivitu a výkon sítě.
- Vyrovnávání zatěže se stejnými náklady je implementováno automaticky pomocí protokolů dynamického směrování. Je povoleno i se statickými trasami, pokud existuje více statických tras do stejné cílové sítě pomocí různých směrovačů dalšího směrování.

Vyrovnávání zatížení s nestejnými náklady podporuje pouze protokol EIGRP.

Počítačové sítě 2

13, IP Statické směrování

Název tématu	Cíl
Statické trasy	Popsat syntaxi příkazu pro statické trasy.
Konfigurace statické trasy IP	Nakonfigurovat statické trasy IPv4 a IPv6.
Konfigurace výchozí statické trasy IP	Nakonfigurovat výchozí statické trasy IPv4 a IPv6.
Konfigurace plovoucí statické trasy	Nakonfigurovat plovoucí statickou trasu tak, aby poskytovala záložní připojení.
Konfigurace statické trasy hostitele	Nakonfigurovat statické trasy hostitele IPv4 a IPv6, které směrují provoz na konkrétního hostitele.

13.1 Statické trasy

Typy statických tras

Statické trasy jsou záznamy manuálně zadané do směrovací tabulky a jsou v sítích běžně implementovány. Platí to, i když je nakonfigurován protokol dynamického směrování.

Lze je konfigurovat jak pro IPv4, tak IPv6. Oba protokoly podporují následující typy statických cest:

- Standardní statická trasa
- Výchozí statická trasa
- Plovoucí statická trasa
- Souhrnná statická trasa

Všechny typy statických tras se konfigurují pomocí globálních konfiguračních příkazů **ip route** a **ipv6 route**.

Možnosti dalšího skoku (Next-Hop Options)

Při konfiguraci statické trasy lze další směrování identifikovat pomocí IP adresy, výstupního rozhraní nebo obou těchto údajů. Jak je zadán cíl, určuje jeden ze tří následujících typů statické tras:

- **Next-hop route** - Je zadána pouze IP adresa dalšího rozhraní
- **Přímo připojená statická trasa** - je zadáno pouze výstupní rozhraní směrovače
- **Plně specifikovaná statická trasa** - je zadána adresa IP dalšího rozhraní a výstupní rozhraní

Příkaz statické trasy IPv4

Statické trasy IPv4 se konfigurují pomocí následujícího globálního konfiguračního příkazu:

ip route network-address subnet-mask { ip-address | exit-intf [ip-address]} [distance]

Nakonfigurované musí být parametry buď *ip-address* nebo *exit-intf*, a nebo dvojice *ip-address* a *exit-intf*.

Příkaz statické trasy IPv6

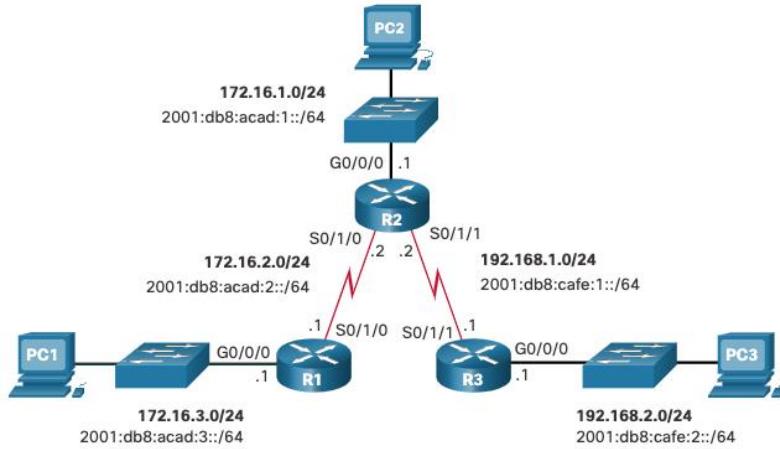
Statické trasy IPv6 protokolu se konfigurují pomocí globálního konfiguračního příkazu:

ipv6 route ipv6-prefix/prefix-length {ipv6-address | exit-intf [ipv6-address]} [distance]

Většina parametrů je shodná s verzí příkazu IPv4, rozdíl je, že zde uvádíme prefix namísto masky.

Dual-Stack topologie

Následující postup je postaven na příkladu dual-stack topologie sítě uvedené na obrázku níže. V této fázi ještě nejsou nakonfigurovány žádné statické trasy pro IPv4 ani IPv6.



IPv4 Vytváření směrovací tabulky

Když si provedeme výpis `show ip route` s parametrem `begin Gateway`, pak vidíme, že

- směrovač obsahuje položky pouze pro přímo připojené sítě a přidružené místní adresy.
- R1 může provést ping na R2, ale nemůže udělat ping na R3 LAN.

```

R1# show ip route | begin Gateway
Gateway of last resort is not set
  172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C    172.16.2.0/24 is directly connected, Serial0/1/0
L    172.16.2.1/32 is directly connected, Serial0/1/0
C    172.16.3.0/24 is directly connected, GigabitEthernet0/0/0
L    172.16.3.1/32 is directly connected, GigabitEthernet0/0/0
R1#
R1# ping 172.16.2.2
Type escape sequence to abort. Sending 5, 100-byte ICMP Echos to 172.16.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5)
R1# ping 192.168.2.1
Type escape sequence to abort. Sending 5, 100-byte ICMP Echos to 192.168.2.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
  
```

IPv6 Vytváření směrovací tabulky

Taktéž pro IPv6 adresní prostor obsahuje každý směrovač položky pouze pro přímo připojené sítě a přidružené místní adresy.

R1 může provést příkaz ping na R2, ale nemůže jej provést na R3 LAN.

```

R1# show ipv6 route | begin C
C 2001:DB8:ACAD:2::/64 [0/0]
  via Serial0/1/0, directly connected
L 2001:DB8:ACAD:2::1/128 [0/0]
  via Serial0/1/0, receive
C 2001:DB8:ACAD:3::/64 [0/0]
  via GigabitEthernet0/0/0, directly connected
L 2001:DB8:ACAD:3::1/128 [0/0]
  via GigabitEthernet0/0/0, receive
L FF00::/8 [0/0]
  via Null0, receive
R1#
R1# ping 2001:db8:acad:2::2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:ACAD:2::2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms)
R1# ping 2001:DB8:cafe:2::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:CAFE:2::1, timeout is 2 seconds:
% No valid route for destination
Success rate is 0 percent (0/1)
  
```

13.2 Konfigurace statické trasy IP

IPv4 Next-Hop statická trasa

Standardní statická trasa využívá next hop neboli dalšího skoku. Rozumíme tím, že je za adresou sítě zadána pouze IP adresa rozhraní následujícího uzlu. Můžeme říci, že výstupní rozhraní je odvozeno od dalšího skoku.

Například ve výpisu na snímku jsou uvedeny tři statické next-hop trasy protokolu IPv4 z roteru R2. Ty jsou na R1 konfigurovány pomocí IP adresy dalšího skoku, R2:

```
R1(config)# ip route 172.16.1.0 255.255.255.0 172.16.2.2
```

```
R1(config)# ip route 192.168.1.0 255.255.255.0 172.16.2.2
```

```
R1(config)# ip route 192.168.2.0 255.255.255.0 172.16.2.2
```

Výsledné položky směrovací tabulky na R1 jsou markantní na výpisu na snímku.

```
R1# show ip route | begin Gateway
Gateway of last resort is not set
    172.16.0.0/16 is variably subnetted, 5 subnets, 2 masks
S      172.16.1.0/24 [1/0] via 172.16.2.2
C      172.16.2.0/24 is directly connected, Serial0/1/0
L      172.16.2.1/32 is directly connected, Serial0/1/0
C      172.16.3.0/24 is directly connected, GigabitEthernet0/0/0
L      172.16.3.1/32 is directly connected, GigabitEthernet0/0/0
S      192.168.1.0/24 [1/0] via 172.16.2.2
S      192.168.2.0/24 [1/0] via 172.16.2.2
```

IPv6 Next-Hop statická trasa

Použijeme následující příkazy ke konfiguraci R1 se statickými IPv6 trasami do tří vzdálených sítí pro shodnou topologii. Nejdříve je ale třeba zapnout ipv6 routování příkazem:

```
R1(config)# ipv6 unicast-routing
```

Pak následuje konfigurace směrů

```
R1(config)# ipv6 route 2001:db8:acad:1::/64 2001:db8:acad:2::2
```

```
R1(config)# ipv6 route 2001:db8:cafe:1::/64 2001:db8:acad:2::2
```

```
R1(config)# ipv6 route 2001:db8:cafe:2::/64 2001:db8:acad:2::2
```

Směrovací tabulka uvedená níže nyní obsahuje trasy do tří vzdálených sítí IPv6 na R1 .

```
R1# show ipv6 route
IPv6 Routing Table - default - 8 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
      B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
      I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
      EX - EIGRP external, ND - ND Default, NDP - ND Prefix, DCE - Destination
      NDr - Redirect, RL - RPL, O - OSPF Intra, OI - OSPF Inter
      OE1 - OSPF ext 1, OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1
      ON2 - OSPF NSSA ext 2, la - LISP alt, lr - LISP site-registrations
      ld - LISP dyn-eid, lA - LISP away, le - LISP extranet-policy
      a - Application
S  2001:DB8:ACAD:1::/64 [1/0]
  via 2001:DB8:ACAD:2::2
C  2001:DB8:ACAD:2::/64 [0/0]
  via Serial0/1/0, directly connected
L  2001:DB8:ACAD:2::1/128 [0/0]
  via Serial0/1/0, receive
C  2001:DB8:ACAD:3::/64 [0/0]
  via GigabitEthernet0/0/0, directly connected
L  2001:DB8:ACAD:3::1/128 [0/0]
  via GigabitEthernet0/0/0, receive
S  2001:DB8:CAFE:1::/64 [1/0]
  via 2001:DB8:ACAD:2::2
S  2001:DB8:CAFE:2::/64 [1/0]
  via 2001:DB8:ACAD:2::2
L  FF00::/8 [0/0]
  via Null0, receive
```

IPv4 Přímo připojená statická trasa

Zatímco v next hop konfiguraci využíváme k určení adresy dalšího skoku IP adresu rozhraní Routeru, přes který se směrování pro danou síťovou adresu provede, v přímo připojené statické adrese toto nahradí identifikátor místního rozhraní na konfigurovaném prvku.

Na příkladu jsou tři přímo připojené statické cesty IPv4 konfigurovány na R1 pomocí výstupního rozhraní.

```
R1(config)# ip route 172.16.1.0 255.255.255.0 s0/1/0
```

```
R1(config)# ip route 192.168.1.0 255.255.255.0 s0/1/0
```

```
R1(config)# ip route 192.168.2.0 255.255.255.0 s0/1/0
```

Poznámka: Obecně se doporučuje používat adresu dalšího skoku. Přímo připojené statické trasy by se měly používat pouze se sériovými rozhraními typu point-to-point.

IPv6 Přímo připojená statická trasa

Pro shodný příklad jsou pak použity tři přímo připojené statické trasy IPv6. Na R1 jsou konfigurovány pomocí výstupního rozhraní.

Obecně se doporučuje používat adresu dalšího skoku. Přímo připojené statické trasy by se měly používat pouze se sériovými rozhraními typu point-to-point.

```
ipv6 route 2001:db8:acad:1::/64 s0/1/0
```

```
ipv6 route 2001:db8:cafe:1::/64 s0/1/0
```

```
ipv6 route 2001:db8:cafe:2::/64 s0/1/0
```

IPv4 Plně specifikovaná statická trasa

V plně specifikované statické trase je pak určeno výstupní rozhraní a IP adresa dalšího skoku.

Tato forma statické trasy se používá, když je výstupním rozhraním rozhraní s více přístupy a je nutné explicitně identifikovat další směrování. Další směrování musí být přímo připojeno k uvedenému výstupnímu rozhraní. Použití výstupního rozhraní je volitelné, je však nutné použít adresu dalšího skoku.

Pokud je výstupním rozhraním síť

Ethernet, doporučuje se, aby statická trasa obsahovala adresu dalšího skoku. Můžete také použít plně specifikovanou statickou trasu, která zahrnuje jak výstupní rozhraní, tak adresu dalšího skoku.

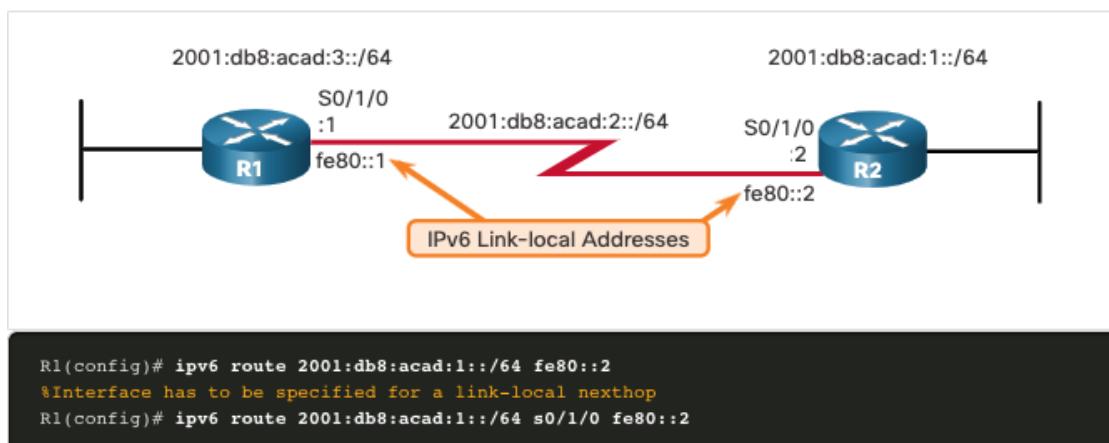
```
R1# show ip route | begin Gateway
Gateway of last resort is not set
      172.16.0.0/16 is variably subnetted, 5 subnets, 2 masks
S       172.16.1.0/24 [1/0] via 172.16.2.2, GigabitEthernet0/0/1
C       172.16.2.0/24 is directly connected, GigabitEthernet0/0/1
L       172.16.2.1/32 is directly connected, GigabitEthernet0/0/1
C       172.16.3.0/24 is directly connected, GigabitEthernet0/0/0
L       172.16.3.1/32 is directly connected, GigabitEthernet0/0/0
S       192.168.1.0/24 [1/0] via 172.16.2.2, GigabitEthernet0/0/1
S       192.168.2.0/24 [1/0] via 172.16.2.2, GigabitEthernet0/0/1
```

IPv6 Plně specifikovaná statická trasa

V plně specifikované statické trase je určeno výstupní rozhraní a adresa IPv6 dalšího skoku.

V IPv6 je situace, kdy je nutné použít plně specifikovanou statickou cestu. Pokud statická trasa IPv6 používá jako adresu dalšího směrování místní (link-local) adresu IPv6, použijte plně specifikovanou statickou trasu.

Obrázek ukazuje příklad plně specifikované statické trasy IPv6 s použitím místní adresy IPv6 jako adresy dalšího skoku.



Důvodem, proč je nutné použít plně specifikovanou statickou trasu, je to, že místní adresy IPv6 nejsou obsaženy ve směrovací tabulce IPv6. Místní adresy jsou jedinečné pouze na dané lince nebo síti. Místní adresa dalšího skoku může být platná adresa ve více sítích připojených ke směrovači. Proto je nutné zahrnout výstupní rozhraní.

Následující příklad ukazuje položku směrovací tabulky IPv6 pro tuto trasu. Všimněte si, že položka obsahuje jak místní adresu dalšího skoku, tak výstupní rozhraní.

```
R1# show ipv6 route static | begin 2001:db8:acad:1::/64
S   2001:DB8:ACAD:1::/64 [1/0]
    via FE80::2, Serial0/1/0
```

Ověření statické trasy

Posledním krokem v konfiguraci je pak ověření nakonfigurované statické trasy. Kromě všeobecně známých příkazů **show ip route**, **show ipv6 route**, **ping** a **traceroute** jsou k dispozici následující příkazy pro ověření statických tras:

- **show ip route static**
- **show ip route network**
- **show running-config | section ip route**

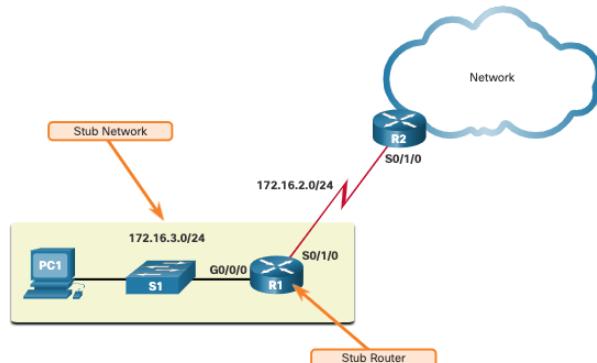
Pro verze příkazu IPv6 nahraďte **ip** za **ipv6**.

13.3 Konfigurace výchozí statické trasy IP

Výchozí statická trasa

Co rozumíme pod termínem Výchozí statická trasa a k čemu slouží:

- Výchozí trasa je statická trasa, která odpovídá všem paketům. Jedna výchozí trasa představuje jakoukoli síť, která není ve směrovací tabulce.
- Směrovače běžně používají výchozí trasy, které jsou buď konfigurovány místně, nebo se naučily od jiného směrovače. Výchozí trasa se používá jako brána poslední instance.
- Výchozí statické trasy se běžně používají při připojení hraničního směrovače k síti poskytovatele služeb nebo pahýlového směrovače (směrovač pouze s jedním sousedním upstream směrovačem).
- Obrázek ukazuje typický scénář použití výchozí statické trasy.



Syntaxe příkazu pro výchozí statickou trasu IPv4 protokolu je podobná jakékoli jiné statické trase, až na to, že síťová adresa a maska podsítě je **0.0.0.0**. Hodnota 0.0.0.0 0.0.0.0 na trase bude odpovídat jakékoli síťové adrese.

Výchozí statická trasa IPv4 protokolu se běžně označuje jako trasa se čtyřmi nulami.

Základní syntaxe příkazu pro výchozí statickou cestu IPv4 je pak **ip route 0.0.0.0 0.0.0.0 {ip-address | exit-intf}**

U protokolu IPv6 je Syntaxe příkazu pro výchozí statickou trasu podobná jakékoli jiné jeho statické trase, až na to, že ipv6-prefix/prefix-length je ::/0, což odpovídá všem trasám.

Základní syntaxe příkazu pro výchozí statickou cestu IPv6 je následující:

ipv6 route ::/0 {ipv6-address | exit-intf}

Konfigurace výchozí statické trasy

Ještě si uvedeme reálný příklad. Ten ukazuje výchozí statickou IPv4 trasu nakonfigurovanou na R1.

V konfiguraci zobrazené na snímku jsou všechny pakety, které neodpovídají konkrétnějším trasám, předány směrovači R2 na rozhraní 172.16.2.2.

ip route 0.0.0.0 0.0.0.0 172.16.2.2

Výchozí IPv6 statická trasa je nakonfigurována podobným způsobem. Také s touto konfigurací jsou všechny pakety, které neodpovídají konkrétnějším položkám směrovací tabulky IPv6 protokolu, přeposílaný na R2 rozhraní 2001:db8:acad:2::2.

Syntaxe příkazu je:

ipv6 route ::/0 2001:db8:acad:2::2

Ověření výchozí statické trasy

Výstup příkazu **show ip route static** z R1 zobrazuje obsah statických tras ve směrovací tabulce.

Všimněte si hvězdičky (*) vedle trasy s kódem „S“. Hvězdička označuje, že tato statická trasa je kandidátskou výchozí trasou, a proto je vybrána jako brána poslední instance.

Všimněte si, že konfigurace výchozí statické trasy používá pro výchozí trasy IPv4 masku /0. Nezapomeňte, že maska podsíť IPv4 ve směrovací tabulce určuje, kolik bitů se musí shodovat mezi cílovou IP adresou paketu a cestou ve směrovací tabulce. Maska /0 označuje, že ke shodě není potřeba žádný bit. Pokud neexistuje konkrétnější shoda, výchozí statická trasa odpovídá všem paketům.

```
R1# show ip route static
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
+ - replicated route, % - next hop override

Gateway of last resort is 172.16.2.2 to network 0.0.0.0

S* 0.0.0.0/0 [1/0] via 172.16.2.2
```

Tento příklad ukazuje výstup příkazu **show ipv6 route static** pro zobrazení obsahu směrovací tabulky.

Všimněte si, že konfigurace statické výchozí trasy používá předvolbu ::/0 pro výchozí trasy IPv6. Pamatujte si, že délka předpony IPv6 ve směrovací tabulce určuje, kolik bitů se musí shodovat mezi cílovou IP adresou paketu a cestou ve směrovací tabulce. Předpona ::/0 označuje, že ke shodě není vyžadován žádný bit. Pokud neexistuje konkrétnější shoda, výchozí statická trasa odpovídá všem paketům.

```
R1# show ipv6 route static
IPv6 Routing Table - default - 8 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
      B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
      I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
      EX - EIGRP external, ND - ND Default, NDP - ND Prefix, DCE - Destination
      NDr - Redirect, RL - RPL, O - OSPF Intra, OI - OSPF Inter
      OE1 - OSPF ext 1, OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1
      ON2 - OSPF NSSA ext 2, la - LISP alt, lr - LISP site-registrations
      ld - LISP dyn-eid, IA - LISP away, le - LISP extranet-policy
      a - Application

S  ::/0 [1/0]
    via 2001:DB8:ACAD:2::2
```

13.4 Konfigurace plovoucí statické trasy

Plovoucí statické trasy

Dalším typem statické trasy je plovoucí statická trasa. Jde o statické trasy, které se používají k poskytnutí záložní cesty k primární statické nebo dynamické trase. Plovoucí statická trasa se používá pouze v případě, že primární trasa není k dispozici.

K dosažení tohoto cíle je plovoucí statická trasa nakonfigurována s vyšší administrativní vzdáleností než primární trasa. Administrativní vzdálenost představuje důvěryhodnost trasy. Pokud existuje více cest k cíli, router vybere cestu s nejnižší administrativní vzdáleností.

Ve výchozím nastavení mají statické trasy administrativní vzdálenost 1, což je činí vhodnějšími než trasy naučené z protokolů dynamického směrování.

Administrativní vzdálenost statické trasy lze zvýšit, aby byla trasa méně žádoucí než vzdálenost jiné statické trasy nebo trasy naučené prostřednictvím protokolu dynamického směrování. Tímto způsobem statická trasa „plave“ a nepoužívá se, když je aktivní trasa s lepší administrativní vzdáleností.

Konfigurace IPv4 a IPv6 plovoucích statických adres

Příkazy ke konfiguraci výchozích a plovoucích výchozích tras IP jsou následující:

```
R1(config)# ip route 0.0.0.0 0.0.0.0 172.16.2.2
```

```
R1(config)# ip route 0.0.0.0 0.0.0.0 10.10.10.2 5
```

```
R1(config)# ipv6 route ::/0 2001:db8:acad:2::2
```

```
R1(config)# ipv6 route ::/0 2001:db8:feed:10::2 5
```

Hlavní rozdíl syntaxe mezi první a druhou adresou a třetí a čtvrtou je, že u druhé a čtvrté je mimo běžně nastavovaných parametrů se zde na samém konci zápisu řádku uvádí administrativní vzdálenost.

Výstupy příkazů **show ip route** a **show ipv6 route** ověří, že jsou ve směrovací tabulce nainstalovány výchozí trasy do R2.

Test plovoucí statické trasy

Co by se stalo, kdyby R2 selhal?

- Abychom to simulovali, vypneme na R2 obě sériová rozhraní.
- R1 automaticky generuje zprávy syslog o linkách, které se vypínají.
- Náhledem do směrovací tabulky R1 bychom našli použitou sekundární trasu.

13.5 Konfigurace statické trasy hostitele

Hostitelské trasy

Hostitelská trasa je adresa IPv4 s 32bitovou maskou nebo adresa IPv6 se 128bitovou maskou.

Následující tři způsoby ukazují, jak lze přidat trasu hostitele do směrovací tabulky:

- Je Automaticky nainstalována, když je na routeru nakonfigurována IP adresa

- Nakonfigurována statická cesta hostitele
- Nebo to může být
- Hostitelská trasa automaticky získaná jinými metodami (popsáno v dalších kurzech)

Automaticky nainstalované hostitelské trasy

Jak bylo naznačeno, kdy jsou nainstalované hostitelské trasy automaticky. Jde o případy

- Když je na směrovači nakonfigurována adresa rozhraní, Cisco IOS automaticky nainstaluje hostitelskou trasu, známou také jako místní hostitelská trasa. Trasa hostitele umožňuje efektivnější procesování paketů, které jsou směrovány na samotný směrovač, než pro předávání paketů mimo něj.
- Toto se přidá navíc k připojené trase, která je ve směrovací tabulce označena písmenem **C** pro síťovou adresu rozhraní.
- Místní trasy jsou na výstupu směrovací tabulky označeny písmenem **L**.

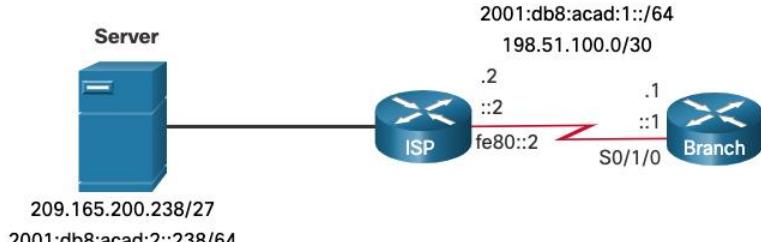
Statické trasy hostitele

Hostitelská trasa může být ručně nakonfigurovaná statická trasa k směrování provozu na konkrétní cílové zařízení, jako je server zobrazený na obrázku. Statická trasa používá cílovou IP adresu a masku 255.255.255.255 (/32) pro IPv4 trasy hostitele a délku prefixu /128 pro IPv6 trasy hostitele.

```

graph LR
    Server[Server] --- ISP((ISP))
    ISP --- Branch[Branch]
    style Server fill:#4682B4,color:#FFF
    style ISP fill:#4682B4,color:#FFF
    style Branch fill:#4682B4,color:#FFF
    
```

209.165.200.238/27
fe80::2
198.51.100.0/30
::1
S0/1/0



Příklad ukazuje takovou konfiguraci statické trasy hostitele pro protokoly IPv4 i IPv6 na směrovači pojmenovaný Branch pro přístup k serveru.

```
Branch(config)# ip route 209.165.200.238 255.255.255.255 198.51.100.2
```

```
Branch(config)# ipv6 route 2001:db8:acad:2::238/128 2001:db8:acad:1::2
```

```
Branch(config)# exit
```

```
Branch#
```

Je zde uvedena IP adresa koncového uzlu s maskou, respektive prefixem, a adresou rozhraní, přes které je dostupná.

Ověření statické trasy hostitele

Na obrázku vidíme výpis z kontroly obou směrovacích tabulek IPv4 a IPv6, která ukazuje, že trasy jsou aktivní.

```

Branch# show ip route | begin Gateway
Gateway of last resort is not set
  198.51.100.0/24 is variably subnetted, 2 subnets, 2 masks
C    198.51.100.0/30 is directly connected, Serial0/1/0
L    198.51.100.1/32 is directly connected, Serial0/1/0
  209.165.200.0/32 is subnetted, 1 subnets
S      209.165.200.238 [1/0] via 198.51.100.2

Branch# show ipv6 route
(Output omitted)
C  2001:DB8:ACAD:1::/64 [0/0]
    via Serial0/1/0, directly connected
L  2001:DB8:ACAD:1::1/128 [0/0]
    via Serial0/1/0, receive
S  2001:DB8:ACAD:2::238/128 [1/0]
    via 2001:DB8:ACAD:1::2

Branch#

```

Konfigurace statické trasy hostitele IPv6 pomocí Link-Local Next-Hop

Konfigurace statické trasy hostitele s použitím protokolu IPv6 může proběhnout také pomocí Link-Local Next-Hop adresy. Pro statické trasy IPv6 může být adresou dalšího skoku (next-hop) adresa místního spoje (link-local) sousedního směrovače. Pokud však jako další směrování použijete adresu místní linky, musíte zadat typ rozhraní a číslo rozhraní, jak je znázorněno v příkladu. Nejprve je odstraněna původní statická trasa hostitele IPv6, poté je plně specifikovaná trasa nakonfigurovaná s IPv6 adresou serveru a IPv6 adresou místní linky směrovače ISP.

13.6 Odstraňování problémů se statickými a výchozími trasami

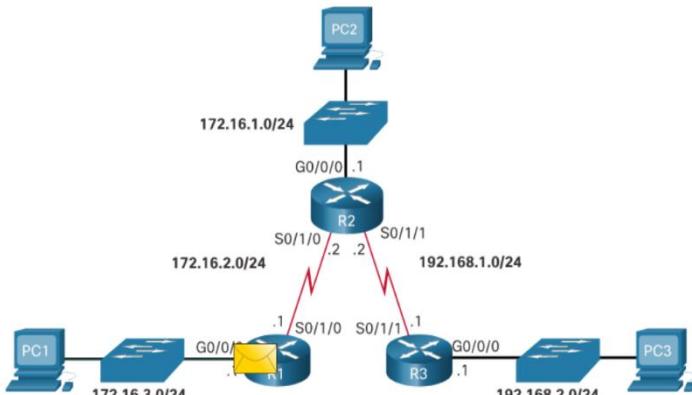
Zpracování paketů v případě statických tras

Statické trasy a předávání paketů

Pro správné provedení konfigurace a případné hledání chyb nastavení je potřeba znát, jak se definují a vyhodnocují statické trasy, a jak na jejich základě probíhá předávání paketů.

K tomu nám poslouží příklad topologie, který si lze připomenout na obrázku na snímku vpravo. Situace je postavena na komunikaci PC1 a PC3.

- PC1 adresuje paket na PC3 a odešle jej na adresu výchozí brány.
- Když paket dorazí na rozhraní G0/0/0 Routeru R1, router paket rozbalí a prohledá směrovací tabulku pro odpovídající položku cílové sítě.



Pokud se cílová IP adresa shoduje s položkou statické trasy, R1 použije statickou trasu k identifikaci IP adresy dalšího skoku nebo výstupního rozhraní.

- Neodpovídá-li konkrétní trase do cílové sítě, pak R1 použije výchozí statickou trasu, je-li tato nakonfigurována.
- Pokud se neshoduje se s žádnou položkou směrovací tabulky a konfigurace výchozí cesty nebyla provedena, pak R1 zahodí paket a odešle zprávu ICMP zpět ke zdroji (tj. PC1).

Za předpokladu, že R1 našel odpovídající položku ve směrovací tabulce, zapouzdří paket do nového rámce a přepoše je z rozhraní S0/1/0 na R2.

- R2 přijímá paket na svém rozhraní S0/1/0.
- Rozbalí a zpracovává paket stejným způsobem jako R1.
- Když R2 najde shodu ve směrovací tabulce, použije identifikovanou adresu IP dalšího skoku nebo výstupní rozhraní a odešle paket ze svého rozhraní S0/1/1 směrem k R3.
- R3 přijímá paket, rozbalí jej a hledá ve směrovací tabulce shodu.
- Cílová IP adresa PC3 odpovídá přímo připojenému rozhraní G0/0/0. Tudíž R3 prohledá ARP tabulku s cílem nalézt MAC adresu 2 vrstvy PC3.

- Pokud neexistuje žádná položka ARP, potom R3 odešle požadavek ARP z rozhraní G0/0/0.
- PC3 reaguje ARP odpověď obsahující jeho MAC adresu.
- R3 zapouzdřuje paket do nového rámce a používá MAC adresu PC3 jako cílovou MAC adresu a MAC adresu G0/0/0 jako zdrojovou MAC adresu.
- Rámcem je přeposlán z rozhraní G0/0/0 a PC3 jej podle toho přijímá a zpracovává.

Paket byl doručený. Je třeba si uvědomit, že záznam v routovací tabulce musí obsahovat záznamy pro oboustrannou komunikaci, a že u komplexnějších sítí, na základě hodnoty administrativní vzdálenosti, může být cesta vedena přes jiné routery. To je třeba pak brát na zřetel při řešení problémů.

13.7 Řešení potíží s konfigurací statické a výchozí trasy IPv4

Změny v síti

Sítě selhávají z mnoha důvodů:

- Může selhat rozhraní
- Poskytovatel služeb ukončí připojení
- Linky mohou být přetížené
- Správce může zadat nesprávnou konfiguraci.

Za určení a vyřešení problému jsou zodpovědní správci sítě. Ti musí jednat rychle.

Pro efektivní nalezení problémů se směrováním a jejich rychlého vyřešení je výhodné se důkladně seznámit s nástroji, které je pomáhají rychle izolovat.

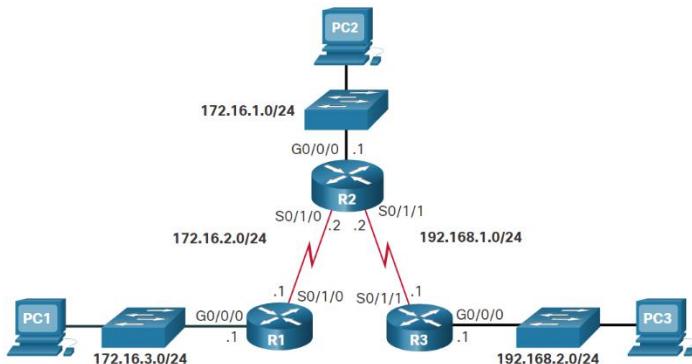
Běžné příkazy pro řešení potíží

Příkaz	Popis
ping	<ul style="list-style-type: none"> • Ověří připojení k cíli na 3. vrstvě. • Rozšířené pingy poskytují další možnosti.
traceroute	<ul style="list-style-type: none"> • Ověří trasu do cílové sítě. • Využívá ICMP echo reply zprávy k určení skoků do cíle.
show ip route	<ul style="list-style-type: none"> • Zobrazí směrovací tabulku. • Používá se k ověření záznamů trasy pro cílové adresy IP.
show ip interface brief	<ul style="list-style-type: none"> • Zobrazí stav rozhraní daného zařízení. • Slouží k ověření provozního stavu a IP adresy na rozhraní.
show cdp neighbors	<ul style="list-style-type: none"> • Zobrazí seznam přímo připojených Cisco zařízení. • Používá se také k ověření konektivity na vrstvách 1 a 2.

Řešení problému s připojením

Na příkladu vidíme, že připojení z PC1 na PC3 selhalo. Je potřeba postupovat systematicky.

- Admin použije rozšířené pingy z rozhraní R1 G0/0/0 směrem na PC3, ty selžou.
- Následné pingy z R1 (tj. rozhraní S0/1/0) na libovolné rozhraní R2 jsou úspěšné.
- Stejně tak pingy z R1 (tj. rozhraní S0/1/0) na R3 jsou úspěšné.



Zaměříme se proto na směrovací tabulku směrovače R2. Zde se příkazem

show ip route | begin Gateway

```
R2# show ip route | begin Gateway
Gateway of last resort is not set
    172.16.0.0/16 is variably subnetted, 5 subnets, 2 masks
C          172.16.1.0/24 is directly connected,
GigabitEthernet0/0/0
L          172.16.1.1/32 is directly connected,
GigabitEthernet0/0/0
C          172.16.2.0/24 is directly connected, Serial0/1/0
L          172.16.2.2/32 is directly connected, Serial0/1/0
S  172.16.3.0/24 [1/0] via 192.168.1.1
    192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C          192.168.1.0/24 is directly connected, Serial0/1/1
L          192.168.1.2/32 is directly connected, Serial0/1/1
S  192.168.2.0/24 [1/0] via 192.168.1.1
```

odhalí problém, kterým je nastavení nesprávné statické trasy.

- Problém vyřeší odstranění chybného záznamu příkazem **no ip route 172.16.3.0 255.255.255.0** v globálním konfiguračním režimu
- Vytvoření nové statické trasy příkazem **ip route 172.16.3.0 255.255.255.0 172.16.2.1**

Následným pingem ověříme funkčnost trasy.

Ne vždy musí být situace takto jednoduchá, častěji narazíte na problémy složitější, skryté (vadný kabel konektoru, chybné rozhraní – neexistují jen stavy funkční a nefunkční, může jít o nahodilé výpadky díky chybě na sběrnici Routeru a další závady, které se snáze řeší s nabýtými zkušenostmi z provozu.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenční schopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Tato publikace je spolufinancována Evropským sociálním fondem a státním rozpočtem České republiky

Registrační číslo projektu: CZ.1.07/1.1.12/01.0004

Studijní materiál

CCNA Exploration – Směrování, koncepce a protokoly

(Semestr 2)



VOŠ a SPŠE Plzeň

2011

Tato publikace je spolufinancována Evropským sociálním fondem a státním rozpočtem České republiky v rámci projektu „Výuka počítačových sítí v mezinárodním programu Sítová akademie Cisco na střední průmyslové škole elektrotechnické“.

Registrační číslo projektu: CZ.1.07/1.1.12/01.0004.

Vydala VOŠ a SPŠE Plzeň, Kotterovská 85, 326 00 Plzeň v roce 2011.

Kolektiv autorů (řešitelé projektu):

- **Koncepce a text:** Ing. Miroslav Páv
- **Vektorová grafika:** Mgr. Jan Syřínek
- **Konzultace angličtiny:** Mgr. Jana Hošková

- Tato publikace je určena jako doplňkový studijní materiál ke kurzu CCNA Exploration – Routing Protocols and Concepts. Nejedná se o doslovný překlad celého kurikula ale o nově vytvořené vlastní výklady podporující představivost žáků a komentované upravené překlady vybraných částí jednotlivých kapitol anglického kurikula určené pro usnadnění výuky i studia originálního kurzu v prostředí české odborné střední školy.
- Obsah kurzu je integrován v rámci ŠVP naší školy.
- Tento dokument je zpracovaný v kancelářském balíku OpenOffice.org a jeho vektorová grafika v grafickém editoru Dia.
- Protože se jedná o materiál podléhající v rámci projektu průběžné aktualizaci, používejte vždy poslední dostupnou verzi.
- Pro zachování vazby na původní učební materiály (kurikula) jsou u českých termínů uváděny i jejich anglické originály.
- Aktuální verze originálních materiálů v angličtině (pro registrované účastníky programu NetAcad): <http://www.cisco.com/web/learning/netacad/index.html> (<http://cisco.netacad.net>, www.cisco.com/go/netacad , <http://www.cisco.com/edu>)

NEPRODEJNÉ

Prosím, dodržujte licenci pro použití této publikace: **Určeno pro komunitu Sítové akademie Cisco - LCNA a RCNA programu Cisco NetAcad (CNA, CNAP) v ČR i v SR s licencí Creative Commons (Uveďte autora-Neužívejte dílo komerčně-Nezasahujte do díla 3.0 Česko):**



Dílo smíte šířit za těchto podmínek: Uveďte autora, neužívejte dílo komerčně, nezasahujte do díla (viz plný text licence). To znamená, že ve své vlastní Sítové akademii můžete tuto publikaci šířit volně a nekomerčně tak jak je.

Děkujeme. Tato publikace je předmětem průběžné aktualizace, používejte proto poslední dostupnou verzi. V rámci komunity instruktorů Sítové akademie je aktuální elektronická verze této publikace šířena pomocí komunitního portálu iPortal.

Pokud tuto publikaci používáte při své výuce, prosím Vás o informaci o této skutečnosti. Věcné a konstruktivní připomínky, náměty i popřípadě nalezené chyby mi zasílejte, prosím, na adresu: pav@spse.pilsedu.cz, věc: **CCNA_Exploration_2.PDF (verze: 3.06)**.

Vaší spolupráce si vážím a děkuji Vám za ni!

Za kolektiv autorů

Miroslav Páv, CCNA CCAI

Obsah

Předpokládané znalosti.....	6
Směrování, koncepce a protokoly.....	6
Úvod.....	9
Kapitola 1 – Úvod do směrování a přeposílání paketů na směrovači.....	10
Směrovač.....	10
Struktura směrovače.....	10
Směrovače vybírají nejlepší cestu.....	11
CPU a paměti.....	11
Sítový operační systém IOS.....	12
Rozhraní směrovače.....	15
Směrovač na 3. vrstvě OSI modelu.....	16
Základní konfigurace směrovače.....	17
Obsah a tvorba obsahu směrovací tabulky.....	19
Průvodce základní konfigurací (nastavením) směrovače.....	24
Kontrolní opakovací otázky a odpovědi (kvíz):.....	34
Kapitola 2 – Statické směrování.....	35
Role směrovače v síti.....	35
Topologie a tabulka adres.....	35
Použití kabeláže.....	36
Zkoumání obsahu směrovací tabulky a stavu rozhraní.....	38
Průzkum přímo připojených sítí a protokol CDP.....	41
Obsah vyrovnávací paměti ARP na směrovači.....	44
Zjišťování změn ve směrovací tabulce.....	44
Statická cesta s adresou dalšího skoku.....	45
Statická cesta s odchozím rozhraním.....	46
Sumarizace (agregace) cest.....	47
Implicitní cesta.....	48
Definice implicitní sítě.....	49
Správa a modifikace cest.....	50
Hledání a odstraňování chyb statické cesty.....	50
Příkazy pro kapitolu 2, Statické směrování.....	51
Volitelné příkazy pro kapitolu 2, Statické směrování.....	52
Časté a „oblíbené“ chyby.....	54
Testování sítí.....	54
Kontrolní opakovací otázky a odpovědi (kvíz):.....	55
Kapitola 3 - Protokoly pro dynamické směrování.....	57
Účel směrovacích protokolů.....	57
Klasifikace směrovacích protokolů.....	58
Konvergence.....	63
Metriky cest.....	63
Administrativní vzdálenosti protokolů.....	64
Vyrovnávání zátěže.....	66
Identifikace prvků směrovací tabulky.....	66
Kontrolní opakovací otázky a odpovědi (kvíz):.....	66

Kapitola 4 - Směrovací protokoly typu vektor vzdálenosti.....	68
Směrovací protokoly typu vektor vzdálenosti.....	68
Objevování sítí.....	72
Údržba směrovací tabulky – periodické aktualizace.....	73
Porovnání směrovacích protokolů (typu vektor vzdálenosti):.....	78
Výpočet metriky u algoritmu typu vektor vzdálenosti.....	78
Kontrolní opakovací otázky a odpovědi (kvíz):.....	78
Kapitola 5 - Protokol RIP verze 1.....	81
RIPv1: třídní směrovací protokol typu vektor vzdálenost.....	81
Příkazy pro kapitolu 5, RIPv1.....	91
Kontrolní opakovací otázky a odpovědi (kvíz):.....	93
Kapitola 6 - VLSM a CIDR.....	95
Adresní systémy pro IPv4.....	95
Třídní a beztřídní adresace a směrování.....	97
Zvláštní typy rozhraní směrovače.....	98
Výpočet sumarizované (agregované) cesty.....	99
Kontrolní opakovací otázky a odpovědi (kvíz):.....	102
Kapitola 7 - Protokol RIP verze 2.....	104
RIP verze 2 a verze 1.....	104
Omezení protokolu RIPv1.....	105
Formát zpráv RIPv1 a RIPv2.....	105
Automatická summarizace a RIPv2.....	106
Charakteristiky RIPv2.....	107
Postup hledání chyb konfigurace.....	108
Autentizace.....	108
Příkazy pro kapitolu 7, RIPv2.....	109
Kontrolní opakovací otázky a odpovědi (kvíz):.....	112
Kapitola 8 - Směrovací tabulka – bližší pohled.....	114
Podrobnější pohled na směrování.....	114
Obsah směrovací tabulky.....	116
Postup vyhledání nejlepšího směru.....	119
Příkazy pro kapitolu 8, Směrovací tabulka – bližší pohled.....	121
Kontrolní opakovací otázky a odpovědi (kvíz):.....	121
Kapitola 9 - Protokol EIGRP.....	123
Úvod do EIGRP.....	124
Administrativní vzdáleností.....	128
Metrika.....	129
Konvergenční algoritmus DUAL.....	130
Autonomní systém.....	132
Příkazy pro kapitolu 9, EIGRP.....	132
Kontrolní opakovací otázky a odpovědi (kvíz):.....	141
Kapitola 10 - Směrovací protokoly typu stav linky (Link-State).....	143
Směrování typu stav linky.....	143
Kontrolní opakovací otázky a odpovědi (kvíz):.....	147
Kapitola 11 - Protokol OSPF.....	149
Úvod do OSPF.....	149

Cisco NetAcad: CCNA Exploration - Routing Protocols and Concepts – studijní materiál

Příkazy pro kapitolu 11, OSPF.....	158
Kontrolní opakovací otázky a odpovědi (kvíz):.....	165
Přílohy.....	167
Opakování - příklady na adresaci IPv4.....	168
Zabezpečení sítě pomocí přístupových seznamů IP.....	175
Základní informace o přístupových seznamech.....	175
Čísla ACL.....	176
Zástupné masky.....	176
Klíčová slova pro ACL.....	177
Vytvoření standardního ACL.....	177
Aplikace standardního ACL na rozhraní.....	178
Kontrola ACL.....	179
Odstranění ACL.....	179
Vytvoření rozšířeného ACL.....	179
Aplikace rozšířeného ACL na rozhraní.....	180
Klíčové slovo „established“ (nepovinné).....	180
Vytvoření pojmenovaného (named) ACL.....	181
Použití pořadového čísla řádky v pojmenovaném ACL.....	181
Odstranění řádky v pojmenovaném ACL s použitím čísla řádky.....	182
Tipy pro číslování řádek.....	182
Komentáře k řádkům v ACL.....	182
Omezení přístupu k virtuálnímu terminálu.....	183
Rychlé zopakování druhého semestru (Cram Sheet).....	184
Posloupnost zavádění OS (BOOT Sequence) pro směrovač/přepínač.....	184
Nastavení konfiguračního registru.....	184
Paměti směrovače/přepínače.....	184
Zabezpečení směrovače.....	184
Směrování.....	185
Přístupové seznamy (Access Lists).....	187
Použitá literatura.....	188
Doporučená motivační četba – bezpečnost datových sítí.....	188

CCNA Exploration – Směrování, koncepce a protokoly

Upozornění: Tento materiál nenahrazuje samotné kurikulum ani Vaše vlastní školní poznámky.

- Pro procvičování jednotlivých příkazů a celých konfigurací sítí používejte **simulátor Packet Tracer** (v poslední dostupné verzi).
- Pro analýzu síťového provozu na stanici používejte **analyzátor síťových protokolů Wireshark** v režimu s **právy lokálního administrátora na stanici**.
- Samostatně si odpovídejte na kontrolní otázky v souhrnu a kvízu pro každou kapitolu v kuriku.
- Postupujte podle pravidla: **pochopit** – **naučit se** – **procvičit** – **otestovat znalosti i dovednosti**.
- Při nastavování na reálných zařízeních v učebně i pro Packet Tracer používejte stále stejná hesla:
 - pro privilegovaný režim enable: **cisco**
 - pro linku vty - telnet a také pro linku konzole: **class**
- Protože se jedná o pracovní verzi (stále se upravuje), používejte vždy poslední dostupnou verzi dle data exportu do PDF (a zbytečně netiskněte).
- Originální materiály v angličtině: <http://www.cisco.com/web/learning/netacad/index.html> (<http://cisco.netacad.net>, <http://www.cisco.com/edu>).

Předpokládané znalosti

Kurz navazuje na *CCNA Exploration - Network Fundamentals (CCNA1 Exploration)* (Informace o e-learningové iniciativě Cisco CNAP a obsah celého kurzu CCNA viz soubor **CCNA_Exploration_1.PDF**).

Směrování, koncepce a protokoly

Základní dovednosti a kompetence absolventa kurzu *CCNA Exploration - Routing Protocols and Concepts*:

- Konfiguruje a ověřuje činnost rozhraní směrovače
- Demonstруje obsáhlé dovednosti nastavení RIPv1
- Navrhuje a implementuje beztrídní IP adresní schéma sítě
- Aplikuje základní konfigurační příkazy RIPv2 a vyhodnocuje směrovací aktualizace RIPv2 u beztrídního směrování
- Používá pokročilých konfiguračních příkazů na směrovačích s protokolem EIGRP
- Identifikuje charakteristiky směrovacích protokolů s vektorem vzdálenosti.
- Implementuje základní nastavení směrovacího protokolu OSPF.

Obsah kursu CCNA Exploration - Routing Protocols and Concepts:

- 1 Úvod do směrování a přenosu paketů na směrovači
 - 1.1 struktura směrovače (druhy a účel jednotlivých druhů pamětí)
 - 1.2 síťová rozhraní a jejich konfigurace
 - 1.3 obsah a tvorba obsahu směrovací tabulky
 - 1.4 určení nejlepší cesty
 - 1.5 funkce přepínání na směrovači
- 2 Statické směrování
 - 2.1 statická cesta
 - 2.2 summarizace
 - 2.3 implicitní cesta
 - 2.4 správa cest
 - 2.5 hledání a odstraňování chyb
- 3 Protokoly pro dynamické směrování
 - 3.1 klasifikace směrovacích protokolů
 - 3.2 metriky cest
 - 3.3 administrativní vzdálenosti protokolů
 - 3.4 směrovací protokoly a podsítě
- 4 Směrovací protokoly typu vektor vzdálenosti (Distance vektor)
 - 4.1 průzkum a propagace sítí, konvergence
 - 4.2 vytváření směrovací tabulky
 - 4.3 aktuální použití směrovacích protokolů tohoto typu
 - 4.4 prevence vzniku směrovacích smyček
- 5 Protokol RIP verze 1
 - 5.1 třídní směrovací protokol
 - 5.2 ověření a oprava chyb
 - 5.3 automatické summarizace cest
 - 5.4 propagace implicitní cesty
- 6 VLSM a CIDR
 - 6.1 IP adresace v celé třídě a beztřídní
 - 6.2 podsítě s proměnnou délkou masky
 - 6.3 automatická summarizace cest při směrování
- 7 Protokol RIP verze 2
 - 7.1 omezení protokolu RIPv1
 - 7.2 použití RIPv2 společně s VLSM nebo CIDR
- 8 Směrovací tabulka – bližší pohled
 - 8.1 podrobnější pohled na směrování
 - 8.2 struktura směrovací tabulky
 - 8.3 hledání „nejlepší“ cesty
 - 8.4 chování směrovače v závislosti na jeho různých nastaveních
- 9 Protokol EIGRP
 - 9.1 propagace směrovacích informací
 - 9.2 výpočet metriky
 - 9.3 základní konfigurace
 - 9.4 potlačení směrovacích smyček pomocí konvergenčního algoritmu DUAL (Diffusing Update Algorithm)

Cisco NetAcad: CCNA Exploration - Routing Protocols and Concepts – studijní materiál

- 10 Směrovací protokoly typu stav linky (Link-State)
 - 10.1 principy
 - 10.2 implementace protokolů tohoto typu
- 11 Protokol OSPF
 - 11.1 propagace směrovacích informací
 - 11.2 základní konfigurace
 - 11.3 výpočet metriky
 - 11.4 sítě s vícenásobnými přístupy (*multi-access network*), s více branami do sítě

Úvod

Zopakujte si úvodní kapitolu v kurikulu pro první semestr.

Příkazy pro nastavení směrovačů jsou uváděny kromě povinných (mandatorních) též jako nepovinné (volitelné), tyto nepovinné příkazy/parametry sice nejsou přímo obsahem kurikula, ale je poměrně vhodné je alespoň rámcově znát.

Pro každou kapitolu si v rámci originálního kurikula vždy zpracujte pro každý směrovací protokol následující tři aktivity *Configuration Labs* – konfigurační laboratorní cvičení - v simulátoru sítě Packet Tracer:

- *Basic Configuration* – základní konfigurace s detailním návodom,
- *Challenge Configuration* – pokročilejší konfigurace, bez detailního návodu,
- *Troubleshooting* – hledání neznámých chyb a jejich odstraňování v demonstrační konfiguraci (toto cvičení znalé studenty baví nejvíce).

Kapitola 1 – Úvod do směrování a přeposílání paketů na směrovači

V této kapitole se naučíme:

- Směrovač je počítač s operačním systémem (OS) a HW, který je speciálně navržený pro směrování.
- Demonstrovat schopnost konfigurování zařízení a nastavení adres rozhraní.
- Popsat strukturu směrovací tabulky.
- Popsat jak směrovač určuje cestu a přepíná pakety.

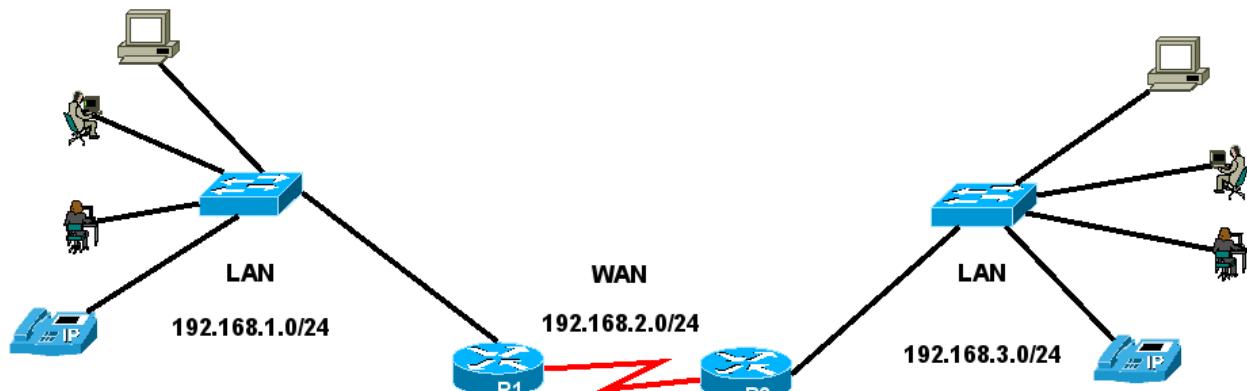
Směrovač

Směrovač (router) je centrem datové sítě. Vyjádřeno jednoduše: **směrovač propojuje jednu síť s jinou sítí**. Proto je směrovač zodpovědný za doručení dat mezi různými sítěmi včasným způsobem. Efektivita komunikace mezi vzájemně propojenými sítěmi je ve velké míře závislá na schopnosti směrovače přeposílat pakety co nejvíce efektivním způsobem. Aby vyhověly těmto požadavkům, používají se směrovače také k:

- zajištění 24x7 dostupnosti (24 hodin denně a 7 dnů v týdnu). V případě, že jedna cesta selhalá, směrovač použije jinou.
- Poskytování integrovaných služeb pro data, video a hlasové služby prostřednictvím drátových či bezdrátových sítí. Směrovače používají při kvalitě služeb (Quality of service (QoS)) nastavení priorit IP paketů tak, aby se zaručil v reálném čase provoz, který nesmí přerušen nebo zpožděn (jako jsou hlasová služba, video nebo kritická data).
- Zmírnění vlivu červů, virů a jiných útoků v síti pomocí povolení nebo zakázání přeposlání paketů.

Primární odpovědností routeru ale stále zůstává **přeposílání paketů z jedné sítě do druhé**.

Co je to směrovač ?



Struktura směrovače

Směrovač je počítač, právě tak jako jakýkoliv jiný počítač včetně PC. Úplně první směrovač, použitý pro síť ARPANET (*Advanced Research Projects Agency Network*), byl *Interface Message Processor* (IMP). IMP byl minipočítač Honeywell 316 a byl uveden do provozu 30.8.1969.

Poznámka: Sít' ARPANET byla vytvořena agenturou ministerstva obrany USA ARPA (Advanced Research Projects Agency = Agentura pro pokročilé výzkumné projekty). ARPANET byla první funkční síť na bázi přepínání paketů na světě a byla předchůdcem dnešního Internetu.

Směrovače mají mnoho stejných HW a SW komponent, jaké jsou v jiných počítačích včetně:

- CPU
- RAM
- ROM
- (síťový) operační systém (OS)
- síťové rozhraní – každé leží v jiné (pod)síti (LAN, WAN) – LAN je obvykle Ethernet, používá různá média, respektive sloty na zasunutí modulů s rozhraními a WAN technologie¹ zahrnující sériové linky, T1 připojení s protokolem PPP (*Point to Point Protocol*), Frame Relay a nebo ATM (*Asynchronous Transfer Mode*).
- konzolový port – pro počáteční konfiguraci – není to síťové rozhraní – síť nemusí být v okamžiku připojení nakonfigurována
- pomocný port (AUX) – pro vzdálené připojení modemem
- jsou obvykle bezdiskové a místo pevných disků používají paměť typu flash.

Směrovače vybírají nejlepší cestu

Primární odpovědností směrovače je směrovat (přeposlat) pakety mřížící (směrující, mající cílovou adresu) do lokální nebo vzdálené sítě pomocí:

- **určení nejlepší cesty** pro poslání paketu (na L3)²
- **posílání (přepínání) paketů** směrem k jejich cíli (na L2) – včetně zapouzdření na linkové vrstvě.

Výběr nejlepší cesty probíhá na základě obsahu směrovací tabulky. **Obsah směrovací tabulky** se vytváří:

- staticky (administrátor ručně)
- dynamicky (dynamický směrovací protokol).

CPU a paměti

Směrovač obvykle nepotřebujete otvírat, pokud zrovna nechcete upgradovat paměť.

Podobně jako PC směrovač obsahuje:

- procesor CPU (*Central Processing Unit* (CPU))
- operační paměť RAM (*Random-Access Memory*)
- paměť typu ROM (*Read-Only Memory*)

Použití:

1 Rozhraní řešené obvykle jako zásuvné výměnné moduly WIC (WAN Interface Card).

2 Směrovač propojuje sítě. Pokud ho nějaké do sítě přidáme (a změníme adresaci), zmenšuje broadcastovou doménu.

- CPU - vykonává příkazy operačního systému, jako je inicializace OS, funkce směrování a funkce přepínání.
- RAM – 128MB rozšířitelná na 384MB, při restartu směrovače ztrácí svůj obsah a ukládá následující komponenty:
 - OS se do RAM zkopiuje během zavádění systému (bootup) (je to rychlejší než pravovat přímo s pamětí Flash, jak tomu bylo ve starých routerech),
 - aktuální běžící konfigurační soubor (*running-config*),
 - směrovací tabulka,
 - ARP cache – mapování IP adres na MAC adresy,
 - vyrovnávací paměť paketů (*packet buffer*) – když je přijat na rozhraní nebo dokud není odeslán z rozhraní
- ROM – permanentní paměť, která obsahuje firmware, jež obvykle není třeba upgradovat:
 - instrukce pro zavádění systému (*bootstrap, loader*) - zavaděč,
 - základní diagnostický SW pro HW směrovače (= *POST = Power-On Self-Test*),
 - odlehčená verze IOS (*scaled-down version*) (= tzv. ROM monitor).
- Flash paměť – permanentní paměť na SIMM nebo PCMCIA kartě – 32, 64, 128 MB – implicitně 32MB, která lze elektricky vymazat a nahrát
 - obrazy (*images*) operačního systému (různě zvolená vybraná funkcionalita OS)
- NVRAM (*Nonvolatile RAM*) – 2-4MB, energeticky nezávislá permanentní paměť, po vypnutí napájení či při restartu neztrácí svůj obsah:
 - startovací konfigurace směrovače (*startup-config*) – při změnách v aktuální konfiguraci je třeba potom aktuální konfiguraci nahrát do startovací konfigurace.

Sítový operační systém IOS

IOS (*Cisco Internetwork Operating System*) – spravuje HW a SW zdroje směrovače (protože může být použit i na L3 přepínači, tak také L3 přepínače) jako alokace paměti, zabezpečení a souborový systém. IOS je víceúlohový (multitasking) OS, který integruje úlohy vztahující se ke směrování, přepínání, propojování sítí a telekomunikaci.

Ačkoliv se IOS může jevit jako stejný na mnoha směrovačích, je zde mnoho různých obrazů (*images*) IOS. Obraz systému obsahuje kompletní IOS pro určitý směrovač. Obrazy jsou závislé na modelu (typu) směrovače a funkčních obsažených v systému. Typicky, čím více funkcí, tím je větší obraz a tím větší je potřeba flash i operační paměť pro systém. Například některé funkce obsahují schopnost spustit IPv6 nebo NAT (*Network Address Translation*).

Jako jiné OS má i IOS svoje vlastní uživatelské rozhraní. Ačkoliv některé směrovače mají grafické uživatelské rozhraní (*GUI, graphical user interface*), je nejběžnějším rozhraním příkazová řádka (*CLI, command line interface*). V tomto kurikulu je použita výhradně příkazová řádka.

Během zavádění systému je startovací konfigurační soubor (*startup-config*) z NVRAM zkopirován do RAM a uložen jako běžící konfigurace (*running-config*). Jakékoli změny vložené administrátorem sítě jsou uloženy do běžící konfigurace a **bezprostředně** uvedeny v činnost v IOS.

Postup zavedení OS

Jsou čtyři hlavní fáze postupu zavedení operačního systému (*bootup process*):

1. Provedení testu POST (*Power-On Self Test*) - automatický test po zapnutí (v ROM) testuje HW směrovače – diagnostika procesoru, RAM, NVRAM.
2. Natažení zaváděcího programu (*bootstrap program, loader*) - zaváděč je natažen do operační paměti, jednotlivé instrukce provádí procesor z RAM, od této chvíle je funkční konzolové připojení a na monitoru konzole je možné vidět průběžné výpisy stavu. V této chvíli verze bootstrap.
3. Nalezení a zavedení IOS – IOS je typicky uložen v paměti flash, ale může být také uložen na TFTP serveru. Jak se začne natahovat IOS na konzoli se vypisuje znak dvojitý kříž (*hash mark*) (#) jak postupuje dekomprese systému.
4. Nalezení a zavedení souboru startovací konfigurace nebo spuštění režimu nastavování – setup. Startup-config je natažen z NVRAM a obsahuje uloženou předchozí běžící konfiguraci Obsahem jsou konfigurační příkazy a parametry jako:
 - adresy rozhraní,
 - směrovací informace,
 - hesla,
 - všechna ostatní nastavení uložené administrátorem.
 - Jestliže v NVRAM není startovací konfigurace, může jí směrovač hledat na TFTP serveru pomocí všesměrového vysílání.
 - Pokud se konfigurační soubor nalezne, jsou jeho jednotlivé příkazy vykonány.
 - Pokud se konfigurační soubor nenalezne, směrovač uživateli nabídne vstup do interaktivního nastavovacího režimu – *setup mode*. (Lze spustit přímo příkazem #setup. Ale tomu se v tomto kursu vyhneme.)

Would you like to enter the initial configuration dialog? [yes/no] : **no**

Pokud se do něho (setup mode) náhodou dostanete, ukončíte ho stiskem **Ctrl+C**.

Rozhraní příkazové řádky z konzole (CLI): Před vstupem do něj směrovač nabídne ukončení automatické instalace (to přijmeme):

Would you like to terminate autoinstall? [yes] : <**Enter**>

Press the **Enter key** to accept the default answer.

Router>

POZOR pokud na předchozí dotaz směrovače na ukončení **AutoInstall** odpovíte NO, nebo pokud má směrovač smazanou konfiguraci nebo je úplně nový, směrovač se bude pokoušet získat konfiguraci z TFTP serveru, nastavit Ethernetová rozhraní pomocí protokolu DHCP a nastavit sériová rozhraní pomocí protokolu SLARP (Serial Line Address Resolution Protocol) a to zabere několik minut.

PROTO: **Před zapnutím takovéhoto nenastaveného směrovače odpojte všechna síťová rozhraní.**

ROM	----->	Bootstrap – loader = ROM monitor - nouzový režim	
Flash paměť	----->	Cisco IOS	Naleze a zavede operační systém
TFTP server	----->		
ROM (ROM monitor, omezená verze IOS)	----->		
NVRAM	----->	Configuration File – konfigurační soubor	Naleze a zavede konfigurační soubor nebo vstup do interaktivního režimu „Setup“.
TFTP Server	----->		
Console	----->		

Ověření zavedeného systému

Příkazem **#show version** zjistíte:

1. zavedená verze IOS
2. použitý program bootstrap z ROM
3. umístění obrazu IOS (odkud byl zaveden) a jméno obrazu
4. procesor a velikost paměti RAM směrovače
5. rozhraní směrovače (názvy rozhraní na výmenných modulech ve slotech: FastEthernet0/0, Serial0/1/1, ...)
6. velikost paměti NVRAM
7. velikost paměti Flash
8. hodnotu konfiguračního registru (Nastavená hodnota konfiguračního registru různými způsoby mění chování směrovače například: odkud zavádí OS, chování během jeho zavedení, například přeskočení konfigurace, rychlosť konzolového připojení apod.
 - Implicitní tovární nastavení je 0x2102: pokusí se zavést IOS z Flash a konfigurační soubor z paměti NVRAM.
 - Hodnota 0x2142 přeskočí konfiguraci v NVRAM.
 - Má to více použití, například obnova zapomenutého hesla. Viz *Password Recovery Procedure*, kterou probereme později.

```
R3#show version
Cisco IOS Software, 1841 Software (C1841-ADVIPSERVICESK9-M), Version
12.4(15)T1, RELEASE SOFTWARE (fc2)
<vynecháno>
ROM: System Bootstrap, Version 12.3(8r)T8, RELEASE SOFTWARE (fc1)
<vynecháno>
System image file is "flash:c1841-advp�servicesk9-mz.124-15.T1.bin"
<vynecháno>
```

```
Cisco 1841 (revision 5.0) with 114688K/16384K bytes of memory.
```

```
Processor board ID FTX0947Z18E
```

```
M860 processor: part number 0, mask 49
```

```
2 FastEthernet/IEEE 802.3 interface(s)
```

```
2 Low-speed serial(sync/async) network interface(s)
```

```
191K bytes of NVRAM.
```

```
63488K bytes of ATA CompactFlash (Read/Write)
```

Configuration register is **0x2102**³

R3#

Rozhraní směrovače

Administrativní porty – fyzické konektory pro správu směrovače. Jsou dvojího druhu:

- **Konzolový (console) port** – pro (počáteční) konfiguraci. Protože to není síťové rozhraní, není třeba mít konfigurované síťová rozhraní a síťové služby. Není určený pro přeposílání paketů. Připojené PC (přes RS232 a DB9) musí mít nainstalovaný SW pro emulaci terminálu (HyperTerminal, TerraTerm).
- **Pomocný (AUX, auxiliary) port** – pro konfiguraci po připojení modemu. Také nesíťový port. V tomto kurzu nebudeme používat.

(Síťová) rozhraní směrovače (interface) – pojem rozhraní směrovače odkazuje na fyzický port na směrovači, jehož hlavní funkcí je přijímat a posílat pakety. Směrovač má více rozhraní, které jsou **připojeny do různých sítí**⁴. (Cisco IOS nepovolí na jednom směrovači, aby bylo v jedné síti více rozhraní.) Typicky se rozhraní zapojují do různých typů sítí, což znamená, že potřebují různé druhy konektorů a médií.

- Obvykle směrovač používá pro **připojení do sítí LAN** rozhraní typu FastEthernet (kabeláž UTP (podle druhu propojovaných zařízení přímý nebo překřížený) a konektory RJ-45). Používají fyzickou MAC adresu i IP adresu a protokol ARP pro jejich vzájemné spárování.
- Pro **připojení do sítí WAN** směrovač používá různé typy sériových linek jako T1, DSL,

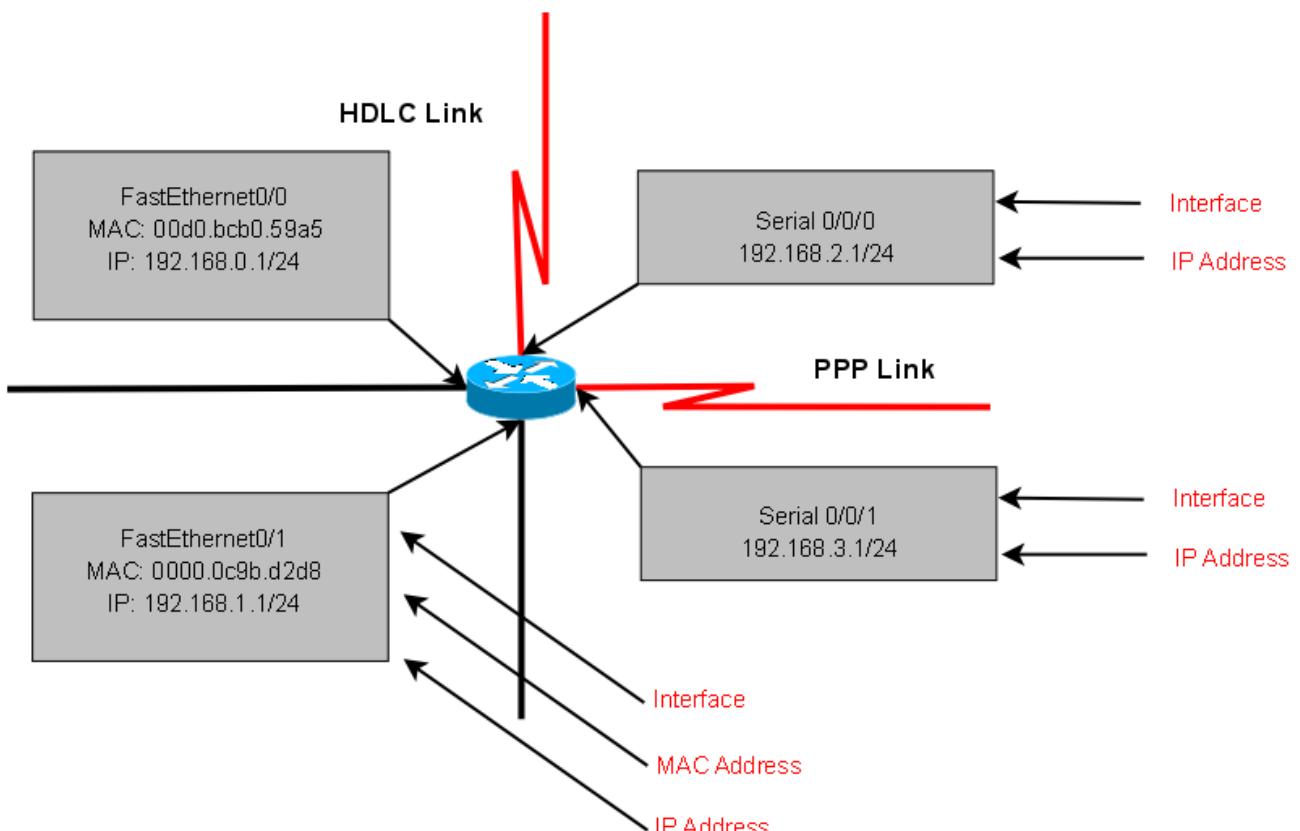
³ Nastavení hodnot konfiguračního registru:

- 0x2102 (implicitní, default): Zkontroluje v NVRAM příkazy "boot system", pokud nejsou zavede první platný IOS z paměti Flash
- 0x2100: Zavede z ROM režim ROM Monitor (ROMMON).
- 0x2101: Zavede režim ROM RxBoot. RxBoot se může připojit k serveru TFTP a stáhnout IOS do paměti Flash.
- 0x2142: Během zavádění ignoruje počáteční konfiguraci z NVRAM – používá se pro obnovu ztraceného hesla (password recovery).

⁴ Každé rozhraní na jednom směrovači je v jedné jiné síti. Poznámka: U virtuálních sítí VLAN má potom jedno rozhraní několik virtuálních podrozhraní (virtual subinterface). A každé podrozhraní je v jiné virtuální síti. (Bude probíráno v CCNA3.)

IDSN nebo technologie Frame Relay (nyní už často ale i Gigabit Ethernet). Ve WAN se MAC adresy nepoužívají (jde obvykle o dvoubodové připojení), ale některé technologie WAN všešměrovou MAC adresu mají použitou v záhlaví protokolu např. PPP a HDLC. **Síťová rozhraní směrovače mají vždy IP adresu.**

Rozhraní směrovače - logická reprezentace



Jako většina síťových zařízení, směrovače používají k indikaci stavu rozhraní elektroluminiscenční diody LED (*Light Emitting Diode*). Konkrétní význam světelného signálu je závislý na konkrétním směrovači. Například nepřerušované světlo znamená stav obsazená linka (*busy*).

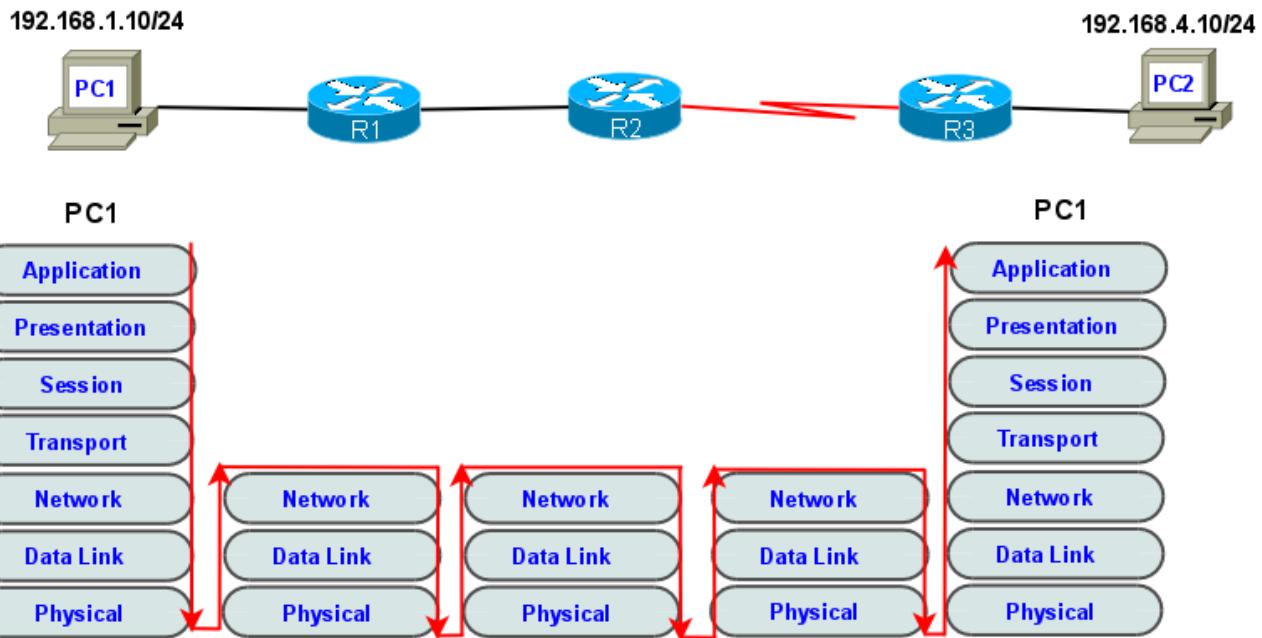
Rozhraní se mohou do slotů směrovače přidávat jako zásuvné moduly například High-Speed WAN Interface Card (HWIC) nebo WAN Interface Card (WIC) s, například, sériovými konektory Smart Serial nebo většími konektory DB60.

Směrovač na 3. vrstvě OSI modelu.

Směrovač pracuje na vrstvách 1., 2. a 3 modelu OSI.

- Na 3. vrstvě směruje pakety z jedné sítě do druhé (nejlepším směrem k cílové síti).
- Na 2. vrstvě odpouzdřuje a zapouzdřuje pakety do rámců a přepíná.
- Na 1. vrstvě zpracovává, tj. přijímá a vysílá, signály.

Směrovač pracuje na vrstvách 1, 2 a 3 modelu OSI



Implementace základního adresního schéma

Když navrhujete novou síť nebo mapujete a dokumentujete existující síť, tak minimální dokumentace by měla zahrnovat schéma topologie sítě (topologický diagram) a tabulkou IP adres s následujícími informacemi:

- jméno zařízení,
- použité rozhraní,
- IP adresa a maska podsítě,
- pro koncová zařízení jako jsou PC také adresu implicitní brány (*default gateway*).

Viz 1.2.1 obr. 1. (Oba „konce drátu“ leží vždy v jedné IP síti.)

Základní konfigurace směrovače

Základní úkoly (kroky) při konfiguraci směrovače:

- pojmenování směrovače (*hostname*)
- nastavení hesel (*password*)
- konfigurace rozhraní (*interface*)
- konfigurace denní uvítací zprávy (*banner Message of the Day, MOTD*)
- uložení změn na směrovači
- ověření konfigurace a správné funkce směrovače

Globální konfigurační režim

```
Router>enable
Router#
Router#configure terminal
Router(config) #
```

Pojmenování směrovače

```
Router(config) #hostname <jméno>
```

Nastavení hesel

```
Router(config) #enable secret <heslo>
Router(config) #line console 0
Router(config-line) #password <heslo>
Router(config-line) #login
Router(config) #line vty 0 4
Router(config-line) #password <heslo>
Router(config-line) #login
```

Uvítací zpráva

```
Router(config) #banner motd # <zpráva> #
Znak # (hash mark, dvojitý kříž) vložíte na české klávesnici pomocí dvojhmatu pravý_ALT+X.
```

Konfigurace rozhraní

```
Router(config) #interface <typ> <cíl>
Router(config-if) #ip address <ip adresa> <maska>
Router(config-if) #description <popis, a dále například číslo na helpdesk poskytovatele služby>
Router(config-if) #no shutdown
```

Každé síťové rozhraní směrovače je (musí být) v jiné síti (podsíti). => jinak vznikne chyba překrývání (overlap) sítí (IOS tuto chybu detektuje a nepovolí ji provézt, smaže nově vkládanou překrývající se adresu).

Uložení konfigurace

```
Router#copy running-config startup-config
```

Kontrola výpisů příkazu SHOW

```
Router#show running-config
Router#show startup-config
```

```
Router#show ip route
Router#show ip interface brief
Router#show interfaces
```

Obsah a tvorba obsahu směrovací tabulky

Obsah směrovací tabulky

Výpis na směrovači: *Router#show ip route*

a na hostitelské počítači PC (ve Windows:**C:\>route print** a v Linuxu **\$route**).

Obsahuje:

- u přímo připojené sítě (sousední sítě):
C 192.168.1.0/24, is directly connected, FastEthernet0/0
- u vzdálené sítě (dostupné přes alespoň jeden další směrovač):
 - kód protokolu (statická (S) nebo dynamická cesta (například R pro protokol RIP),
 - IP adresu cílové sítě/masku,
 - next hop (*gateway*) – IP adresu vstupního portu následujícího směrovače,
 - administrativní vzdálenost/metrika (např.: [120/1] pro protokol RIP).

Statické směrování (statická cesta)

Kód = S

Příklad výpisu:

S 192.168.5.0/24 [1/0] via 192.196.2.2, 00:00:20, Serial0/0/0

Použije se v následujících případech:

- Síť se skládá z pouze několika mála směrovačů. Použití dynamického směrovacího protokolu v tomto případě nemá žádný významný přínos. Naopak, dynamické směrování může přidat více režie na administrátora.
- Síť je do Internetu připojena pouze přes jednoho ISP. Není zde třeba dynamické směrování, protože ISP je jediným výstupním bodem ze sítě do Internetu.
- Velká síť konfigurovaná v topologii s jedním jediným centrálním zařízením (*hub-and-spoke topology*). Použití dynamického směrovacího protokolu je zbytečné, protože z každé větve sítě je do cíle pouze jedna cesta přes toto centrální zařízení.

Obvykle se používá kombinace statického a dynamického směrování. Než se nastavují cesty do vzdálených sítí, musí být nastaveny rozhraní a linky do přímo připojených sítí.

Dynamické směrování

Příklad výpisu:

R 192.168.4.0/24 [120/1] via 192.196.2.2, 00:00:20, Serial0/0/0

Sloupce: protokol, cílová síť/maska, brána (next-hop), stáří řádky, odchozí rozhraní (outgoing interface).

Dynamické směrovací protokoly jsou určeny pro sdílení informací o směrování mezi jednotlivými směrovači.

Základní činnosti směrovacího protokolu:

- automatické prozkoumávání sítě
- aktualizace a správa směrovacích tabulek

Směrovací protokoly pro IP

Pro IP existuje několik směrovacích protokolů. Zde je několik nejběžnějších dynamických směrovacích protokolů pro **směrování IP paketů**:

- RIP (Routing Information Protocol)
- IGRP (Interior Gateway Routing Protocol)
- EIGRP (Enhanced Interior Gateway Routing Protocol)
- OSPF (Open Shortest Path First)
- IS-IS (Intermediate System-to-Intermediate System)
- BGP (Border Gateway Protocol)

Principy směrovací tabulky.

Občas se v tomto kurzu odkazujeme na **tři principy vztahující se ke směrovací tabulce**, které vám pomohou porozumět, nastavit a odstranit chyby směrování. (Tyto principy jsou převzaty z knihy *Alex Zinin: Cisco IP Routing.*)

1. **Každý směrovač činí svá rozhodnutí samostatně a založené pouze na svojí vlastní směrovací tabulce.**
2. **Skutečnost, že jeden směrovač má ve své směrovací tabulce určité informace, neznamená, že ostatní směrovače mají tytéž informace.**
3. **Směrovací informace o cestě z jedné sítě do druhé neposkytuje směrovací informace o opačné neboli zpětné cestě.** (Při asymetrickém směrování může být navíc zpětná cesta jiná.)

Jaké jsou **důsledky těchto principů?**

Představte si, že máme za sebou propojené tři směrovače R1, R2 a R3. K R1 a R3 jsou připojeni klienti PC1 a PC2.

1. Po směrovacím rozhodnutí, směrovač R1 pošle paket adresovaný do PC2 na směrovač R2. R1 zná pouze informace ze své směrovací tabulky, které říkají, že R2 je další skok na cestě. R1 neví jestli R2 skutečně má či nemá cestu do cílové sítě.
2. Je v zodpovědnosti administrátora sítě zajistit, aby každý směrovač, který spravuje, měl úplné a správné směrovací informace, pomocí kterých mohou být pakety poslány mezi libovolnými dvěma sítěmi. To lze zajistit statickými cestami, dynamickými směrovacími protokoly nebo kombinací obojího.
3. Směrovač R2 je schopen odeslat paket směrem k cílové síti PC2. Přesto byl paket z PC2 do

PC1 zahoven směrovačem R2. Ačkoliv má R2 ve své směrovací tabulce informace o cílové síti paketu z PC1, nevíme zda má informace pro zpětnou cestu do sítě s PC1.

Asymetrické směrování

Protože směrovač nemusí mít nutně ve svých směrovacích tabulkách ty samé informace, pakety mohou cestovat síť v jednom směru jednou cestou a zpátečním směrem jinou cestou. To se nazývá asymetrické směrování. Asymetrické směrování je běžnější v Internetu, který používá směrovací protokol BGP, než v interních sítích.

V tomto případě, když síť navrhuje a odstraňuje chyby, měl by administrátor ověřit následující směrovací informace:

Je cesta od zdroje k cíli dostupná v obou směrech?

Je cesta braná v obou směrech ta samá cesta? (Asymetrické směrování není sice úplně neběžné, ale někdy může přinášet vznik dalších problémů.)

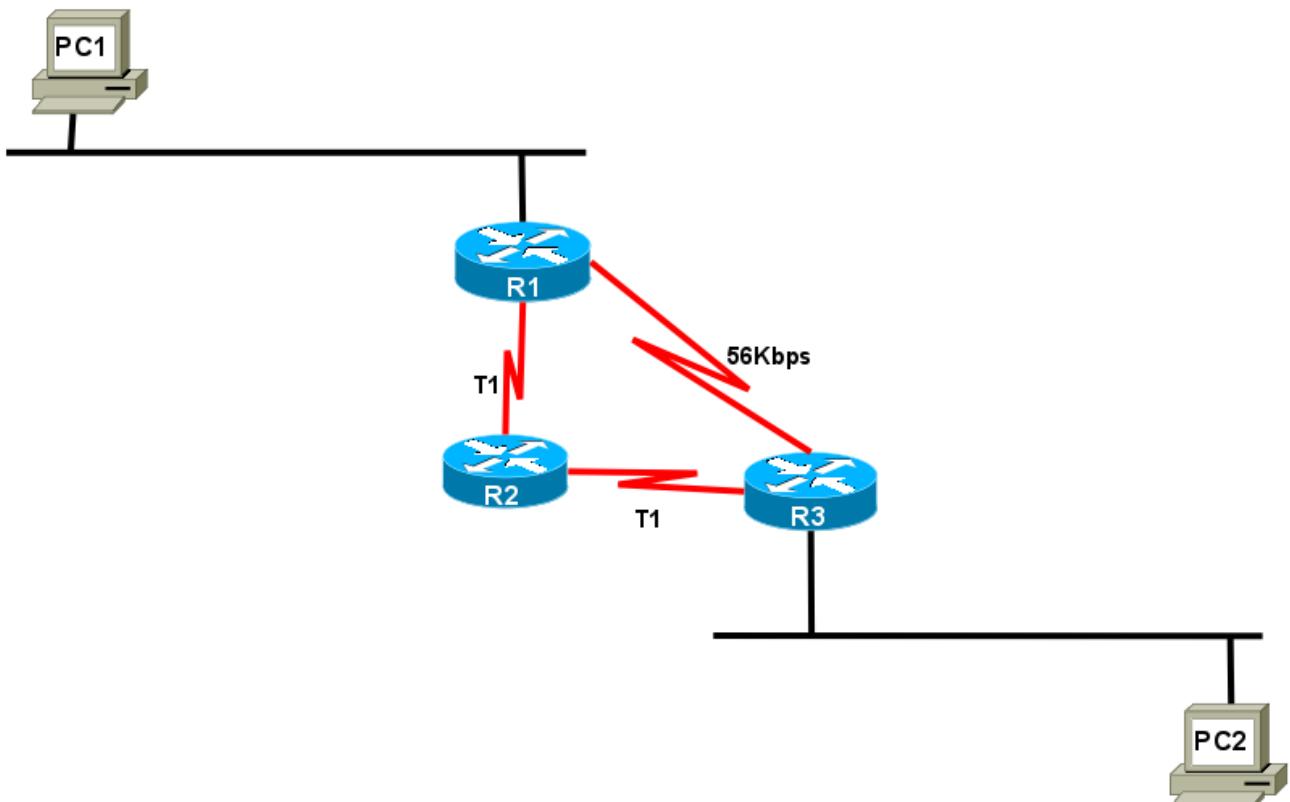
Určení cesty a přeposlání

Zopakujte si nejprve strukturu IP paketu a rámce Ethernet. Význam a formát nejdůležitějších polí příslušné PDU.

- IPv4: Verze, životnost - TTL, zdrojová IP adresa, cílová IP adresa
- Ethernet: zdrojová MAC adresa, cílová MAC adresa, FCS

Nejlepší cesta a metrika

Metrika - přenosová kapacita versus počet přeskoků



Směrování je nalezení v nějakém smyslu nejlepší cesty. Nejlepší cesta se potom vybírá podle nejmenší hodnoty Administrativní vzdálenosti (*Administrative Distance, AD*) a při stejné AD nejmenší metriky. Administrativní vzdálenost je číselné vyjádření kvality („ceny“, nákladů, *cost*) či důvěryhodnosti (*trustfulness*) směrovacího protokolu, kterým byla vytvořena příslušná řádku ve směrovací tabulce⁵. Metrika (*metric*) je potom pro jeden konkrétní směrovací protokol vyjádření kvality (=ceny) linky (směru, cesty). Nejkvalitnější je ta cesta s číselně nejmenší metrikou. Metrika může být určena na základě různých veličin (například počet přeskoků, přenosová rychlosť linky, zátěž, cena apod.). Přeposlání paketu

Přeposlání paketu zahrnuje dvě funkce:

- určení cesty (*routing*),
- přepnutí (*switching*).

Určení cesty je proces, jak směrovač určí kterou cestu má použít při přeposlání paketu. Aby ji našel, směrovač hledá ve své směrovací tabulce adresu sítě, která lze spárovat s cílovou IP adresou paketu.

To může mít jeden ze tří následujících výsledků:

- **Přilehlá síť** – pokud cílová IP adresa paketu patří zařízení, které je v přilehlé síti na jednom z rozhraní směrovače, je paket přímo předán tomuto zařízení.

⁵ Na jednom směrovači může najednou běžet více různých směrovacích protokolů. Směrovací protokol může také pomocí tzv. Redistribuce přebírat směry z jiných směrovacích protokolů.

- **Vzdálená síť** - pokud cílová IP adresa paketu patří do vzdálené sítě, je paket předán na následující směrovač. Vzdálenou síť lze dosáhnout pouze přes jiný směrovač.
- **Není určena žádná cesta** - pokud cílová IP adresa paketu nepatří ani do přilehlé ani do vzdálené sítě a směrovač nemá implicitní cestu (*default route*) je paket zahozen. Směrovač potom pošle ICMP zprávu o nedosažitelnosti cíle na zdrojovou IP adresu paketu.

V prvních dvou případech směrovač IP paket znova zapouzdří do rámce spojové vrstvy podle typu zjištěného odchozího rozhraní (*outgoing interface*) pro určenou trasu. (Například pro rozhraní FastEthernet je to rámc ve formátu Ethernet nebo pro sériovou linku nakonfigurovanou pro protokol PPP je to rámc ve formátu PPP.)

Funkce přepnutí je proces přijetí paketu na jednom rozhraní směrovače a jeho přeposlání z jiného rozhraní tohoto směrovače. Klíčové je zapouzdření (*encapsulation*) paketů do odpovídajícího typu rámce spojové vrstvy pro odchozí linku přidáním záhlaví a zápatí L2.

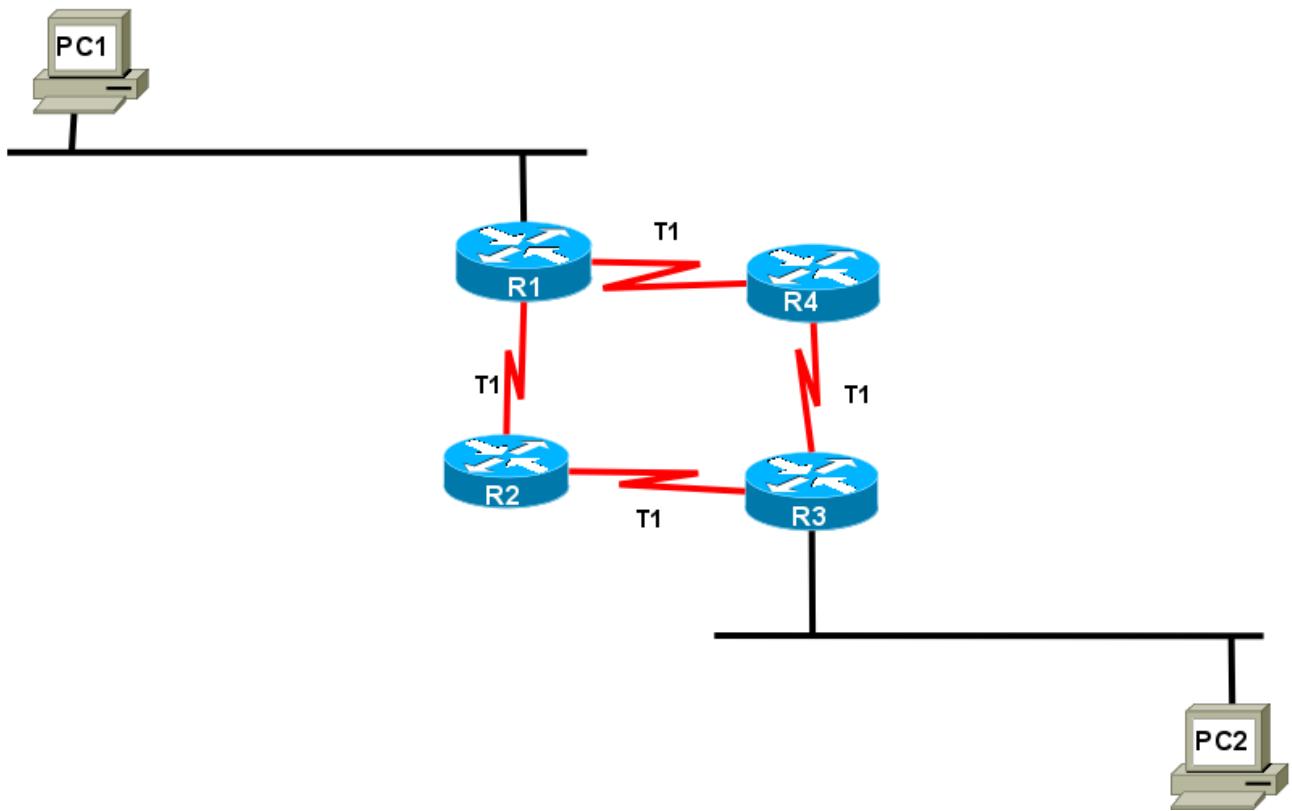
Zjednodušený postup zpracování dat na směrovači, když přijme paket z jedné sítě, který je adresovaný a přeposlaný do jiné sítě:

1. Odpouzdří L3 paket odstraněním záhlaví a zápatí přijatého L2 rámce.
2. Zjistí cílovou IP adresu paketu, aby ve směrovací tabulce našel nejlepší cestu do cílové sítě.
3. Zapouzdří L3 paket do nového L2 rámce a odešle tento rámc ven z odchozího rozhraní.

Vyvažování zátěže u cest se stejnou cenou

Pokud je více linek se stejnou cenou (administrativní vzdáleností a metrikou), byla by vždy používána pouze první linka (směr) ve směrovací tabulce. To není vždy vhodné, je zatížena jedna linka a ostatní nejsou zatížené vůbec, proto je možné na směrovači zapnout vyvažování zátěže cest se stejnou cenou (*equal cost load balancing*). Jednotlivé směry jsou potom cyklicky přepínány (*round robin approach*).

Vyrovnaná zátěž při stejné ceně trasy (Equal Cost Load Balancing)

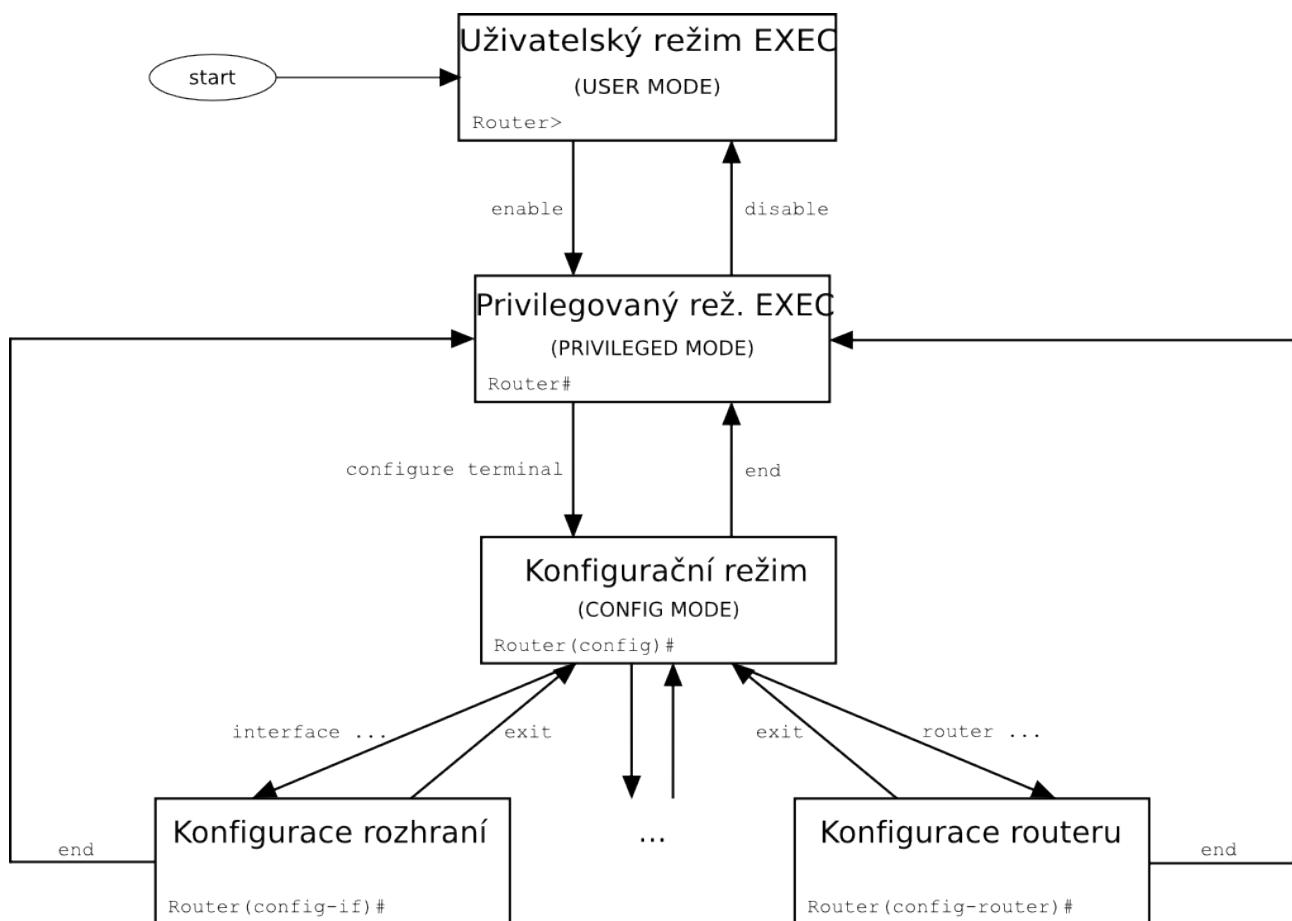


(Směrovací protokol EIGRP podporuje i vyvažování zátěže pro cesty s různými cenami.)

Průvodce základní konfigurací (nastavením) směrovače

V této podkapitole jsou uvedeny informace a příkazy týkající se následujících oblastí:

- Konfigurace routeru a to zvláště:
 - názvy (*Names*)
 - hesla (*Passwords*)
 - rozhraní (*Interfaces*)
 - uvítací zprávy (*MOTD banners*)
 - tabulky IP hostitelů (*IP host tables*)
 - ukládání a rušení vašich nastavení (*Saving and erasing your configurations*)
- příkazy show, ke kontrole nastavení (konfigurací) směrovače.



(Zdroj obrázku: modifikace http://CS.Wikipedia.org/wiki/Cisco_IOS)

Režimy směrovače

Router>	Uživatelský (User EXEC)
Router#	Privilegovaný (Enable EXEC)
Router(config)#	Globální konfigurační
Router(config-if)#	Konfigurace rozhraní
Router(config-subif)#	Konfigurace pod-rozhraní (Subinterface)
Router(config-line)#	Konfigurace linky (např. virtuální linky Telnetu = vty, console)
Router(config-router) #	Konfigurace (nastavení) směrovacího protokolu

TIP: Existují ještě další režimy (módy) než zde uvedené.

POZOR:

Ne všechny příkazy pracují ve všech režimech. Proto si dávejte pozor. Jestliže napišete příkaz (například `show run`) a vrátí se chyba, ujistěte se, že jste ve správném režimu.

I když umíte používat návod „?“, musíte vědět v jakém režimu lze ten který příkaz použít. Naopak, pokud jste si jistí syntaxí příkazu, můžete použít příkaz **do (=proved)** (viz kapitola Příkaz DO).

Globální konfigurační mód

Router>	Vidíte nastavení, ale nelze měnit
Router#	Vidíte nastavení a můžete se přesunout do režimu, kde lze změnit.
Router#config t Router(config)#	Přechod do globálního konfiguračního režimu Tento prompt indikuje, že můžete začít dělat změny.

Nastavení názvu směrovače

Tento příkaz pracuje jak na směrovačích tak i na přepínačích.

Router(config)#hostname Cisco Cisco(config)#	Název může být jakékoliv slovo si zvolíte. Změní prompt (systémovou výzvu).
--	---

Nastavení hesel

Pracuje jak na směrovačích tak i přepínačích.

Router(config)#enable password cisco	Nastaví nešifrované heslo příkazu enable (na slovo cisco). NEPOUŽÍVEJTE
Router(config)#enable secret class	Nastaví šifrované heslo příkazu enable (na slovo class).
Router(config)#line con 0 Router(config-line)#password console Router(config-line)#login	Vstup do režimu konfigurace konzole Nastaví heslo konzolové linky na console Vynutí ověření hesla při přihlášení (login).
Router(config)#line vty 0 4 Router(config-line)#password telnet Router(config-line)#login	Vstup do režimu linky virtuálního terminálu (vty) – Telnet pro všech 5 linek vty Nastaví heslo vty na slovo telnet. Vynutí ověření hesla při přihlášení (login).
Router(config)#line aux 0 Router(config-line)#password backdoor Router(config-line)#login	Vstup do režimu konfigurace pomocné linky (auxiliary line mode) Nastaví heslo pomocné linky na slovo backdoor . Vynutí ověření hesla při přihlášení (login).

Varování: Heslo *Enable secret* je implicitně šifrované a *Enable password* není. Z tohoto důvodu je doporučena praxe nikdy nepoužívat heslo *enable password*. Při konfiguraci směrovače používejte pouze heslo *enable secret*. (Pokud jsou z historických důvodů nastavena obě, musí být z bezpečnostních důvodů různá a *enable secret* má přednost.)

Varování: Nelze nastavit stejné heslo v obou příkazech *enable secret* a *enable password*. Pokud by to šlo, anulovalo by to šifrování.

Šifrování hesel

Router(config)#service password-encryption	Na všechna hesla je aplikováno slabé šifrování.
--	---

Router(config)#enable password <i>cisco</i>	Nastavuje heslo <i>enable password</i> na slovo cisco
Router(config)#line con 0	...
Router(config-line)#password <i>cisco</i>	Pokračuje nastavování hesel stejně jako nahoře.
	...
Router(config)#no service password-encryption	Vypne šifrování hesel.

Varování: Jestliže zapnete službu šifrování hesel, použije se, a jestliže ji potom opět vypnete, všechna zašifrovaná hesla zůstanou zašifrovaná. Nová hesla budou již ovšem nezašifrovaná.

Příkazy show

Router#show ?	Vypíše všechny dostupné příkazy show.
Router#show interfaces	Zobrazí statistiky pro všechna rozhraní.
Router#show interface serial 0	Zobrazí statistiky pro určité rozhraní, v tomto případě Serial 0.
Router#show ip interface brief	Zobrazí přehled všech rozhraní, včetně stavu (status) a přiřazené IP adresy.
Router#show controllers serial 0	Zobrazí statistiky pro HW rozhraní. Statistiky zobrazují zda jsou nastaveny hodiny (<i>clock rate</i>) a zda je kabel DCE, DTE, nebo není připojen.
Router#show clock	Zobrazí čas nastavený na zařízení.
Router#show hosts	Zobrazí lokální lokální „kešovanou“ tabulku hosts (lokální hostitel-IP adresa). Zde jsou názvy a IP adresy hostitelů v síti, ke kterým se můžete připojit.
Router#show users	Zobrazí všechny uživatele připojené k zařízení.
Router#show history	Zobrazí historii použitých příkazů.
Router#show flash	Zobrazí informace o paměti Flash.
Router#show version	Zobrazí informace o zavedené verzi SW.
Router#show arp	Zobrazí tabulku ARP.
Router#show protocols	Zobrazí stav nastavených protokolů L3.
Router#show startup-config	Zobrazí konfiguraci uloženou v paměti NVRAM.
Router#show running-config	Zobrazí konfiguraci aktuálně běžící v RAM.
Router#show processes	Zobrazí využití CPU a běžící procesy na směrovači.

Názvy rozhraní

Jedním z největších problémů, kterým čelí noví administrátoři jsou názvy rozhraní na různých modelech směrovačů. V následující tabulce jsou vypsány názvy rozhraní pro Ethernet, Fast Ethernet, a Sériových rozhraní na směrovačích řady 2500, 1700 a 2600.

Pevná rozhraní (řada 2500)	Modulární (výměnná) rozhraní (řada 1700)	Modulární (výměnná) rozhraní (řada 2600)
Router(config)#interface type port	Router(config)#interface type port	Router(config)#interface type slot/port
Router(config)#int serial 0 (s0)	Router(config)#int interface serial 0	Router(config)#interface serial 0/0 (s0/0)
Router(config)#int ethernet 0 (e0)	Router(config)#intf fastethernet 0 ace	Router(config)#int fastethernet 0/0 (fa0/0)

Přechod mezi rozhraními

To co se děje ve sloupci 1 je totéž co ve sloupci 2.

Router(config)#int s0	Router(config)#int s0	Přechod do režimu konfigurace rozhraní S0.
Router(config-if)#exit	Router(config-if)#int e0	Z int S0, přechod do E0.
Router(config)#int e0	Router(config-if) #	Nyní v režimu E0
Router(config-if) #		Prompt se nezměnil. Dávejte si pozor.

Konfigurace sériového rozhraní

Router(config)#int s0/0	Přechod do režimu konfigurace rozhraní Serial 0/0.
Router(config-if)#description Link to ISP	Volitelný popis linky může být důležitý – něco významného.
Router(config-if)#ip address 192.168.10.1 255.255.255.0	Přiřadí adresu a masku podsítě k rozhraní.
Router(config-if)#clock rate 56000	Přiřadí takt hodin (<i>clock rate</i>) pro rozhraní DCE . (<u>v Kb</u>)
Router(config-if)#no shutdown	Zapne rozhraní. (Implicitně je rozhraní směrovače vždy administrativně vypnuté (administratively down)!)

TIP: Příkaz nastavení taktu hodin (*clock rate*) se použije jen na sériovém rozhraní, do kterého je zastrčen DCE kabel (na druhé straně kabelu resp. v jeho polovině, pokud jsou spojené, je konektor **V.35 Female**). Takt hodin musí být nastaven na každé sériové lince mezi směrovači. Nezáleží na tom, do kterého směrovače je zastrčen DCE kabel, nebo do kterého rozhraní je kabel zastrčen. Serial 0 na jednom směrovači může být zastrčen do Serial 1 na druhém směrovači.

Konfigurace rozhraní Ethernet/FastEthernet

Router(config)#int fa0/0	Přechod do režimu konfigurace rozhraní Fast Ethernet 0/0.
Router(config-if)#description Accounting LAN	Volitelný popis linky může být důležitý – něco lokálně významného.
Router(config-if)#ip address 192.168.20.1 255.255.255.0	Přiřadí adresu a masku podsítě k rozhraní.
Router(config-if)#no shut	Administrativně zapne rozhraní.

Vytvoření uvítacího hlášení (MOTD Banner)

Router(config)#banner motd # This is a secure system. Authorized Personnel Only! #	# (dvojitý kříž, <i>hash mark</i>) je známý jako oddělovací znak. Oddělovací znak musí obklopovat text uvítacího hlášení z obou stran a nesmí být použit uvnitř samotného textu hlášení. Znak # vložíte pomocí pravý_ALT+X.
--	---

Nastavení časového pásma (Clock Time Zone)

Router(config)#clock timezone EST -5	Nastaví časovou zónu pro zobrazení. Založeno na světovém času UTC (coordinated universal time). (EST - Eastern Standard Time (na východě USA) je 5 hodin za UTC, SEČ je UTC + 1 hodina = 1 hodinu před.)
--------------------------------------	--

Přiřazení lokálního jména hostitele k IP adresě

Router(config)#ip host london 172.16.1.3	Přiřadí jméno/název hostitele k IP adrese. Po tomto přiřazení můžete použít jméno místo IP adresy v příkazu <i>telnet</i> nebo <i>ping</i> .
Router#ping london =	

TIP: Implicitní číslo portu v příkazu *ip host* je 23, nebo Telnet. Když se chcete připojit Telnetem k nějakému zařízení, pouze vložte samotné jméno zařízení:

Router#london = Router#telnet london = Router#telnet 172.16.1.3

Příkaz *no ip domain-lookup* (vypnutí překladu jména domény na IP adresu)

Router(config)#no ip domain-lookup	Vypíná automatický vyhodnocení neznámého příkazu jako lokální jméno hostitele.
------------------------------------	--

TIP: Musíte čekat minutu nebo dvě, vždy když napíšete příkaz nesprávně a když směrovač zkouší přeložit váš příkaz na doménový server 255.255.255.255? Směrovač je implicitně nastaven na pokus vyhodnotit jakékoli slovo, které není příkaz, na DNS serveru s adresou 255.255.255.255. Jestliže se nechystáte nastavit DNS server, vypněte tuto vlastnost, abyste šetřili čas - zvlášť pokud

jste špatný písář.

Příkaz *logging synchronous*

Router(config)#line con 0 Router(config-line)#logging synchronous	Zapne synchronní logování (synchronous logging). Informace odesílané na konzoli nebudou přerušeny příkazem, který píšete na klávesnici. Příkaz se přesune na novou řádku.
--	---

TIP: Objeví se příkaz, který píšete, vždy uprostřed řádky? Ztratíte pozici v řádku? Nevíte, kde jste v příkazu, a tak stisknete ENTER a začnete vše od začátku? Příkaz *logging synchronous* řekne směrovači, že jestliže se na display zobrazují nějaké informace, měl by se prompt a nový příkaz přesunout na novou řádku pokud se snažíte něco psát na klávesnici, aby vás to nemálo. Jestliže jste pokračovali v psaní, měl by se příkaz tak jako tak vykonat správně, i když na obrazovce vypadá zobrazený špatně.

Příkaz *exec-timeout*

Router(config)#line con 0 Router(config-line)#exec-timeout 0 0 Router(config-line)#+	Nastavuje časový limit pro automatické odhlášení konzole. Nastavení na 0 0 (minuty sekundy) znamená, že konzole nebude nikdy automaticky odhlášena.
--	---

TIP: Příkaz *exec-timeout 0 0* je skvělý v laboratoři, protože se konzole nikdy neodpojí. Ale v reálném světě je to velmi nebezpečné (mizerné zabezpečení).

Uložení konfigurace

Router#copy run start	Uloží aktuální běžící konfiguraci (running-config) do lokální paměti NVRAM.
Router#copy run tftp	Uloží aktuální běžící konfiguraci na vzdálený TFTP server.

Smazání počáteční konfigurace

Router#erase start	Smaže soubor <i>startup-config</i> z paměti NVRAM.
Router#reload	Znovu zavedení operačního systému po smazání počáteční konfigurace. => prázdná aktuální běžící konfigurace

TIP: Running-config je ale i po smazání startup-config z NVRAM stále v dynamické operační paměti RAM. Znovu natažení (*Reload*) OS směrovače smaže aktuální konfiguraci running-config.

Smazání předchozí konfigurace (erase start) proved'te, pokud Vám nebude řečeno jinak, na začátku a na konci každého praktického cvičení se směrovači v laboratoři datových sítí. Je to slušnost vůči ostatním studentům.

Příkaz DO

Router(config)# do show running-config	Provede (do , \' dü \') příkaz z privilegovaného režimu enable (zde například show running-config) přímo v globálním konfiguračním režimu.
Router(config)#	Po tom, co byl příkaz vykonán, směrovač stále zůstává v globálním konfiguračním režimu.

TIP: Příkaz **do** je užitečný, když potřebujete provést příkazy z privilegovaného režimu EXEC (jako **show**, **clear** nebo **debug**) aniž opustíte globální konfigurační režim nebo libovolný jeho konfigurační podrežim.

POZOR:

- Příkaz **do** nemůžete použít pro provedení příkazu **configure terminal**, protože ten změní daný režim na globální konfigurační režim.
- Při použití příkazu **do** také navíc přijdete o nápovědu pro, za příkazem **do**, vkládaný příkaz.

Příklad konfigurace: základní nastavení směrovače

Síťová topologie pro příklad: Zleva doprava. **PC1**, překřížený kabel UTP, rozhraní FastEthernet 0/0 **směrovače Plzeň** – rozhraní Serial 0/0 (172.16.20.1, maska 255.255.255.0), **strana DCE** sériového kabelu - sériová linka, rozhraní Serial 0/0 (172.16.20.2, maska 255.255.255.0) - **směrovač Praha**, přímý UTP kabel, **switch**, přímý UTP kabel, **PC2**. Jednotlivé IP sítě **zleva doprava**: 172.16.10.0/24, 172.16.20.0/24 a 172.16.30.0/24.

1. Nakreslete si nejprve detailní schéma zapojení (fyzickou topologii sítě) s IP adresami a názvy rozhraní. Zapište si též i adresy a masky jednotlivých sítí.
2. Případně i vyplňte tabulkou adres rozhraní.
3. Ověřte návrh (výpočet) adresace (že to, co má být v jedné síti je skutečně v jedné síti a co má být v různých sítích je v různých sítích).

Nastavení pro směrovač Plzen (jsou zadávané zkrácené příkazy)

Prompt (výzva uživatele) a příkaz	Popis
Router>en	Vstup do privilegovaného režimu.
Router#clock set 18:30:00 15 Mar 2008	Nastaví lokální čas na směrovači.
Router#config t	Vstup do globálního konfiguračního režimu.
Router(config)#hostname Plzen	Nastaví jméno směrovač na Plzen.
Plzen(config)#no ip domain-lookup	Vypne překlad jmen pro neznámé příkazy (chyby pravopisu).
Plzen(config)#banner motd # This is the Plzen Router. Authorized Access Only #	Vytvoří uvítací hlášení (MOTD banner).

Cisco NetAcad: CCNA Exploration - Routing Protocols and Concepts – studijní materiál

Plzen(config)#clock timezone SEC 1	Nastaví časovou zónu na SEČ (+1 od UTC)
Plzen(config)#enable secret cisco	Nastaví šifrované heslo privilegovaného režimu <i>enable secret</i> na slovo <i>cisco</i> .
Plzen(config)#service password-encryption	Hesla budou šifrována slabou šifrou.
Plzen(config)#line con 0	Vstup do režimu nastavení konzole.
Plzen(config-line)#logging synchronous	Příkazy nebudou přerušeny nevyžádanými hlášeními nebo zprávami.
Plzen(config-line)#password class	Nastaví heslo na slovo <i>class</i> .
Plzen(config-line)#login	Umožní kontrolu hesla při přihlášení.
Plzen(config-line)#line vty 0 4	Přesun do režimu konfigurace virtuálních linek Telnet od 0 do 4 (implicitně je jich 5).
Plzen(config-line)#password class	Nastaví heslo na slovo <i>class</i> .
Plzen(config-line)#login	Umožní kontrolu hesla při přihlášení.
Plzen(config-line)#line aux 0	Přesun do režimu konfigurace pomocné linky (auxiliary line mode).
Plzen(config-line)#password class	Nastaví heslo na slovo <i>class</i> .
Plzen(config-line)#login	Umožní kontrolu hesla při přihlášení.
Plzen(config-line)#exit	Přesun zpět do režimu globální konfigurace.
Plzen(config)#no service password-encryption	Vypne šifrování hesel.
Plzen(config)#int fa 0/0	Přesun do režimu konfigurace Fast Ethernet 0/0.
Plzen(config-if)#desc Engineering LAN	Nastaví lokálně významný popis rozhraní.
Plzen(config-if)#ip address 172.16.10.1 255.255.255.0	Přiřadí IP adresu a masku podsítě k rozhraní.
Plzen(config-if)#no shut	Administrativně zapne rozhraní.
Plzen(config-if)#int s0/0	Přesun přímo do režimu konfigurace Serial 0/0.
Plzen(config-if)#desc Linka na smerovac Praha	Nastaví lokálně významný popis rozhraní.
Plzen(config-if)#ip address 172.16.20.1 255.255.255.0	Přiřadí IP adresu a masku podsítě k rozhraní.
Plzen(config-if)#clock rate 56000	Nastaví takt hodin pro synchronní sériový přenos. (V tomto rozhraní musí být zastrčen kabel DCE .)
Plzen(config-if)#no shut	Administrativně zapne rozhraní.
Plzen(config-if)#exit	Přesun zpět do režimu globální konfigurace.
Plzen(config)#ip host Praha 172.16.20.2	Nastaví překlad lokálního jména hostitele Praha na IP adresu 172.16.20.2.

Plzen(config)#exit	Přesun zpět do privilegovанého režimu.
Plzen#copy run start	Uložení aktuální běžící konfigurace do paměti NVRAM
Plzen#reload	Znovu natažení operačního systému a konfigurace. Pokud nebyla konfigurace uložena, je nenávratně ztracena.

Zdroj: CCNA2 Companion Guide, Cisco Press, kapitola 3 + úpravy

Obnova zapomenutého hesla pro směrovače Cisco

(*Password Recovery Procedure⁶*)

Krok	Příkazy pro řadu 2500	Příkazy pro řady 1700/2600/ISR
Krok 1: Při zavádění OS přerušte zavádění.	Stiskněte CTRL+BREAK >	Stiskněte CTRL+BREAK ⁷ rommon 1>
Krok 2: Změna konfiguračního registru, aby byl ignorován obsah konfigurace v paměti NVRAM.	>o/r 0x2142 >	rommon 1>confreg 0x2142 rommon 2>
Krok 3: Znovuzavedení OS.	>i	rommon 2>reset
Krok 4: Vstup do privilegového režimu. (Nevstupujte do interaktivního konfiguračního režimu setup.)	Router>enable Router#	Router>enable Router#
Krok 5: Kopie startovací konfigurace do běžící konfigurace.	Router#copy startup-config running-config ...<vynechaný výstup>... Denver#	Router#copy startup-config running-config ...<vynechaný výstup>... Denver#
Krok 6: Změna hesla.	Denver#configure terminal Denver(config)#enable secret new Denver(config)#	Denver#configure terminal Denver(config)#enable secret new Denver(config)#
Krok 7: Znovunastavení konfiguračního registru do na jeho implicitní hodnotu.	Denver(config)#config-register 0x2102	Denver(config)#config-register 0x2102
Krok 8: Uložení konfigurace.	Denver(config)#exit Denver#copy running-config startup-config Denver#	Denver(config)#exit Denver#copy running-config startup-config Denver#

⁶ Postup pro konkrétní síťové zařízení najeznete v jeho dokumentaci pod tímto anglickým označením.

⁷ Ukončovací sekvence závisí na použitém operačním systému, ve které běží emulátor terminálu a na nastavení samotného emulátoru (typicky pro Windows Ctrl+Break nebo Ctrl+C.)

IOS Escape Sequence

Kdykoliv potřebujete ukončit běh určitého příkazu v operačním systému IOS, použijte jako **ukončovací sekvenci** (*escape sequence*) trojhmat **Shift+Ctrl+6**.

Popřípadě **uspat relaci Telnet pomocí Shift+Ctrl+6 X**. A opět obnovit stiskem ENTER na prázdné řádce.

Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Kroky při startu směrovače:
 - a) provedení testu HW POST z ROM,
 - b) spuštění zavaděče systému (bootstrap) z ROM,
 - c) nalezení a zavedení operačního systému z paměti flash,
 - d) zavedení konfiguračního souboru z NVRAM.
- 2) Dva příkazy, které by technik mohl použít pro zjištění IP adres na sériovém rozhraní směrovače:
 - a) show interfaces
 - b) show ip interface brief
- 3) Pravdivé tvrzení o směrování:
 - a) Každý směrovač se rozhoduje sám a pouze na základě svých směrovacích tabulek.
- 4) Dvě základní činnosti směrovacího protokolu:
 - a) objevuje nové sítě,
 - b) aktualizuje a udržuje obsah směrovací tabulky.
- 5) Administrátor konfiguruje nový směrovač. Jsou nastaveny IP adresy a masky ale nikoliv směrovací protokol nebo statické cesty. Které směry jsou v této chvíli ve směrovací tabulce?
 - a) Přímo připojené sítě.
- 6) Jak směrovač přeposílá pakety?
 - a) Pokud je cílová IP adresa v přímo připojené síti, směrovač odešle paket z odchozího rozhraní uvedeného ve směrovací tabulce pro cílovou síť.
 - b) Pokud je cílová IP adresa ve vzdálené síti, směrovač odešle paket na další skok (next hop) uvedený ve směrovací tabulce pro cílovou síť.
- 7) Definice metriky:
 - a) Metrika je kvantitativní hodnota, kterou směrovací protokol hodnotí cenu, náklady pro určitou cestu/směr.
- 8) Administrátor nastavil na směrovači příkaz „ip route 0.0.0.0 0.0.0.0 serial0/0“. Jak se tento příkaz projeví ve směrovací tabulce za předpokladu, že je rozhraní serial0/0 zapnuté?
 - a) S* 0.0.0.0/0 is directly connected, Serial0/0

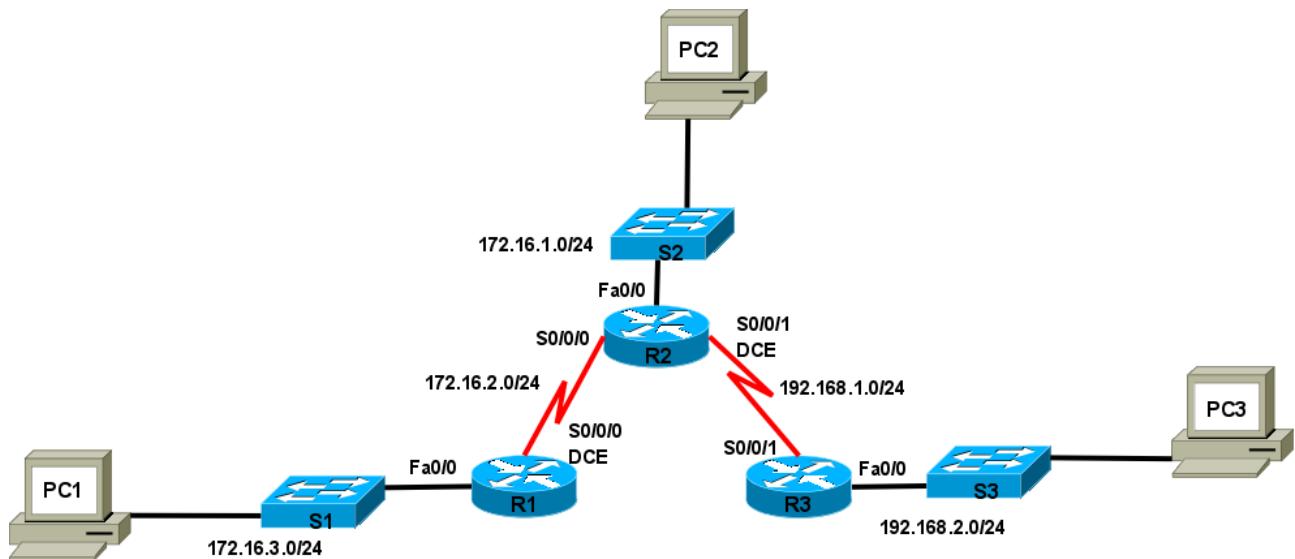
Kapitola 2 – Statické směrování

V této kapitole se naučíme:

- Definovat všeobecnou roli, kterou směrovač hraje v síti
- Popsat přímo připojené sítě a různá rozhraní směrovače
- Prozkoumat přímo připojené sítě ve směrovací tabulce a použít protokol CDP
- Popsat statické cesty s jejich odchozími rozhraními
- Popsat sumarizovanou a implicitní cestu
- Prozkoumat jak jsou pakety přeposílány když je použita statická cesta
- Spravovat statické cesty a odstraňovat jejich závady

Role směrovače v síti

Propojovací zařízení mezi jednotlivými sítěmi, které směruje a přepíná pakety a dále slouží k segmentaci (rozdelení a zmenšení) domén všesměrového vysílání.



Topologie a tabulka adres

Před zahájením práce je nanejvýš vhodné si připravit:

1. Topologické schéma sítě (s názvy, IP adresami a maskami rozhraní i jednotlivých sítí)
2. Tabulka adres (například):

Zařízení	Rozhraní	IP adresa	Maska podsítě	Brána
R1	Fa0/0	172.16.3.1	255.255.255.0	N/A
	Serial0/0/0	172.16.2.1	255.255.255.0	N/A
...				

Použití kabeláže

Router je do LAN zapojen obvykle přes rozhraní Ethernet nebo Fast Ethernet a s LAN komunikuje přes hub či switch. K propojení routeru a switche/hubu je třeba přímý (*straight-through*) kabel.

Rozhraní 10BASE-TX či 100BASE-TX vyžadují UTP kabel kategorie 5 a lepší. Při propojení routeru přímo k počítači či jinému routeru je třeba křížený (*crossover*) kabel.

Je třeba použít správné rozhraní; mnohé konektory pro různá rozhraní vypadají stejně. Např. rozhraní pro Ethernet, ISDN BRI, konzolový, integrovaný CSU/DSU⁸ a Token Ring používají stejný osmi-pinový konektor – RJ-45, RJ-48 nebo RJ-49.

Pro služby WAN potřebuje zákazník vybavení, což je často router jako DTE. Je připojen k poskytovateli služeb skrze DCE jednotku, tou bývá modem nebo CSU/DSU; ta slouží k převodu dat z DTE do podoby vhodné pro poskytovatele služeb WAN.

Nejčastějšími rozhraními routerů pro WAN jsou sériová rozhraní (konektor SmartSerial nebo DB60) (za ním je ale ještě připojen modem nebo CSU/DSU, telekomunikační vedení a na druhé straně opět modem nebo CSU/DSU a sériová linka vedoucí do dalšího směrovače). Je třeba dát pozor na typ kabelu, typ konektoru a zda jde o DTE či DCE jednotku (**DTE nebo DCE stranu V.35 kabelu, DCE je tam, kde je konektor V.35 Female na opačné straně ke SmartSerial rozhraní**). Z příkazové řádky (CLI) směrovače stranu sériového kabelu na portu zjistíte příkazem **show controllers ...**.

Poznámka k zapojení dvoudílného sériového kabelu se zástrčkou *SmartSerial* do výměnného modulu rozhraní (WIC-2A/S) směrovačů C1720:

- **delší zástrčku SmartSerial** (strana DCE) zapojit **do spodní zásuvky Serial0** (na tomto rozhraní typu DCE nezapomeňte nastavit takt hodin „clock rate“)
- **kratší zástrčku SmartSerial** (strana DTE) zapojit **do horní zásuvky Serial1**

Připojení LAN (propojení mezi zařízeními)

Port nebo připojení	Typ portu	Připojené do	Barva (originální kabelu Cisco)	Kabel
Ethernet	RJ-45	Ethernet switch	Světle žlutá	RJ-45
T1/E1 WAN	RJ-48C/CA81A	T1 nebo E1	Světle zelená	RJ-48 T1
Konzole	8 pin	COM port počítače	Světle modrá	Rollover RJ-45 (redukce DB-9)
AUX	8 pin	Modem	Černá (světle modrá)	Rollover RJ-45 (redukce DB-25)

8 Digitální modem (*Channel Service Unit/Data Service Unit*).

BRI S/T	RJ-48C/CA81A	zařízení NT1 nebo Private Integrated Network Exchange (PINX)	Oranžová	RJ-45
BRI U WAN	RJ-49C/CA11A	ISDN	Oranžová	RJ-45

Určení typu kabelu propojujícího zařízení

<i>Jestliže zařízení A má/je:</i>	<i>Jestliže zařízení B má/je:</i>	<i>Potom použijte tento kabel:</i>
Sériový (COM) port počítače	Konzolový port směrovače/přepínače	Konzolový (Rollover)
Síťová karta (NIC) počítače	Switch	Přímý (Straight-through)
Síťová karta (NIC) počítače	Síťová karta (NIC) počítače	Překřížený (Crossover)
Switch port	Ethernetový port směrovače	Přímý (Straight-through)
Switch port	Switch port	Překřížený (Crossover) (vypněte uplink)
Ethernetový port směrovače	Ethernetový port směrovače	Překřížený (Crossover)
Síťová karta (NIC) počítače	Ethernetový port směrovače	Překřížený (Crossover)
Sériový port směrovače	Sériový port směrovače	Sériový DCE/DTE Cisco

Propojení konektorů pro různé typy kabelů

<i>Straight-Through Cable 568A/568A</i>	<i>Crossover Cable 568A/568B</i>	<i>Rollover Cable</i>
Pin 1 – Pin 1	Pin 1 – Pin 3	Pin 1 – Pin 8
Pin 2 – Pin 2	Pin 2 – Pin 6	Pin 2 – Pin 7
Pin 3 – Pin 3	Pin 3 – Pin 1	Pin 3 – Pin 6
Pin 4 – Pin 4	Pin 4 – Pin 4	Pin 4 – Pin 5
Pin 5 – Pin 5	Pin 5 – Pin 5	Pin 5 – Pin 4
Pin 6 – Pin 6	Pin 6 – Pin 2	Pin 6 – Pin 3
Pin 7 – Pin 7	Pin 7 – Pin 7	Pin 7 – Pin 2
Pin 8 – Pin 8	Pin 8 – Pin 8	Pin 8 – Pin 1

Standardy 568A a 568B

<i>568A Standard</i>				<i>568B Standard</i>			
<i>Pin</i>	<i>Barva</i>	<i>Pár</i>	<i>Popis</i>	<i>Pin</i>	<i>Barva</i>	<i>Pár</i>	<i>Popis</i>
1	Bílá/zelený	3	RecvData +	1	Bílá/oranžový	2	TxData +
2	Zelená	3	RecvData -	2	Oranžová	2	TxData -

3	Bílá/oranžový	2	Txdata +	3	Bílá/zelený	3	RecvData +
4	Modrá	1	Nepoužitý	4	Modrá	1	Nepoužitý
5	Bílá/modrý	1	Nepoužitý	5	Bílá/modrý	1	Nepoužitý
6	Oranžová	2	TxData -	6	Zelená	3	RecvData -
7	Bílá/hnědý	4	Nepoužitý	7	Bílá/hnědý	4	Nepoužitý
8	Hnědá	4	Nepoužitý	8	Hnědá	4	Nepoužitý

Tx – vysílání

Recv – příjem

Typy linek WAN

- Pronajatá linka – synchronní sériová,
- Přepínaný okruh – asynchronní sériová, ISDN L1,
- Přepínané pakety – synchronní sériová – poskytovatel.

Zkoumání obsahu směrovací tabulky a stavu rozhraní**Router1>show ip route**

Codes: C – connected, S – static, I – IGRP, R – RIP, M – mobile, B – BGP
 D – EIGRP, EX – EIGRP external, O – OSPF, IA – OSPF inter area
 N1 – OSPF NSSA external type 1, N2 – OSPF NSSA external type 2
 E1 – OSPF external type 1, E2 – OSPF external type 2, E – EGP
 i – IS-IS, L1 – IS-IS level-1, L2 – IS-IS level-2, ia – IS-IS inter area
 * – candidate default, U – per-user static route, o – ODR
 P – periodic downloaded static route

Gateway of last resort is not set

```

C    192.168.1.0/24 is directly connected, FastEthernet0/0
S    192.168.2.0/24 [1/0] via 192.168.3.1
      192.168.3.0/30 is subnetted, 1 subnets
C        192.168.3.0 is directly connected, FastEthernet0/1
  
```

Router2#show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	192.168.2.254	YES	manual	up	up
FastEthernet0/1	192.168.3.1	YES	manual	up	up

```
Vlan1           unassigned      YES manual administratively down down
Router2#
```

Pole **status** (stav) ve výpisu **show ip interfaces brief** (i v jiných výpisech) může nabývat následujících hodnot a významů:

- **up** - rozhraní je zapnuto/pracuje
- **down** – rozhraní je vypnuto/nepracuje
- **administratively down** – rozhraní je vypnuto administrátorem, respektive nebylo povoleno při nastavení.

Stav rozhraní a linkového protokolu:

1. Pokud je rozhraní down, pak je down také linkový protokol, protože neexistuje funkční médium. Pokud je rozhraní administratively down, znamená to že je rozhraní administrátorem vypnuté.
2. Stav linkového protokolu, spolu s protokolem sítě LAN, který nad ním pracuje. Pokud je linkový protokol down, není na druhé straně zapnuto a nastaveno rozhraní nebo není kabel v rozhraní.

Stav rozhraní/protokolu a typy možné chyby

Interface	Protokol linky	Typ chyby
UP	UP	L1 a L2 OSI modelu pracují v pořádku. Případné chyby jsou výsledkem činnosti vyšších vrstev (L3-L7).
UP	DOWN	Chyba na L2 OSI modelu. - Chyba protokolu na L2 nebo chyba zapouzdření L2 (například značkování rámců ve VLAN protokolem IEEE 802.1Q). Na sériové lince chybějící hodiny na straně DCE (show controllers serial ...). Nepřijata zpráva typu keepalive (protilehlý směrovač nemá zapnuté síťové rozhraní).
DOWN	DOWN	Závada na L1 OSI modelu. - Kabely, fyzické rozhraní, další routery musí být prověřeny na přítomnost napájení a správnou instalaci a konfiguraci.
DOWN	UP	Duplikace MAC adresy v lokální síti připojené k rozhraní Ethernet, nebo chyba servisního modulu na interní rozšiřující kartě. Na routerech lze administrativně měnit (klonovat) MAC adresu rozhraní, na rozdíl od běžné (starší) síťové karty.
ADMINISTRATIVELY DOWN	DOWN	Síťové rozhraní je administrativně vypnuto (není zapnuto administrátorem).

Příkaz show interfaces

Jeden z nejdůležitějších příkazů show je **show interfaces**, který vypíše status a statistiky na všech portech směrovače. **Show interfaces <jméno rozhraní>** vypíše stav a statistiky požadovaného rozhraní (např. show interfaces serial 0/0). Pomocí show interfaces se mohou zjistit problémy na fyzické vrstvě, hardware a logické vrstvě nebo software.

Další informace vypsané pomocí **show interfaces** o rozhraní

- IP adresa ,
- MAC adresa ,
- maska podsítě ,
- statistické údaje o síti ,
- poslední vynulování čítače
- výskyt chyb

```
Router1#sh interfaces fa0/1
```

```
FastEthernet0/1 is up, line protocol is up (connected)
  Hardware is Lance, address is 0005.5ec2.7b02 (bia 0005.5ec2.7b02)
  Internet address is 192.168.3.2/30
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set
  ARP type: ARPA, ARP Timeout 04:00:00,
  Last input 00:00:08, output 00:00:05, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue :0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    0 input packets with dribble condition detected
    0 packets output, 0 bytes, 0 underruns
    0 output errors, 0 collisions, 1 interface resets
    0 babbles, 0 late collision, 0 deferred
    0 lost carrier, 0 no carrier
    0 output buffer failures, 0 output buffers swapped out
```

```
Router1#
```

Průzkum přímo připojených sítí a protokol CDP

CDP (Cisco Discovery Protocol) je proprietární protokol vytvořený firmou Cisco běžící na 2. vrstvě (L2). Tento protokol běží na drtivé většině síťových zařízeních (Cisco ale i jiných firem) a je používán na sdílení informací o jiných **přímo připojených sousedních (*neighbor*⁹) zařízeních (na L2)**¹⁰.

CDP se používá k získání HW platformy, IP adresy a názvu rozhraní sousedních zařízení. CDP je nezávislý na přenosovém médiu a linkovém (i síťovém) protokolu a běží na Cisco routerech, můstcích, přístupových serverech i přepínačích¹¹.

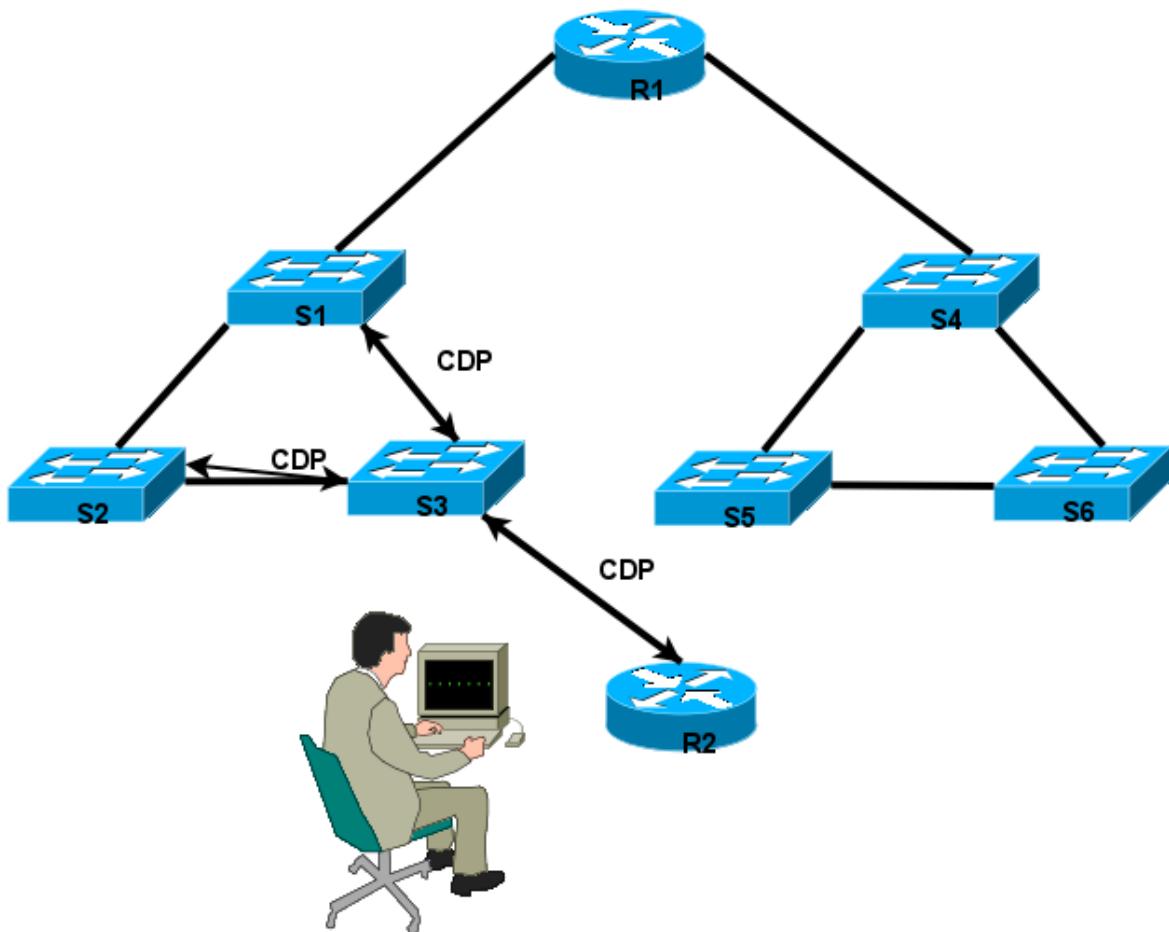
Každé zařízení, které je konfigurované pro CDP posílá periodické zprávy, tzv. oznamovače (*advertisements*). Oznamovače obsahují různé informace jako je: HW platforma, IP adresa a název rozhraní, doba uchování (*holdtime*) což je čas, po kterou si přijímací zařízení ponechá konkrétní CDP informaci, než ji smaže. Každé zařízení také zjišťuje obsah CDP zpráv odeslaných z jiných zařízení, aby zjistilo informace o jednotlivých sousedech.

⁹ Neighbor = soused v americké angličtině. V britské angličtině častěji *neighbour*.

¹⁰ Pojem soused: na L2 jde skutečně o fyzicky sousední zařízení, (sousední na L3 = je ve stejné broadcastové doméně, ve stejně síti – sousední směrovač).

¹¹ Existují i klienti a agenti (= servery) CDP pro hostitelské počítače.

CDP - Cisco Discovery Protocol



Příkazy pro protokol CDP

Příkaz	Účel
cdp enable	Povolí CDP na konkrétním rozhraní – v konfiguračním režimu rozhraní (config-if)# (je implicitně zapnuté)
cdp advertise-v2	Povolí CDPv2 na rozhraní
clear cdp counters	Resetuje čítače provozu na nulu – v enable režimu#
show cdp	V privilegovaném (enable) režimu: Zobrazuje interval mezi vysíláními CDP oznamovačů (CDP advertisement), počet sekund, po které je oznamovač platný pro daný port a verzi oznamovače.
show cdp entry entry-name [protocol version]	Zobrazuje informace o určitém sousedovi, které mohou být omezeny protokolem nebo verzí.
show cdp interface [type number]	Zobrazuje informace o rozhraních, na kterých je CDP umožněn
show cdp neighbors [type number]	Zobrazuje typy zařízení, které byly objeveny, jméno za-

[detail]	řízení, číslo a typ lokálního rozhraní nebo portu, počet sekund, po které je oznamovač platný pro port, typ zařízení, číslo typu, duplexní mód, VTP doménu asociovánou k sousednímu zařízení, pokud bylo použito klíčové slovo detail.
cdp run	(Je implicitně zapnuté.) CDP lze v globálním konfiguračním režimu vypnout příkazem (config)# no cdp run . V tomto případě je třeba ho znovu zapnout příkazem (config)# cdp run .

Co zobrazují výpisy z CDP:

- **Device ID** (identifiers) – například nastavení host name u přepínače
- Address list – alespoň jednu adresu na síťové vrstvě pro každý podporovaný protokol
- **Port ID** (identifiers) – jméno lokálního a vzdáleného portu v podobě řetězce ASCII znaků například ethernet0
- **Capability** – například zda je zařízení switch nebo router
- **Platform** – HW platforma zařízení, například „Cisco 7200 series router“

Switch>show cdp

Global CDP information:

```
Sending CDP packets every 60 seconds
Sending a holdtime value of 180 seconds
Sending CDPv2 advertisements is enabled
```

Switch>show cdp neighbors

```
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone
```

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
S1	Fas 0/1	146	S	2950	Fas 0/1
R3	Fas 0/0	146	R	C2600	Fas 0/0

Switch>show cdp neighbors detail

Device ID: R3

Entry address(es):

IP address : 192.168.3.254

Platform: cisco C2600, Capabilities: Router

Cisco NetAcad: CCNA Exploration - Routing Protocols and Concepts – studijní materiál

```
Interface: FastEthernet0/0, Port ID (outgoing port): FastEthernet0/0
```

```
Holdtime: 146
```

```
Version :
```

```
Cisco Internetwork Operating System Software
```

```
IOS (tm) C2600 Software (C2600-I-M), Version 12.2(28), RELEASE SOFTWARE (fc5)
```

```
Technical Support: http://www.cisco.com/techsupport
```

```
Copyright (c) 1986-2005 by cisco Systems, Inc.
```

```
Compiled Wed 27-Apr-04 19:01 by miwang
```

```
advertisement version: 2
```

```
Duplex: full
```

Obsah vyrovávací paměti ARP na směrovači

```
R1#show arp
Protocol Address          Age (min)  Hardware Addr   Type      Interface
Internet 192.168.1.1           0    0090.2BD0.EA46  ARPA     FastEthernet0/0
Internet 192.168.1.14          -    0001.427A.D501  ARPA     FastEthernet0/0
Internet 192.168.1.29          0    00D0.9731.5302  ARPA     FastEthernet0/1
Internet 192.168.1.30          -    0001.427A.D502  ARPA     FastEthernet0/1
R1#
```

Řádky se stářím záznamu (*Age*) = - jsou MAC adresy lokálních rozhraní na směrovači.

Ověření předchozího výpisu:

```
R1#sh ip int br
Interface          IP-Address      OK? Method Status      Protocol
FastEthernet0/0    192.168.1.14    YES manual up       up
FastEthernet0/1    192.168.1.30    YES manual up       up
```

Zjišťování změn ve směrovací tabulce

Pokud vypadne přímo připojená síť, ze směrovací tabulky okamžitě zmizí všechny směry (cesty), které se na tuto přímo připojenou síť odkazují. (Sítě, jejichž „next-hop“ ve vypadlé přímo připojení síti leží, nebo sítě, které do ní přepínají na odchozím rozhraní.)

Uvedeme příklad: ladění (*debugging, debug*): Zapnuto. Vytažen kabel ze směrovače (příslušné cesty jsou smazány ze směrovací tabulky). Ladění vypnuto.

POZOR: ladění (*debugging*) velice zatěžuje CPU směrovače, zapínejte ho proto pouze na ne-zbytnou dobu pro dohledávání chyb (a nikoliv při běžném provozu).

Zapnutí ladění:

```
Router1#debug ip routing
```

Cisco NetAcad: CCNA Exploration - Routing Protocols and Concepts – studijní materiál

```

IP routing debugging is on
Router1#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state
to down
RT: interface FastEthernet0/0 removed from routing tableRT: del 192.168.1.0 via
0.0.0.0, connected metric [0/4294967295]
RT: delete network route to 192.168.1.0
RT: NET-RED 192.168.1.0/24

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state
to up
RT: interface FastEthernet0/0 added to routing tableRT: SET_LAST_RDB for
192.168.1.0/24
    NEW rdb: is directly connected

RT: add 192.168.1.0/24 via 0.0.0.0, connected metric [0/0]
RT: NET-RED 192.168.1.0/24

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state
to down
RT: interface FastEthernet0/1 removed from routing tableRT: del 192.168.3.0 via
0.0.0.0, connected metric [0/4294967295]
RT: delete network route to 192.168.3.0
RT: NET-RED 192.168.3.0/30
RT: del 192.168.2.0 via 192.168.3.1, static metric [1/0]
RT: delete network route to 192.168.2.0
RT: NET-RED 192.168.2.0/24

```

Vypnutí ladění:

```

Router1#undebug all
All possible debugging has been turned off
Router1#

```

Statická cesta s adresou dalšího skoku

Nastavení:

```
Router1(config)#ip route 192.168.2.0 255.255.255.0 192.168.3.1
```

Obsah směrovací tabulky:

```

Router1#show ip route
<vynecháno>
C    192.168.1.0/24 is directly connected, FastEthernet0/0

```

```
S      192.168.2.0/24 [1/0] via 192.168.3.1
    192.168.3.0/30 is subnetted, 1 subnets
C      192.168.3.0 is directly connected, FastEthernet0/1
Router1#
```

V tomto případě se při směrování do vzdálené sítě (192.168.2.0) musí nejprve nalézt adresu dalšího skoku (next-hop) a po jejím nalezení **rekurzivně vyhledat** (*recursive lookup*) odchozí rozhraní do přímo připojené sítě, ve které je následující skok.

Statická cesta s odchozím rozhraním

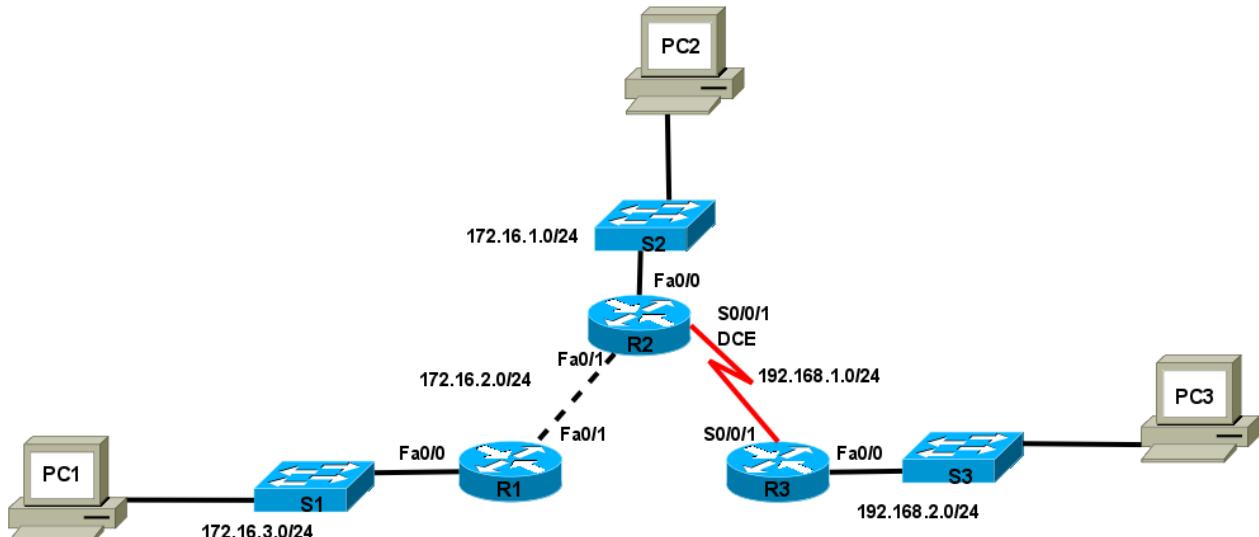
Nastavení:

```
Router2(config)#ip route 192.168.1.0 255.255.255.0 Fa0/1
```

Obsah směrovací tabulky:

```
Router2#sh ip route
<vynecháno>
S      192.168.1.0/24 is directly connected, FastEthernet0/1
C      192.168.2.0/24 is directly connected, FastEthernet0/0
    192.168.3.0/30 is subnetted, 1 subnets
C      192.168.3.0 is directly connected, FastEthernet0/1
Router2#
```

V tomto případě se staticky definovaná vzdálená síť (*remote network*) jeví jako přímo připojená (*directly connected*) a po nalezení cesty (směru) se může paket rovnou odeslat příslušným směrem (ber rekurzivního vyhledávání adresy dalšího skoku). Tato varianta je všeobecně doporučována. Její nevýhoda se ale projeví při Ethernetové síti s vícenásobným přístupem (*multi-access network*) tj.v síti s více branami, kdy se nedá naleznout MAC adresa následujícího skoku pro zapouzdření paketu do rámce.

Odchozí rozhraní a nebo adresa dalšího přeskoku (Exit interface and next-hop address)

```
R1(config)#ip route 192.168.2.0 255.255.255.0 FastEthernet 0/1
R1(config)#ip route 192.168.2.0 255.255.255.0 172.16.2.2
```

Sumarizace (agregace) cest

Zatím jsme se vždy zabírali stavem, kdy každý směrovač má ve své směrovací tabulce informace o všech sítích v určité skupině sítí, kde nastavujeme statické směrování. Pokud máme hierarchickou stromovou strukturu podsítí, je vhodné zavést **sumarizaci cest** (*route summarization*), také známé pod pojmem **agregace cest** (*route aggregation*). Informace o všech sítích ve skupině mají pak pouze směrovače v této skupině (podsítí) a ostatní směrovače (mimo skupinu) mají pak pouze cestu na hraniční směrovač celé skupiny, tato cesta pak ukazuje na sumarizovanou síť (= síť s kratší maskou než podsítě, která pokrývá všechny adresy v jednotlivých podsítích). Šetří se tím rádky ve směrovacích tabulkách a směrování probíhá rychleji.

Sumarizace se s výhodou využije i pro **potlačení vlivu vypadávajících, kolísajících linek** (*flapping link*), kdy ve vzdálených sítích různé linky vypadávají – vypínají se a opět se zapínají – a mění se tak konkrétní trasa směrování, v případě využití summarizace cest se to již ale neprojeví ve směrovací tabulce na vzdálených směrovačích.

Příklad

Máme skupinu podsítí s rozsahem: 192.168.1.0/27 až 192.168.1.120/29. Určete síť a masku pro summarizaci ve směrovací tabulce.

Postup: najdeme síť s nejmenší a největší adresou (určíme adresu všesměrového vysílání (broadcast) pro tuto síť s největší adresou) a odečteme od sebe. Inverzí rozdílu (včetně eventuálního doplnění binárních jedniček vpravo za vedoucí jedničku) získáme masku a odmaskováním zadaných podsítí touto maskou dostaneme sumární síť. $192.168.1.127 - 192.168.1.0 = 0.0.0.127$ inverze (dvojkový doplněk) je 255.255.255.128 tj. /25 a odmaskováním adres podsítí získáme sumární síť 192.168.1.0/25.

Příklad

Máme skupinu třídních sítí 172.16.0.0/16, 172.17.0.0/16 atd. až 172.23.0.0/16.

Ve 2. bajtu je tedy binárně **0001 0000** až **0001 0111** a tedy maska je $8+5=13$. Výsledná summarizovaná síť je tedy 172.16.0.0/13. Maska summarizace /13 (255.248.0.0) je kratší než implicitní třídní maska /16 (255.255.0.0).

Poznámka

CIDR ignoruje omezení hranicemi tříd a umožňuje summarizaci s maskou, která je kratší než je implicitní třídní maska. Této summarizaci se říká **vytváření nadsítí (supernetting)**. **Nadsíť (super-net)** je vždy summarizací cesty na směrovači, ale summarizace cesty není vždy nadsítí.

Tento typ summarizace pomáhá redukovat počet řádek ve směrovacích aktualizacích i počet řádek v lokálních směrovacích tabulkách. Též snižuje využití přenosové kapacity linek pro směrovací aktualizace a vede k rychlejšímu prohledávání směrovacích tabulek při směrování.

Implicitní cesta

Implicitní cesta (*default route, Gateway of last resort*) se nastavuje jako speciální případ statické cesty pro cílové síť mimo naši správu obvykle na našeho poskytovatele - ISP. Na tuto cestu je paket poslán, pokud směrovač nenalezl cílovou síť v předchozích řádkách směrovací tabulky. (Nezapomeňte, že směrovací tabulka je setříděna sestupně podle masky.) Implicitní cesta je ve směrovací tabulce označena hvězdičkou (*) jako kandidát implicitní cesty (*candidate default*).

```
Router(config)#ip route 0.0.0.0 0.0.0.0 [exit-interface | ip-address ]
```

```
Router1#sh ip route
```

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route
```

```
Gateway of last resort is 0.0.0.0 to network 0.0.0.0
```

```
C    192.168.1.0/24 is directly connected, FastEthernet0/0
S    192.168.2.0/24 [1/0] via 192.168.3.1
     192.168.3.0/30 is subnetted, 1 subnets
C        192.168.3.0 is directly connected, FastEthernet0/1
```

```
S* 0.0.0.0/0 is directly connected, FastEthernet0/1
```

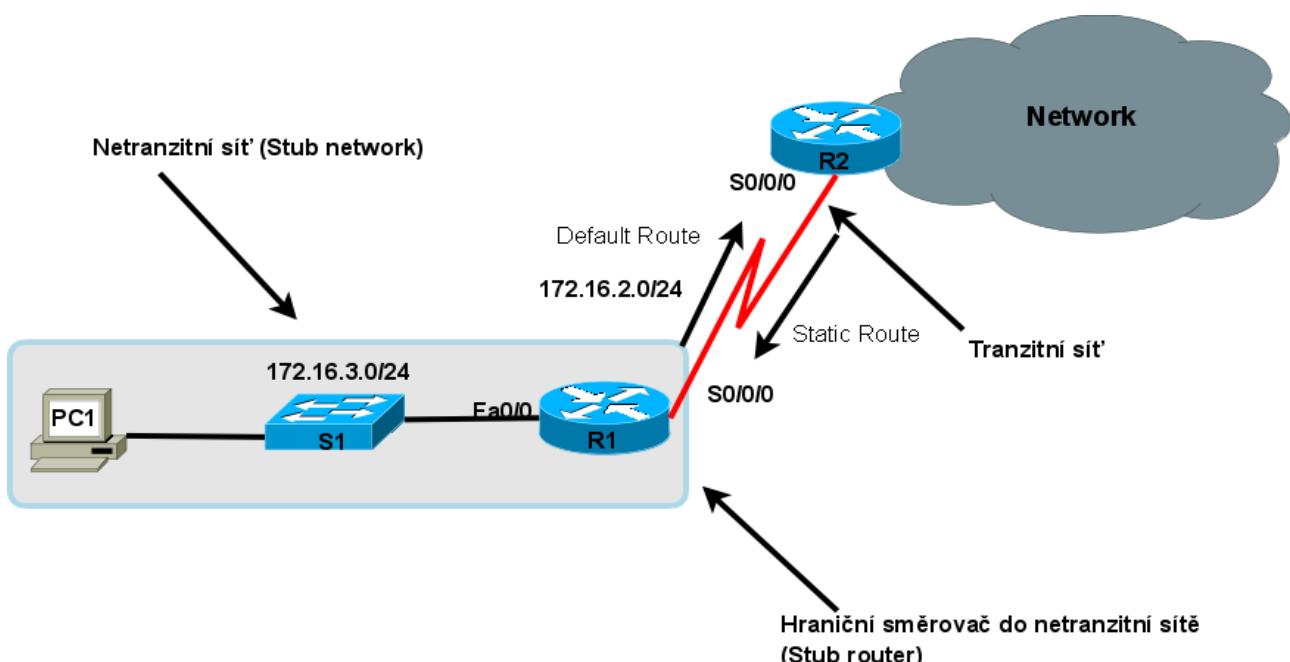
Router1#

Poznámka:

Plná syntaxe příkazu je:

```
Router(config)# ip route prefix mask {ip-address | interface-type interface-number} [distance] [tag] [permanent]
```

- *Distance* je vzdálenost – lze použít k nastavení tzv. **Floating route**, záložní statické cesty, která případně do úvahy pouze při výpadku dynamického směrování (parametr vzdálenost - *distance* se musí nastavit větší než je *administrativní vzdálenost (Administrative Distance, AD)*¹² dynamického směrovacího protokolu).
- *Tag* – poskytuje metodu pro rozlišení mezi interními cestami (získanými od stejného směrovacího protokolu jako na směrovači) a externími cestami (získaných od jiných protokolů). Tento nepovinný volitelný atribut můžete přidat během redistribuce směrovacích protokolů.
- *Permanent* - cesta nezmizí ze směrovací tabulky, ani když spadne odchozí rozhraní.



Definice implicitní sítě

Jiný způsob jak nastavit bránu poslední záchrany (*gateway of last resort*) je nastavení implicitní sítě (pro přilehlou, přímo připojenou, síť).

```
Router(config)# ip default-network network-number
```

Ve směrovací tabulce je potom (místo statické implicitní cesty) přímo připojená síť (C) jako kandidát na implicitní síť, například :

```
C* 10.0.0.0/8 is directly connected, Serial0/0
```

Když se rozhodujete, zda použít implicitní statickou cestu nebo implicitní síť, berte v potaz, zda po-

¹² Administrativní vzdálenosti viz kapitola 3.

třebujete, aby případně použitý dynamický směrovací protokol propagoval tuto implicitní cestu. Příkaz **ip default-network** to udělá za vás. Jinak musíte propagaci implicitní cesty nastavit v konfiguraci dynamického směrovacího protokolu samostatně (pomocí tzv. redistribuce cest)¹³, probereme ji později pro jednotlivé konkrétní protokoly) a pokud ne, zůstane implicitní cesta pouze na lokálním směrovači. Volba nastavení (statické implicitní cesty nebo implicitní sítě) na hraničním směrovači nemá vliv na statické směrování na nehraničních směrovačích.

Správa a modifikace cest

Pokud chcete některou cestu odstranit ze směrovací tabulky, provedete stejným příkazem, jako byla nastavena a přidáním příkazu **no**. Pouhé přidání nové cesty do stejné sítě nebo na stejném rozhraní by vedlo k existenci dvou cest.

To znamená, pokud chcete vytvořit novou cestu do stejné cílové sítě, musíte nejprve odstranit starou cestu a potom vložit novou.

Hledání a odstraňování chyb statické cesty

Příkazy:

- **ping**
- **traceroute**
- **show ip route**
- **show ip interface brief**
- **show cdp neighbors detail**

Odstraňování problémů se statickým směrováním

Hledání a odstraňování chyb může někdy být docela pracné. Je proto vhodné mít pečlivě zdokumentovanou strukturu sítě a nejprve ověřit, že je to tak opravdu nastavené.

Na ověření dostupnosti cílového zařízení slouží příkaz:

```
ping <cílová_adresa>
```

Zobrazit jednotlivé směrovače, kterými musí paket projít do cílového zařízení umožňuje příkaz:

```
traceroute <cílová_adresa>
```

Odpovědi (a jejich významy) na příkaz ping na směrovači

Znak	Popis
!	Přijetí jedné odpovědi (jednoho paketu).
.	Čas pro odpověď síťového serveru vypršel.
U	Cíl je nedostupný (<i>Unreachable</i>) – přijat PDU.

13 Podle konkrétního směrovacího protokolu. RIP a OSPF: default-information originate, EIGRP: redistribute static.

C	Zahlcení (<i>Congestion</i>) – přijat paket.
I	Uživatel přerušil (<i>Interrupted</i>) test.
?	Neznámý typ paketu.
&	Překročena životnost paketu.

Odpovědi (a jejich významy) na příkaz traceroute na směrovači

Znak	Popis
nn msec	Pro každý uzel, doba cyklu (v milisekundách) pro určený počet pokusů.
*	Doba testu vypršela. V určené době nebyla přijata žádná odpověď.
?	Neznámá chyba.
A	Administrativně nedostupné. Obvykle tento výstup indikuje, že ACL ¹⁴ blokuje provoz.
H	Hostitel je nedostupný.
N	Síť (<i>Network</i>) je nedostupná (mimo rozsah).
P	Protokol je nedostupný.
Q	Zdroj vypnut (<i>Quenche</i>).
P	Port je nedostupný.

Příkazy pro kapitolu 2, Statické směrování

Příkaz (Command)	Popis (Description)
Router# show controllers serial 0/0/0	Zobrazí, která strana kabelu je připojená do rozhraní: DCE nebo DTE
Router# show cdp neighbors	Zobrazí přehledné sumární informace o přímo připojených L2 zařízeních; používá proprietární protokol Cisco (CDP)
Router# show cdp neighbors detail	Zobrazí úplné informace o přímo připojených L2 zařízeních; používá proprietární protokol Cisco (CDP)
Router(config)# cdp run	Povolí Cisco L2 protokol pro celý směrovač.
Router(config-if)# cdp enable	Povolí Cisco L2 protokol pro rozhraní.
Router(config)# ip route 10.0.0.0 255.0.0.0 172.16.0.1	Nastaví statickou cestu do 10.0.0.0/8 s dalším skokem (<i>next hop</i>) 172.16.0.1 .
Router(config)# ip route 10.0.0.0 255.0.0.0 Seri-	Nastaví statickou cestu do 10.0.0.0/8 s odchozím

14 ACL – Access Control List – přístupový seznam – povolení nebo blokování síťového provozu pro adresu popřípadě protokol nastavené administrátorem směrovače.

al0/0/0	(výstupním) rozhraním Serial 0/0/0 .
Router(config)# ip route 172.16.0.0 255.240.0.0 Serial 0/0/0	Nastaví sumární statickou cestu pro všechny adresy spadající do privátní sítě třídy B v rozsahu od 172.16.0.0/16 do 172.31.0.0/16; použije odchozí rozhraní Serial 0/0/0
Router(config)# ip route 0.0.0.0 0.0.0.0 Serial0/1/0	Nastaví implicitní statickou cestu (<i>default static route</i>) s odchozím rozhraním Serial 0/1/0 .

Volitelné příkazy pro kapitolu 2, Statické směrování

Příkaz (Command)	Popis (Description)
Router(config)# ip route 10.0.0.0 255.0.0.0 172.16.0.1 200	Nastaví plovoucí statickou cestu (floating route) do 10.0.0.0/8 s dalším skokem (<i>next hop</i>) 172.16.0.1 s administrativní vzdáleností (AD) = 200 . Administrativní vzdálenost definuje důvěryhodnost cesty. AD=0 pro přímo připojenou síť AD=255 pro neznámou, nedůvěryhodnou cestu
Router(config)# ip route 10.0.0.0 255.0.0.0 172.16.0.1 permanent	Nastaví permanentní statickou cestu (permanent route) do 10.0.0.0/8 s dalším skokem (<i>next hop</i>) 172.16.0.1. Tato permanentní cesta se neodstraní ze směrovací tabulky po výpadku rozhraní do přilehlé sítě 172.16.0.0.

Komplexní praktické laboratorní cvičení – statické směrování

Mějme 3 směrovače R1, R2 a R3 (Cisco 1841) jsou zapojené do kruhu a ke každému je připojeno jedno PC: PC1, PC2 a PC3.

Mezi R1 a R2 je sériová linka s adresou sítě a maskou (délkou prefixu): 172.16.2.32/27.

Mezi R2 a R3 je FastEthernet s adresou sítě a maskou (délkou prefixu): 172.16.1.0/24.

Mezi R3 a R1 je FastEthernet s adresou sítě a maskou (délkou prefixu): 172.17.2.0/25.

Na jednotlivých směrovačích R1, R2 a R3 je připojeno PC1, PC2 a PC3 v sítích 192.168.1.0/24, 192.168.2.0/24 a 192.168.3.0/24.

IP adresy rozhraní jsou vypočteny dle následujících firemních směrnic vlastníka sítě („best practices“): na směrovačích jsou adresy těsně pod adresou všesměrového vysílání v příslušné síti a na klientech jsou IP adresy těsně nad adresou sítě dané sítě.

Zapojení zprovozněte pomocí statického směrování (cestu definujte vždy nejkratším směrem a směrem na odchozí rozhraní (*outgoing interface*), nikoliv na IP adresu dalšího skoku (*next hop*)).

Postup práce:

1. Nejprve si nakreslete topologické schéma zapojení včetně adres sítí.
2. Vyplňte (doplňte) následující tabulku adres síťových rozhraní:

Zařízení	Rozhraní	IP adresa	Maska	Brána
R1	Fa0/0	172.17.2.126	255.255.255.128	-
	Eth0/0/0	192.168.1.254	255.255.255.0	-
	S0/1/0	172.16.2.61	255.255.255.224	-
R2				-
R3				-
PC1	Fast Ethernet	192.168.1.1	255.255.255.0	192.168.1.254
PC2	Fast Ethernet	192.168.2.1	255.255.255.0	192.168.2.254
PC3	Fast Ethernet	192.168.3.1	255.255.255.0	192.168.3.254

3. Propojte zapojení odpovídající kabeláží, případě, při nedostatku portů, přidejte odpovídající zásuvné moduly. (POZOR: nelze použít 4 portový switch modul – nelze na něm nastavovat IP adresy.) Názvy síťových rozhraní si doplňte do topologického schématu sítě.
4. Nastavte na správná rozhraní adresy a masky a zprovozněte jednotlivé linky (zkontrolujte pomocí příkazu **show ip interface brief** funkčnost vrstev L1, L2 (L3)).
5. Nastavte příslušné jméno hostitele (**hostname**) (R1, R2 a R3) na každém ze směrovačů.
6. Na každém směrovači zprovozněte 4 linky Telnet s heslem: class.
7. Na každém směrovači zaheslujte šifrovaným heslem režim enable, heslo: cisco.
8. Zkontrolujte směrovací tabulky příkazem **show ip route** na každém směrovači (v této chvíli byste měli vidět na každém 3 přímo připojené sítě).
9. Na každém směrovači nastavte implicitní cestu ve směru hodinových ručiček. Ověřte funkčnost popřípadě opravte chyby. Všimněte si, že je to asymetrické směrování. (**Vznikla směrovací smyčka**¹⁵ pro všechny IP adresy ležící mimo zadaných 6 sítí.) Po vyzkoušení **ODSTRAŇTE**. Implicitní cesty se **nepoužívají** pro směrování do předem známých sítí. Takto navržené implicitní cesty do kruhu zapříčinují vznik směrovací smyčky.
10. Na každém směrovači **nastavte statické cesty** do tří vzdálených sítí definované na odchozí rozhraní, směr zvolte vždy nejkratší cestou.
11. Zkontrolujte funkčnost a opravte případné chyby. (Použijte příkazy: **ping**, **traceroute**, **show ip route**, **show ip interface brief**).
12. Vyzkoušejte si reakce (změny ve směrovací tabulce) v závislosti na výpadku přímo připojené sítě (o výpadku vzdálené se směrovač při statickém směrování sám nedozví).
13. Nastavte a zaheslujte telnet a zaheslujte šifrovaným heslem režim enable.
14. Zkuste nakreslit zapojení znova pouhým průzkumem pomocí protokolu CDP příkazem **show cdp neighbors detail** a přihlašováním se na sousední zařízení pomocí Telnet. (Před tím „jakoby zapomeňte“, co o síti již víte.)

15 Směrovací smyčka viz kapitola 4.

Časté a „oblíbené“ chyby

Typy častých chyb vyskytující se na prvních třech vrstvách modelu OSI:

Vrstva 1:

- Porušené kabely,
- Odpojené kabely,
- Kabely zapojené **do nesprávných portů**,
- Použití nesprávných typů kabelů (přímé, křížené, roll-over apod),
- Problémy s příjemem a vysíláním signálu,
- DTE/DCE kabely (přehozené strany kabelu),
- Vypnuté porty (nezapnuté příkazem *no shutdown*)
- Vypnutá zařízení (vypnuté napájení).

Vrstva 2:

- Nesprávně nastavená sériová rozhraní,
- Nesprávně nastavená ethernetová rozhraní,
- Nesprávně nastavené zapouzdření (budeme brát později - ve třetím semestru),
- Nesprávně nastavené hodiny (*clock rate*) na sériovém rozhraní (souvisí s DCE kabelem na L1),
- Problémy se síťovými kartami (HW).

Vrstva 3:

- Nekorektní IP adresy (například každý konec kabelu v jiné IP síti),
- Nekorektní masky podsítě (jiná maska => jiná síť, byť se třeba síť překrývají),
- Použití zakázaných IP adres (použita například adresa sítě nebo broadcastu) – odhalí přímo IOS směrovače.
- Překryv sítí (*overlapping*) (na jednom směrovači odhalí přímo IOS směrovače).
- Na klientech špatně nastavena (nebo vůbec nenastavena) výchozí brána,
- Špatně nastavené statické směrování (např. vznik směrovací smyčky),
- Vypnutý směrovací protokol,
- Nesprávné nastavení směrovacího protokolu.

Testování sítí

Většina síťových problémů souvisí s nemožností připojit se k požadovanému hostiteli nebo službě. Problémy s konektivitou mají celou řadu podob, jako je například vypršení časového limitu při

pokusu o spojení, pokus o terminálové spojení bez příslušné odpovědi ze strany hostitele a podobně.

Zapamatujte si že,

- Pokud je konkrétní hostitel A dostupný z hostitele B v jiné síti, naprosto to nevypovídá nic o tom, zda je dostupný hostitel B z opačného směru z A (pokud neuvažujeme po-tvrzování). Směrování jedním směrem je nezávislé na směrování opačným směrem.
- Vznik **směrovací smyčky** nezávisí na existenci fyzické smyčky na médiu.

„Chybičky“

Co bychom již měli opravdu znát:

1. Každý klient IP sítě musí mít nastaveno: IP adresu, masku podsítě a implicitní bránu (a DNS server, pokud ho používáme).
2. Každé použité síťové rozhraní na směrovači musí být zapnuté a mít nastavenou IP adresu a masku.
3. Všechny IP adresy hostitelů připojené k jednomu rozhraní (portu) směrovače musí ležet v jedné (stejné) IP síti (podsíti).
4. Každý port jednoho směrovače musí ležet v jiné síti (podsíti). Vznik překryvu sítí směrovač ohlásí (*overlap*).
5. Ve skupině routerů pod jednou správou směrování musí být všechny jednotlivé sítě různé.
6. Pro propojení portů je třeba použít správné typy kabelů.
7. Na DCE straně sériového kabelu je třeba mít nastaveny hodiny.
8. Při splnění předchozího, by již neměl být žádný problém při statickém směrování.
9. **Sumarizace** má smysl pouze na hranici (kořenu) třídní sítě. (Setří potom řádky v aktualiza-cích směrovacích tabulek posílaných směrem výše.)
10. Při dynamickém směrování je třeba rozlišit, zda se protokolem přenáší maska či nikoliv (**masku nepřenáší pouze RIPv1**) – a tomu musí odpovídat navržené adresní schéma, to bu-deme brát později.
11. Pokud se nepřenáší maska, směrovací protokol IP adresu odmaskuje implicitní maskou, to znamená, že dvě a více podsítí jedné sítě se mu jeví jako jedna nadsíť (v plné třídě) a pokud jsou tyto sítě ve dvou a více různých směrech v nesouvisejících sítích, tak je problém.
12. Funkční směrování jedním směrem naprostě nezaručuje směrování opačným směrem.
13. Příliš mnoho změn při konfiguraci směrovače „najednou“ může vést k jeho atypickým stavům – je vhodné uložit konfiguraci a restartovat směrovač (#copy run start, #reload).

Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Pravdivé tvrzení ohledně konfigurace statické cesty:
 - a) Směrovače s nastavenou statickou cestou používající adresu dalšího skoku (*next hop*) musí mít buď v této cestě ještě nastaveno odchozí rozhraní (*outgoing interface*) nebo mít ještě jednu cestu se sítí (přímo připojenou), ve které je další skok, a s přiřazeným odcho-zím rozhraním. (V tomto případě se potom při směrování provádí rekurzivní dohledání)

cesty s odchozím rozhraním do adresy dalšího skoku.)

- 2) Ve směrovací tabulce je řádka „S 10.0.0.0/8 [1/0] via 172.16.40.2“. Administrátor má tuto cestu ze směrovací tabulky odstranit. Jakým příkazem?

a) no ip route 10.0.0.0 255.0.0.0 172.16.40.2

- 3) Jaký příkaz by mohl vyprodukovat následující výstup?

Router1>				
1	53 ms	43 ms	36 ms	10.0.0.1
2	106 ms	56 ms	40 ms	192.168.0.2
3	*	*	*	Request timed out

a) traceroute

- 4) Tři charakteristiky statické cesty:

- a) snižuje požadavky na paměť a výpočetní výkon směrovače,
- b) používají se na směrovačích, které jsou připojeny do netrannitních sítí (*stub networks*),
- c) používají se u takových sítí, ze kterých je pouze jedna cesta do konkrétní cílové sítě (= stromová struktura sítě).

- 5) Co je funkcí příkazu „show cdp neighbors“?

- a) Zobrazuje typ portů a HW platformu sousedících směrovačů nebo přepínačů Cisco. (CDP je sice proprietární protokol, ale používají ho i někteří jiní výrobci síťových zařízení.)

- 6) Pravdivé tvrzení ohledně přímo připojených směrů:

- a) Objeví se ve směrovací tabulce pokud je na rozhraní nastavená IP adresa a po vydání příkazu „show interfaces“ se ukáže, že příslušné rozhraní je administrativně zapnuté a L2 protokol linky je spuštěný.

- 7) Spárování příkazů a popisů jejich výstupů:

- a) show ip route = zobrazí všechny známé sítě
- b) show interfaces = zobrazí detailní informace o všech rozhraních
- c) debug ip routing = zobrazí online informace potřebné pro odstraňování závad
- d) show interface brief = zobrazí struční informace o rozhraních (včetně stavu rozhraní a stavu L2 protokolu linky)
- e) show cdp neighbors = zobrazí přímo připojené směrovače
- f) show controllers = zobrazí informace o DTE/DCE nastavených.

Kapitola 3 - Protokoly pro dynamické směrování

V této kapitole se naučíme:

- Popsat roli dynamických směrovacích protokolů a místo těchto protokolů v kontextu návrhu moderních sítí
- Určit několik způsobů jak klasifikovat směrovací protokoly
- Popsat, jak směrovací protokol používá metriku a určit různé druhy metrik, které používají různé dynamické směrovací protokoly
- Určit administrativní vzdálenost (*Administrative Distance, AD*) cesty a popsat její důležitost při průběhu směrování
- Určit některé klíčové informace ve směrovací tabulce
- Realistický pohled na daná existující omezení na zařízeních, protokolech a adresních schématech.

Účel směrovacích protokolů

- Zjištění vzdálených (= ne přímo připojených) sítí (*Discovery of remote networks*)
- Udržování aktuálních směrovacích informací (*up-to-date routing information*)
- Výběr nejlepší cesty do cílových sítí (*the best path to destination networks*)
- Schopnost nalézt novou nejlepší cestu pokud současná cesta již není dále dostupná.

Které jsou **klíčové komponenty směrovacího protokolu?**

- **Datové struktury** – některé směrovací protokoly používají pro svou činnost tabulky nebo databáze. Tyto informace jsou uloženy v RAM.
- **Algoritmus** – algoritmus je konečný seznam kroků potřebných k dosažení určitého cíle. Směrovací protokoly používají algoritmy pro získání směrovacích informací a pro určení nejlepší cesty.
- **Zprávy směrovacího protokolu** (*Routing Protocol Messages*) – směrovací protokoly používají různé typy zpráv ke zjištění sousedních směrovačů, výměnu směrovacích informací a další úkoly, aby zjistily a udržely správné informace o síti.

Cinnost směrovacího protokolu: všechny směrovací protokoly mají stejný účel – zjistit vzdálené sítě a rychle se přizpůsobit při změně topologie. Metoda, kterou směrovací protokol užívá závisí na použitém směrovacím algoritmu (typu směrovacího protokolu) a směrovacím protokolu samotném. Obecně může být cinnost směrovacího protokolu popsána následovně:

- Směrovač posílá a přijímá směrovací zprávy na svých rozhraních.
- Směrovač sdílí směrovací zprávy a směrovací informace s jinými směrovači, které používají stejný směrovací protokol.
- Směrovače si vyměňují směrovací informace, aby zjistily vzdálené (*remote*) sítě.
- Když směrovač zjistí změnu topologie, může tuto změnu oznámit ostatním směrovačům

(pomocí tzv. Oznamovačů (*advertisement*)).

Porovnání dynamického a statického směrování

Charakteristika	Dynamické směrování	Statické směrování
Náročnost konfigurace	Obecně nezávislé na velikosti sítě	Čím větší síť, tím složitější
Požadované znalosti administrátora	Pokročilé	Nejsou třeba zvláštní
Změny topologie	Automatické přizpůsobení	Zásah administrátora je nutný
Škálovatelnost	Vhodné pro malé i velké sítě	Vhodné pro malé sítě
Bezpečnost	Méně bezpečné	Více bezpečné
Spotřeba systémových zdrojů	CPU, paměť, šířka pásma	Žádné zvláštní nejsou třeba
Předvídatelnost	Cesta je závislá na aktuální topologii	Cesta do cíle je vždy stejná
Výhody Pro (Pros)	<ul style="list-style-type: none"> • Administrátor má méně práce s údržbou při přidání nebo odpojení sítě. • Protokoly automaticky reagují na změny topologie. • Konfigurace je méně náchylná ke vzniku chyby. • Více škálovatelné, zvětšení sítě obvykle nepředstavuje problém. 	<ul style="list-style-type: none"> • Minimální spotřeba CPU • Snadné na pochopení • Snadná konfigurace
Nevýhody Proti (Cons)	<ul style="list-style-type: none"> • Jsou využívány systémové zdroje směrovače (cykly CPU, paměť a šířka pásma linky). • Pro konfiguraci, ověření a hledání chyb jsou potřeba hlubší znalosti na straně administrátora. 	<ul style="list-style-type: none"> • Časově náročná konfigurace i údržba • Snadný vznik chyby při nastavování, zvlášť u velkých sítí • Administrátor musí zasadovat při změně topologie a měnit obsah směrovací tabulky • Obtížně se rozšiřuje, aktualizace může být těžkopádná • Pro implementaci vyžaduje kompletní znalost celé sítě.

Klasifikace směrovacích protokolů

Evoluce směrovacích protokolů:

- EGP – 1982

- IGRP – 1985
- RIPv1 – 1988
- IS-IS – 1990
- OSPFv2 – 1991
- EIGRP – 1992
- RIPv2 – 1994
- BGP – 1995
- RIPng – 1997
- BGPv6 & OSPFv3 – 1999
- IS-ISv6 – 2000

Klasifikace dynamických směrovacích protokolů:

Vnitřní/vnější protokol		Vnitřní (<i>Interior Gateway Protocol, IGP</i>)				Vnější (<i>Exterior, EGP</i>)
Algoritmus		Vektor vzdálenosti (<i>Distance Vector</i>)		Stav linky (<i>Link State</i>)		Vektor cesty (<i>Path Vector</i>)
IPv4	Třídní	RIPv1	IGRP	-	-	EGP
	Bez-třídní	RIPv2	EIGRP	OSPFv2	IS-IS	BGPv4
IPv6		RIPng	EIGRP pro IPv6	OSPFv3	IS-IS pro IPv6	BGPv4 pro IPv6

V tomto kurzu se budeme zabývat pouze **zvýrazněnými** směrovacími protokoly. (IS-IS a BGP bude až v kurzu CCNP.)

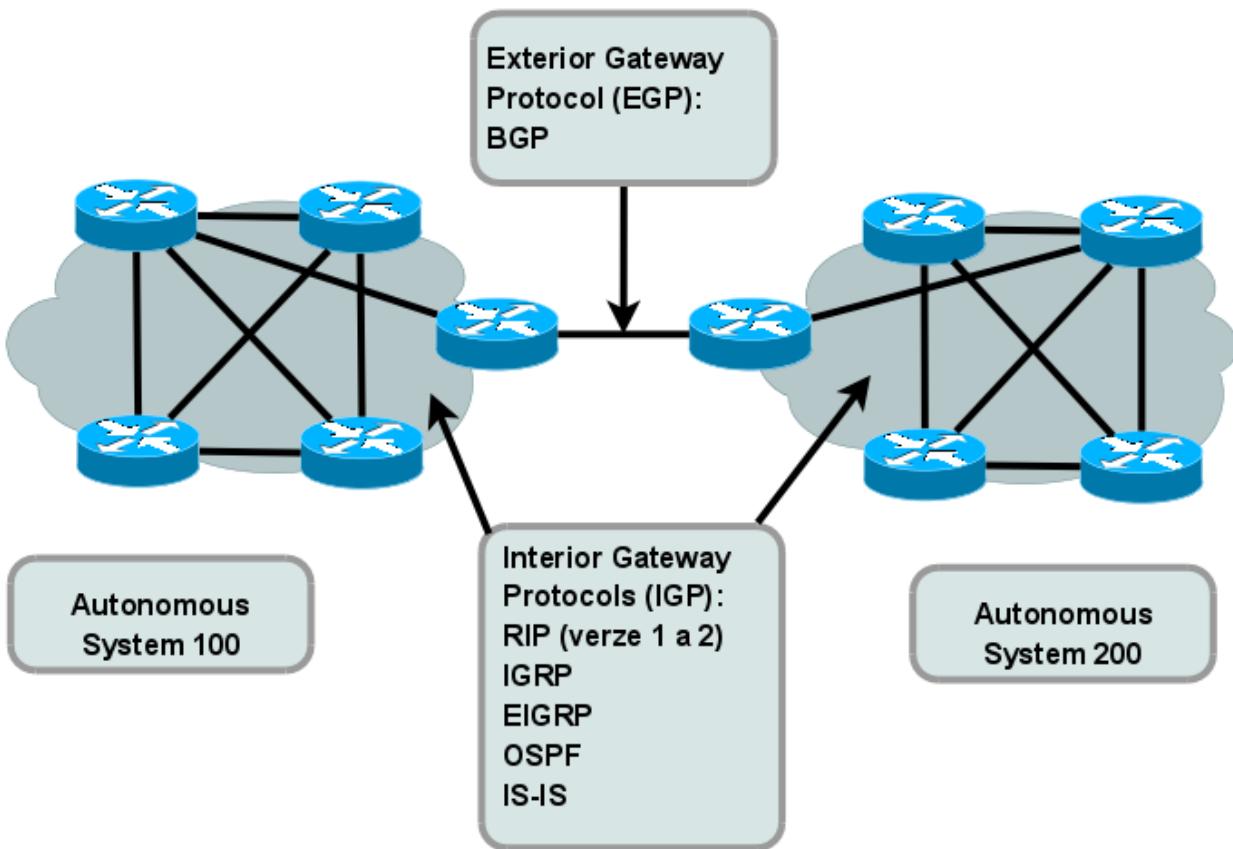
Rozdělení protokolů

Vnitřní a vnější směrovací protokol

- vnitřní směrovací protokoly - (*Interior Gateway Protocols, IGP*) se používají pro směrování uvnitř jednoho autonomního systému¹⁶
- vnější směrovací protokoly - (*Exterior Gateway Protocols, EGP*) se používají pro směrování mezi různými autonomními systémy (používají algoritmus vektor cesty).

¹⁶ Autonomní systém (AS, Autonomous System) (= směrovací doména, *routing domain*) je oblast s jednou směrovací politikou – jednou správou směrování. (Například jeden ISP, jedna firma.)

Vnitřní versus vnější směrovací protokoly (IGP vs. EGP Routing Protocols)



Směrovací algoritmy u vnitřních směrovacích protokolů

Účel směrovacího algoritmu:

1. vysílání a příjem směrovacích aktualizací,
2. výpočet nejlepší cesty a její umístění do směrovací tabulky,
3. detekce změn topologie a reakce na tyto změny.

U vnitřních směrovacích protokolů se používají dva směrovací algoritmy: vektor vzdálenosti a stav linky.

- **vektor vzdálenosti** (*distance vector*) – vektor vzdálenosti znamená že směry (cesty) jsou inzerovány jako vektory vzdálenosti a směr. Vzdálenost je definována termíny **metrika** (*metrics*) (například počet skoků) a **směr** (*direction*) (což jednoduše je následující směrovač nebo odchozí rozhraní tohoto směrovače). Protokoly typu vektor vzdálenosti obvykle používají pro určení nejlepší cesty **Bellman-Fordův algoritmus**. Některé protokoly typu vektor vzdálenosti periodicky posílají kompletní směrovací tabulky na všechny připojené sousedy. Ve velkých sítích mohou být takovéto aktualizace enormně velké a budou přičinou významné části síťového provozu. Ačkoliv algoritmus Bellman-Ford umožňuje získat dostatek informací o topologii sítě, algoritmus směrovači neumožňuje znalost přesné topologie sítě.

Směrovač zná pouze informace potřebné pro směrování, které získal od svého souseda. Vyjádřeno analogií: směrovače používají vektor vzdálenosti jako automobilista používá silniční rozcestníky podél své cesty do konečného cíle. Jedinou informací, kterou směrovač ví o vzdálené síti, je vzdálenost neboli metrika do cílové sítě a který směr neboli odchozí rozhraní použít k jejímu dosažení. Nemá mapu topologie sítě.

- **stav linky (link state)** – na rozdíl od činnosti protokolů vektor vzdálenosti může směrovač s nastaveným protokolem typu stavu linky vytvořit „úplný přehled“ nebo topologii sítě shromažďováním informací ze všech ostatních směrovačů. Jako pokračování naší analogie k rozcestníkům použitým u vektoru vzdálenosti mají směrovací mapy stavu linky kompletní mapu topologie sítě. Rozcestníky potom nejsou potřeba, protože všechny směrovače používají identickou „mapu“ celé sítě. Směrovač s protokolem stavu linky používá informace stavu linky k vytvoření topologické mapy a k výběru nejlepší cesty do všech cílových sítí v celé topologii (stromu). V některých směrovacích protokolech vektoru vzdálenosti, směrovače posílají periodické aktualizace svým sousedům. Protokoly stavu linky nepoužívají periodické aktualizace. Potom, co je síť zkonzvergovaná, jsou aktualizace stavu linky posílány pouze při změně v topologii. Též nazýváno **Dijkstrův algoritmus**¹⁷.

Kdy je vhodné použít ten který typ vnitřního směrovacího protokolu:

Vektor vzdálenosti	Stav linky
<ul style="list-style-type: none"> ● Síť je jednoduchá a plochá (<i>flat</i>) a nevyžaduje zvláštní hierarchickou strukturu ● Administrátor nemá dostatek znalostí o konfiguraci a odstraňování problémů při směrování proto, aby mohl použít algoritmus stavu linky. ● Pokud jsou implementovány zvláštní typy sítí, jako je například topologie s jedním centrálním zařízením (<i>hub-and-spoke</i>). ● Doba konvergence v síti je dlouhá (= doba synchronizace směrovacích tabulek tak, aby si vzájemně odpovídaly, byly ve vzájemně konzistentním stavu). 	<ul style="list-style-type: none"> ● Hierarchický návrh struktury sítě, obvykle u rozsáhlých sítí. ● Administrátor musí mít dobré znalosti jak implementovat směrovací protokol stavu linky. ● Rychlá konvergence.

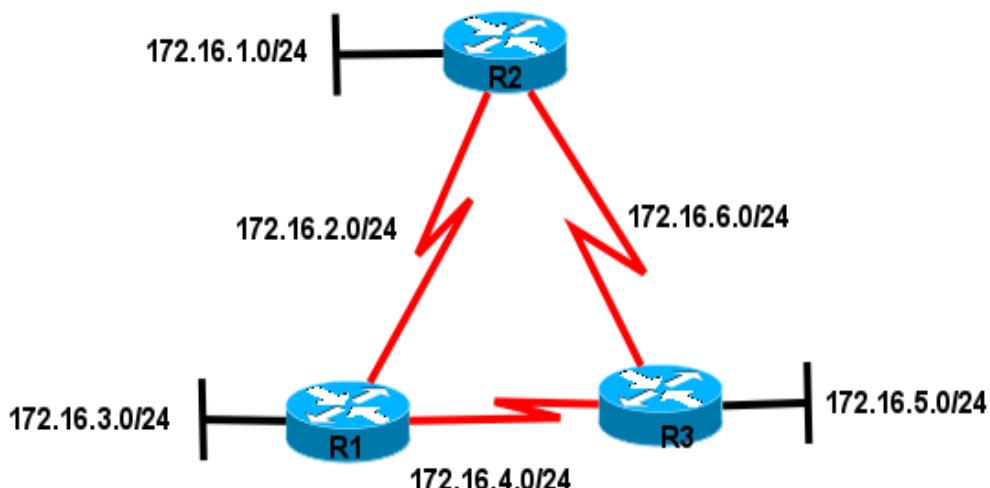
Směrování třídní versus beztřídní

- **třídní (classful)** – v celé topologii sítě jsou stejné masky podsítě. První směrovací protokoly byly použité v době, kdy byly sítě pouze v plných třídách (masky byly implicitně určené dle prvního oktetu IP adresy) a nebylo tedy nutné přenášet ve směrovacích aktualizacích masku podsítě. I přesto lze třídní směrování použít i v současnosti: Pokud je podsíťována síť v plné třídě a používá jednu stejnou masku podsítě (= *CIDR, Classless Inter-Domain Routing*). (Bztřídní směrování nepodporuje VLSM.) Sítě nesmí být nesouvislé (*discontiguous networks*), **sítě musí být souvislé (contiguous networks)**.

17 Popis algoritmů Bellmann-Fordova respektive Dijkstrova najdete též na webové stránce www.algoritmy.net.

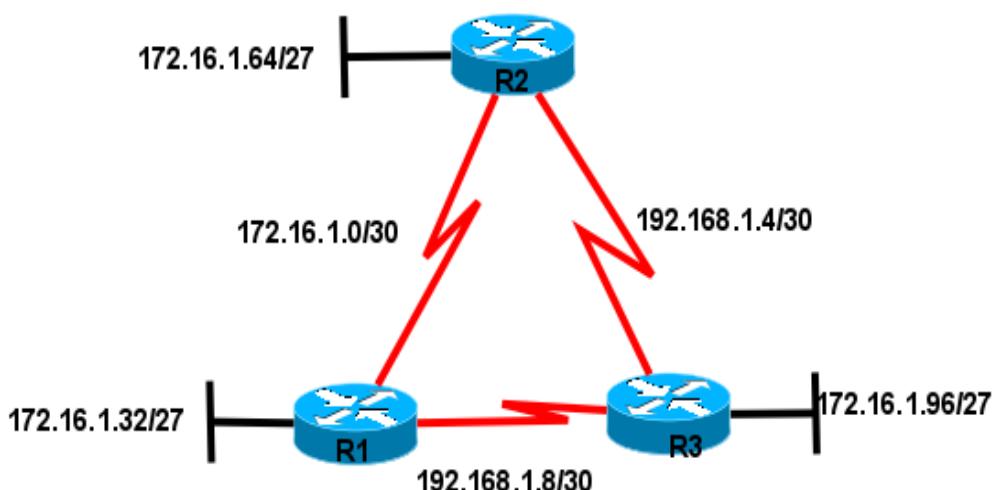
- => **tranzitní síť**¹⁸ může být sice podsítována (stejnou maskou, tj. adresní schéma typu jeden CIDR), ale musí ležet vzhledem k implicitní masce příslušné třídy v jedné síti.
- => **netranzitní síť** musí být buď třídní nebo ve stejném CIDR jako tranzitní síť.
- **beztrídní (classless)** – v topologii sítě může být více různých masek podsítě (= podporuje VLSM, *Variable Length Subnet Masking*). Ve směrovacích aktualizacích jsou masky podsítě. Sítě mohou být nesouvislé.

Třídní versus beztrídní směrování (Classful vs. Classless Routing)



Třídní: maska podsítě je stejná v topologii

Classful: Subnet mask is the same throughout the topology



Beztrídní: může být více různých masek v topologii

Classless: Subnet masks can vary in the topology

¹⁸ Tranzitní síť je síť, která pokračuje na dalším směrovači (je mezi dvěma směrovači). Netranzitní síť (*stub network*) je přístupná pouze přes jednu cestu (typicky lokální síť s jedním připojením k ISP) (nebo obsahuje pouze samá koncová zařízení).

Konvergence

Konvergence (*convergence*) – směrovací tabulky na všech směrovačích jsou konzistentní (*state of consistency*). Sítě je zkonvergovaná pokud všechny směrovače mají úplné a přesné informace o celé síti (směrovací tabulky jsou vzájemně konzistentní). Doba konvergence je doba, kterou směrovačům zabere výměna informací, výpočet nejlepších cest a aktualizace jejich vlastních směrovacích tabulek. Dokud není síť zkonvergovaná, není síť úplně funkční, proto se vyžaduje, aby doba konvergence byla co nejkratší.

Konvergence je obojí – spolupracující i nezávislá. Směrovače jednak sdílejí informace s každými jinými směrovači, ale zároveň musí samostatně počítat dopady změn topologie na jejich vlastní cesty (*routes*). Charakteristiky konvergence zahrnují: rychlosť propagace směrovacích informací a výpočet optimálních cest.

- Pomalejší konvergence – RIP, IGRP
- Rychlejší konvergence – EIGRP, OSPF

Metriky cest

Metrika je číselné vyjádření kvality, ceny (*cost*) cesty. Pokud je více cest do jedné cílové sítě, vybírá se cesta s nejmenší metrikou – nejlevnější cesta.

Metriky se u různých směrovacích protokolů počítají různým způsobem:

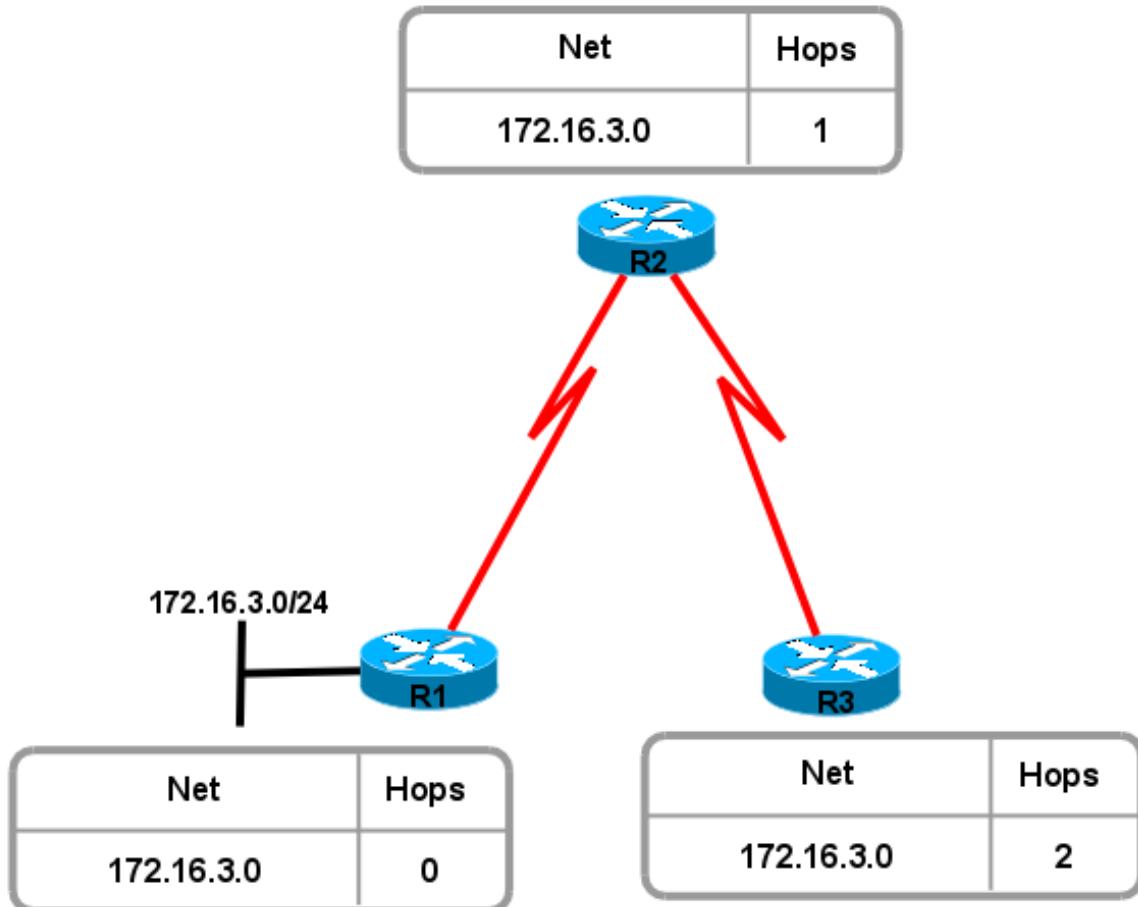
- **Počet skoků** (*Hop count*) – jednoduchá metrika, která počítá počet směrovačů přes které musí paket cestovat do cílové sítě
- **Digitální přenosová rychlosť, přenosová kapacita, šířka pásma** (*Bandwidth*) – při výběru cesty se preferuje linka s větší přenosovou rychlosťí
- **Zatížení** (*Load*) – bere v úvahu vytížení dané linky sítiovým provozem
- **Zpoždění** (*Delay*) – bere v úvahu dobu, kterou paket potřebuje ke své cestě přes síť
- **Spolehlivost** (*Reliability*) – vyhodnocuje pravděpodobnost výskytu chyby na lince, vypočteno z počtu chyb rozhraní nebo předchozích selhání linky
- **Cena** (*Cost*) – hodnota určená buď IOS nebo administrátorem vyznačující preferování dané cesty.

Metriky každého z následujících směrovacích protokolů jsou:

- **RIP**: počet skoků (*Hop count*) – jako nejlepší cesta je vybrán směr s nejnižším počtem skoků.
- **IGRP a EIGRP**: Přenosová rychlosť (přenosová kapacita), zpoždění, spolehlivost a zatížení (*Bandwidth, Delay, Reliability, and Load*) – jako nejlepší cesta je vybrán směr s nejmenší hodnotou složené metriky vypočtené z více různých parametrů. Implicitně je pro výpočet použitá pouze rychlosť a zpoždění.
- **IS-IS a OSPF**: Cena (*Cost*) - jako nejlepší cesta je vybrán směr s nejmenší cenou. Implementace OSPF od Cisco používá přenosovou rychlosť. IS-IS je diskutován v kurzu CCNP.

Metriky

Hops = přeskoky



Administrativní vzdálenosti protokolů

Administrativní vzdálenost (*Administrative Distance, AD*) slouží k odlišení metriky u cest (na jednom směrovači) získaných z různých směrovacích protokolů¹⁹. AD vyjadřuje kvalitu celého směrovacího protokolu, někdy se také říká důvěryhodnost cesty (*trustworthiness*). Pokud existuje více cest do jedné cílové sítě, vybírá se cesta, která má nejmenší administrativní vzdálenost, pokud je více cest se stejnou administrativní vzdáleností, vybírá se cesta s nejmenší metrikou.

Kódy směrovacích protokolů ve směrovací tabulce směrovače:

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP

i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area

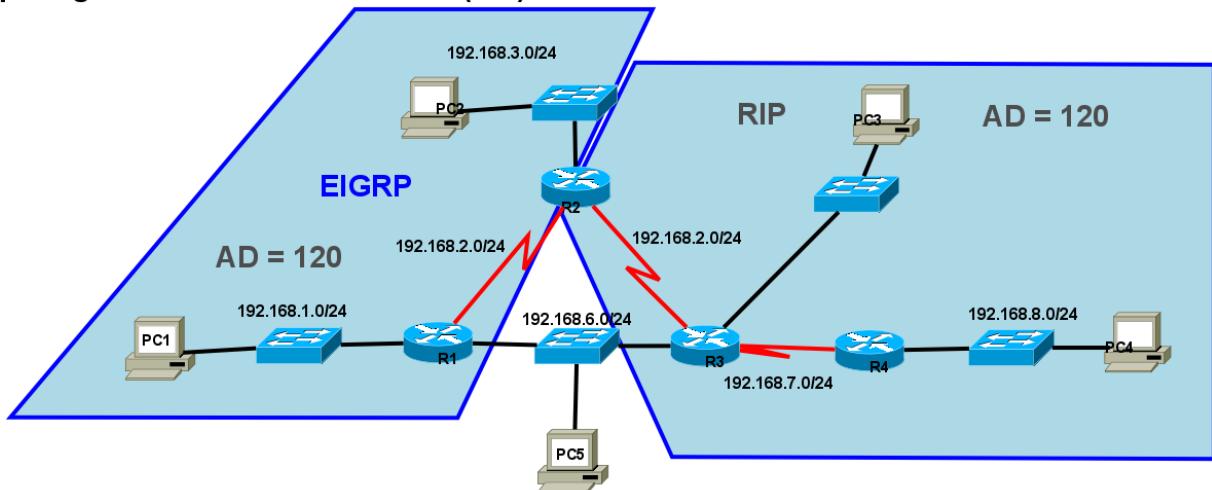
¹⁹ Na jednom směrovači může být spuštěno více různých směrovacích protokolů.

* - candidate default, U - per-user static route, o - ODR

P - periodic downloaded static route

Porovnání administrativních vzdáleností

Comparing Administrative Distances (AD)



R1 a R3 "nemluví" stejnými směrovacími protokoly do not "spe"

Administrativní vzdálenosti jednotlivých směrovacích protokolů

Směrovací protokol	Kód	Administrativní vzdálenost Administrative Distance
Přímo připojená síť (Directly connected)	C	0
Statická cesta (Static route)	S	1
EIGRP sumarizovaná cesta (summary route)		5
External BGP (Border Gateway Protocol)		20
Internal EIGRP	D	90
IGRP	I	100
OSPF (Open Shortest Path First)	O	110
IS-IS (Intermediate System to Intermediate System Routing Exchange Protocol)	i	115
RIP (Routing Information Protocol)	R	120
EGP (Exterior Gateway Protocol)	E	140
ODR (On-Demand Routing)		160
External EIGRP		170

Směrovací protokol	Kód	Administrativní vzdálenost Administrative Distance
Internal BGP		200
Neznámý protokol (Unknown)		255

Vybírá se vždy cesta s nejnižší administrativní vzdáleností (např. Vybere se cesta od OSPF než od RIP) a metrikou („cenou“ cesty).

Vyrovnávání zátěže

Pokud do jedné cílové sítě existuje více stejně nákladných cest (= se stejnou administrativní vzdáleností a stejnou metrikou) umí směrovací protokoly vyvažovat zátěž (*load balance*) (= cyklicky přepínat mezi jednotlivými cestami) mezi implicitně až čtyřmi takovýmito cestami. (V EIGRP lze vyvažovat/vyrovnávat zátěž až mezi 6-ti cestami a tyto cesty navíc nemusejí být stejně nákladné.)

Identifikace prvků směrovací tabulky

R 192.168.1.0/24 [120/1] via 172.16.3.253, 00:00:14, FastEthernet1/0

- směrovací protokol, kterým byla získána řádka
- IP adresa cílové sítě/maska (délka prefixu)
- administrativní vzdálenost/metrika
- IP adresa vstupního rozhraní dalšího směrovače (*next-hop router*)
- stáří aktualizace řádky hod:min:sec
- odchozí rozhraní směrovače pro danou cestu

Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Dvě výhody statického směrování proti dynamickému:
 - a) je bezpečnější, protože směrovače neinzerují cesty,
 - b) není režie (zátěž na směrovači) pro výměnu směrovacích informací.
- 2) Spárování popisků směrovacích protokolů:
 - a) RIP = (*Routing Information Protocol*) interní směrovací protokol typu vektor vzdálenosti
 - b) IGRP = (Interior Gateway Routing Protocol) proprietární vnitřní směrovací protokol Cisco
 - c) EIGRP = (Enhanced Interior Gateway Routing Protocol) vylepšený proprietární vnitřní směrovací protokol Cisco
 - d) OSPF = (Open Shortest Path First) vnitřní směrovací protokol typu stav linky

Cisco NetAcad: CCNA Exploration - Routing Protocols and Concepts – studijní materiál

- e) BGP = (Border Gateway Protocol) vnější směrovací protokol typu vektor cesty
- 3) Tvrzení popisující konvergenci sítě:
 - a) čas, který směrovače v síti potřebují pro aktualizaci svých směrovacích tabulek po změně topologie sítě
- 4) Který směrovací protokol má implicitně nejdůvěryhodnější administrativní vzdálenost? (rozuměj nejmenší hodnotu AD)
 - a) EIGRP (AD = 90) (OSPF = 110, RIP = 120, ...)
- 5) Pro kolik cest se stejnou metrikou mohou implicitně směrovací protokoly vyvažovat jejich zatížení (zátěž) (*load balancing*)?
 - a) 4
- 6) Kterým příkazem můžete vypsat administrativní vzdálenost směru (cesty)?
 - a) show ip route (vypíše obsah směrovací tabulky)
- 7) Kdy se objeví přímo připojené cesty ve směrovací tabulce?
 - a) Ihned jakmile jsou nastaveny adresy a jsou funkční na L3 OSI.
- 8) Směrovač se spuštěným protokolem RIPv2 má více různých cest do cílové sítě. Jak RIPv2 určí nejlepší cestu?
 - a) Podle nejmenší metriky
- 9) Zopakujte si administrativní vzdálenosti pro základní směrovací protokoly.
- 10) Základní rozdíly mezi třídním a beztřídním směrováním
 - a) třídní: neposílají v svých aktualizacích masku podsítě, nepodporují nesouvislé sítě, RIP-v1 a IGRP.
 - b) beztřídní: ve svých aktualizacích posílají masku podsítě, podporují nesouvislé sítě, EIGRP, OSPF a BGP.

Kapitola 4 - Směrovací protokoly typu vektor vzdálenosti

V této kapitole se naučíme:

- Identifikovat charakteristiky směrovacích protokolů založených na algoritmu vektor vzdálenosti (*distance vector*)
- Popsat postup průzkumu sítě protokoly vektoru vzdálenosti s použitím protokolu RIP (*Routing Information Protocol*)
- Popsat proces údržby a aktualizace přesného obsahu směrovacích tabulek tak, jak ho provádí směrovací protokoly založené na vektoru vzdálenosti
- Popsat podmínky vedoucí ke vzniku směrovacích smyček a jejich dopady na výkon směrovače
- Určit typy směrovacích protokolů založené na algoritmu vektoru vzdálenosti, které se v současnosti používají.

Směrovací protokoly typu vektor vzdálenosti

Dynamické směrovací protokoly administrátorovi šetří čas nutný pro časově náročné a přesné konfigurování i údržbu statických cest. Například: dovedete si představit spotřebovaný čas potřebný pro nastavení statického směrování skupiny několika desítek různě vzájemně propojených směrovačů? Co se stane, když nějaká linka spadne? Jak zajistíte, aby byla dostupná náhradní linka? Pro takové velké sítě je nejobvyklejší volbou dynamické směrování.

Účel směrovacího algoritmu:

1. vysílání a příjem směrovacích aktualizací,
2. výpočet nejlepší cesty a její umístění do směrovací tabulky,
3. detekce změn topologie a reakce na tyto změny.

Protokoly typu vektor vzdálenosti (se směrovacím algoritmem vektor vzdálenosti) zahrnují tyto směrovací protokoly: RIP, IGRP, a EIGRP.

RIP

Routing Information Protocol (RIP²⁰) byl původně specifikován v RFC 1058 (pro verzi 1 - RIPv1). Má následující klíčové charakteristiky:

- Jako metrika pro výběr cest je použit počet skoků (*hop count*).
- Jestliže je počet skoků pro nějakou síť větší než 15, nelze RIP použít pro směrování do takové sítě.
- Směrovací aktualizace jsou implicitně všesměrové (*broadcast*) nebo skupinové (*multicast*) (pro verzi 2) každých 30 sekund.

20 Nejde tedy o zkratku pro „Odpočívej v pokoji“ - *Rest in Peace – latinsky Requiescat in pace*, jak tvrdí zlomyslníci.

IGRP

Interior Gateway Routing Protocol (IGRP) je proprietární protokol vyvinutý firmou Cisco. Má následující klíčové charakteristiky:

- Pro vytvoření kompozitní metriky (*composite metric*) jsou použity: přenosová kapacita (šířka pásma) (*bandwidth*), zpoždění (*delay*), zátěž (*load*) a spolehlivost (*reliability*).
- Směrovací aktualizace jsou implicitně všesměrové (*broadcast*) každých 90 sekund.
- IGRP je předchůdce protokolu EIGRP a je dnes již zastaralý.

EIGRP

Enhanced IGRP (EIGRP) je proprietární protokol vyvinutý firmou Cisco. Má následující klíčové charakteristiky:

- Může provádět vyvažování zátěže cest s různou cenou (*unequal cost load balancing*).
- Pro výpočet nejkratší cesty používá difuzní aktualizační algoritmus DUAL (*Diffusing Update Algorithm*).
- Nemá periodické aktualizace jako RIP a IGRP. Směrovací aktualizace jsou zasílány pouze při změnách topologie sítě.

Porovnání vlastností směrovacích protokolů

	Vektor vzdálenosti					Stav linky	
	RIPv1	RIPv2	IGRP	EIGRP	OSPF	IS-IS	
Rychlosť konvergencie	Pomalá	Pomalá	Pomalá	Rychlá	Rychlá	Rychlá	
Rozšíritelnosť – velikosť sítě	Malá	Malá	Malá	Velká	Velká	Velká	
Použití VLSM	Ne	Ano	Ne	Ano	Ano	Ano	
Využití systémových zdrojů	Nízké	Nízké	Nízké	Střední	Vysoké	Vysoké	
Implementace a údržba	Jednoduchá	Jednoduchá	Jednoduchá	Komplikovaná	Komplikovaná	Komplikovaná	

Význam vektoru vzdálenosti

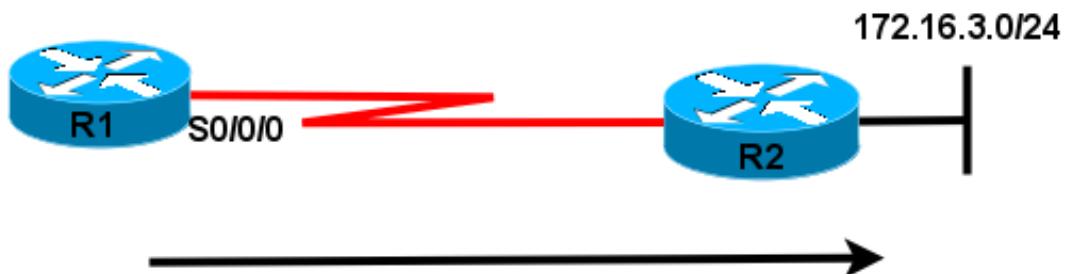
Jak už implikuje samotný název, vektor vzdálenosti znamená, že cesty (směry) jsou inzerovány jako vzdálenost a směr. Vzdálenost je definována termínem metrika (jako je počet skoků) a směr je jednoduše směrovač dalšího skoku nebo odchozí rozhraní.

Směrovač používající vektor vzdálenosti nemá vědomost o celé cestě do cílové sítě. Místo toho směrovač zná pouze:

- **směr** neboli rozhraní, kterým by měl být paket přeposlán a
- **vzdálenost** neboli jak je do cílové sítě daleko.

Význam vektoru vzdálenosti (The Meaning of Distance Vector)

Vzdálenost = jak je daleko
Distance = How Far



Vektor = směr

Vector = Direction

Pro R1 je síť 172.16.3.0/24 vzdálená jeden přeskok (hop) (distance).
A může být dosažena přes R2 (vektor = směr)

Funkce směrovacích protokolů typu vektor vzdálenosti

Některé směrovací protokoly vektoru vzdálenosti vyžadují, aby směrovač periodicky všeobecně posílal celou směrovací tabulku všem svým sousedům. Tato metoda je neefektivní, protože aktualizace pro svoji činnost konzumují nejen šířku pásma, ale i systémové zdroje směrovače.

Protokoly vektor vzdálenosti mají společné určité charakteristiky:

- **Periodické aktualizace (Periodic Updates)** jsou vysílány v pravidelných intervalech (30 sekund u RIP, 90 sekund u IGRP) všem sousedům. Bez ohledu na to, že se topologie nemění třeba několik dnů.
- **Sousedí (Neighbors)** jsou směrovače, které sdílejí linku a mají nastavený stejný směrovací protokol. Směrovač je si vědom pouze síťové adresy svého vlastního rozhraní a adresy vzdálené sítě, kterou může dosáhnout přes svého souseda. **Směrovač nemá žádnou vědomost o topologii sítě.**
- **Všeobecné aktualizace (Broadcast Updates)** jsou vysílány na adresu 255.255.255.255.

Sousedící směrovače s nastaveným stejným směrovacím protokolem aktualizaci provedou. Všechna ostatní zařízení aktualizaci také zpracují až do L3 před tím, než ji zahodí. Některé protokoly vektoru vzdálenosti používají skupinové adresy místo všesměrových adres.

- **Aktualizace obsahují celé směrovací tabulky** (kromě výjimek, které budou diskutovány později) a jsou vysílány periodicky na všechny sousedy. Sousedí musí zpracovat celou aktualizaci, aby našli patřičnou informaci a zahodili zbytek. Některé směrovací protokoly (jako EIGRP) neposílají periodické aktualizace.

Účel algoritmu směrovacího protokolu

Jádrem protokolu typu vektor vzdálenosti je algoritmus. Algoritmus je použit pro výpočet nejlepší cesty a poté co je tato informace vyslána sousedům.

Algoritmus je postup pro dosažení určitého konkrétního cíle (úkolu), začíná v daném počátečním stavu a je ukončen v definovaném koncovém stavu. Různé směrovací protokoly používají různé algoritmy pro instalaci směrů do směrovacích tabulek, vysílání aktualizací sousedům a určení cesty.

Algoritmus použitý pro směrovací protokoly definuje následující procesy:

- mechanizmus pro vysílání a příjem směrovacích informací,
- mechanizmus pro výpočet nejlepší cesty a instalaci tohoto směru (cesty) do směrovací tabulky,
- mechanizmus pro detekci změn topologie a reakce na tyto změny.

Charakteristiky směrovacích protokolů

Směrovací protokoly mohou být porovnávány na základě následujících charakteristik:

- **Doba potřebná pro konvergenci** (*Time to Convergence*) – definuje jak rychle směrovače v síti sdílejí směrovací informace a dosáhnou stavu konzistence znalostí. Čím rychlejší konvergence, tím preferovanější směrovací protokol. Když nejsou (z důvodu pomalé konvergence v měnící se síti) aktualizované nekonzistentní směrovací tabulky, mohou se vyskytnout **směrovací smyčky**.
- **Rozšiřitelnost** (škálovatelnost, *Scalability*) – definuje pro jak velkou síť může být ten který směrovací protokol použit. Čím větší ta síť je, tím více rozšiřitelný směrovací protokol musí být.
- **Beztrídnost** (*Classless*) (použití VLSM) **nebo plnotrídno**st (*Classful*) – Beztrídní směrovací protokoly ve svých aktualizacích obsahují masku podsítě. Tato funkce podporuje použití VLSM (*Variable Length Subnet Masking*) a lepší summarizaci směru. Třídní směrovací protokoly nemají v aktualizaci směrovací masku a nepodporují VLSM.
- **Spotřeba zdrojů** (*Resource Usage*) – zahrnuje požadavky směrovacího protokolu na systémové zdroje jako jsou velikost operační paměti, využití procesoru, využití šířky pásma linky. Vyšší požadavky na systémové zdroje si vynucují použití silnějšího HW.
- **Implementace a údržba** (*Implementation and Maintenance*) – požadavky na znalosti administrátora potřebné pro implementaci a údržbu použitého směrovacího protokolu.

Výhody a nevýhody směrovacích protokolů typu vektor vzdálenosti jsou uvedeny v následující

tabulce:

Výhody směrovacích protokolů typu vektor vzdálenosti	Nevýhody směrovacích protokolů typu vektor vzdálenosti
Jednoduchá implementace a údržba – administrátor nepotřebuje příliš hluboké znalosti.	Pomalá konvergence – použití periodických aktualizací způsobuje pomalou konvergenci. I když jsou použité pokročilé metody jako je automaticky spouštěná aktualizace, jsou tyto protokoly pomalejší než protokoly typu stav linky.
Nízké požadavky na zdroje – nevyžaduje výkonnou CPU a velkou paměť. Požadavky však vzrůstají se zvětšující se síti.	Malá rozšířitelnost – pomalá konvergence ve velkých sítích způsobuje dlouhou nekonzistentnost směrovacích tabulek.
	Směrovací smyčky – ty se mohou objevit, když nejsou nekonzistentní směrovací tabulky aktualizovány kvůli pomalé konvergenci.

Objevování sítí

Studený start

Když se směrovač zapne (*cold start*), neví nic o síťové topologii ani neví nic o tom, že je na druhé straně linky nějaké síťové zařízení. Má pouze informace uložené v počáteční konfiguraci uložené v NVRAM. Jakmile směrovač úspěšně zavede operační systém použije uloženou konfiguraci. Jestliže jsou korektně nastaveny IP adresy, směrovač detekuje svoje vlastní přímo připojené sítě a příslušné masky. Tyto informace jsou přidány do jeho směrovací tabulky. To vše proběhne ještě před jakoukoliv výměnou směrovacích informací pomocí dynamických směrovacích protokolů.

Počáteční výměna směrovacích informací

Jestliže je nakonfigurován směrovací protokol, směrovače si začnou vyměňovat směrovací aktualizace. Na počátku tyto aktualizace obsahují pouze informace o jejich přímo připojených sítích. Potom co přijme aktualizaci v směrovač vyhledá nové informace. Všechny směry do sítí, které aktuálně nejsou ve směrovací tabulce jsou do ní přidány.

Další aktualizace

V té chvíli mají směrovače vědomost o svých vlastních přímo připojených sítích a o připojených sítích jejich bezprostředních sousedů. Pokračujíce v konvergenci si směrovače vymění **další kolo (next round) periodických aktualizací**. Každý směrovač opět ověří aktualizace a vyhledá nové informace a nové směry do konkrétních sítí přidá do směrovací tabulky. Směrovací tabulky se tak postupně mění, dokud není celá síť zkonvergovala, nebo pokud nastane změna topologie.

Konvergence

Doba konvergence je přímo úměrná velikosti sítě. Rychlosť dosažení zkonzervované sítě závisí na:

- rychlosti propagace změn topologie v aktualizacích pro sousedy daného směrovače,

- rychlosti výpočtu nejlepších směrů (cest) na základě získaných informací z aktualizací.

Síť není plně funkční dokud není zkonzvergovaná, proto administrátoři preferují směrovací protokoly s krátkou dobou konvergence.

Údržba směrovací tabulky – periodické aktualizace

Periodické aktualizace u RIP a IGRP

Mnoho protokolů typu vektor vzdálenosti používá **periodické aktualizace** (příkladem je RIP a IGRP).

Například RIP zasílá aktualizace každých 30 sekund jako broadcast (255.255.255.255) bez ohledu na to, zda došlo či nedošlo ke změně topologie. Tento interval je **aktualizační časovač** (*Update Timer*), který také slouží k sledování stáří informací ve směrovací tabulce. Po každém přijetí aktualizace je obnoveno nulové stáří (00:00:00) řádek ve směrovací tabulce. Tyto řádky (směrovací informace) se mohou změnit, když dojde ke změně topologie. Ke změně topologie může dojít například z následujících důvodů:

- selhání linky,
- připojení nové linky,
- selhání směrovače,
- změna parametrů linky.

Časovače u protokolu RIP

K **aktualizačnímu časovači** (*Update Timer*) = perIODA pravidelných aktualizací (30 sekund), zavádí ještě IOS další tři časovače:

- **časovač neplatné cesty** (*Invalid Timer*) – pokud nepřijde aktualizace existující cesty do 180 sekund (implicitně), je cesta označena jako neplatná nastavením nekonečné metriky (na 16). Cesta zůstává jako neplatná ve směrovací tabulce dokud nevyprší **vyprazdňovací časovač**.
- **vyprazdňovací časovač** (*Flush Timer*) – implicitně je nastavený na 240 sekund, o 60 sekund delší než časovač neplatné cesty. Když tento časovač vyprší, je cesta smazána ze směrovací tabulky.
- **zadržovací časovač** (*Holddown Timer*) – tento časovač stabilizuje směrovací informace a **předchází vzniku směrovacích smyček** (*routing loop*). Během doby, kdy topologie konverguje, neinzeruje nové informace vzniklé po změně topologie. Jakmile je cesta (*route*, směr) označena jako nedostupná (*unreachable*), musí zůstat zadržená (*holddown*) dostatečně dlouho, aby se všechny ostatní směrovače byly schopné dozvědět o nedostupné síti. Zadržovací časovač musí být nastaven o chvíli (o několik sekund) **delší než je celková doba konvergence celé sítě** (implicitní hodnota je 180 sekund).

Na směrování je třeba pohlížet jako na dynamický systém měnící se v čase. Smyslem časovačů je tlumení tohoto dynamického systému. Vždy je lepší mít směr do cílové sítě označený jako neplatný, než nevědět vůbec nic a zbytečně směrovat přes implicitní cestu, pokud je implicitní cesta na směrovači nastavena.

- zjištění stáří záznamu ve směrovací tabulce:
 - show ip route
- zjištění všech spuštěných směrovacích protokolů, jejich konfigurace a poslední aktualizace:
 - show ip protocols

Svázané aktualizace: EIGRP

Na rozdíl od ostatních protokolů vektoru vzdálenosti, EIGRP nevysílá periodické aktualizace. Místo toho EIGRP vysílá svázané aktualizace (*bounded update*) o směru, ve kterém se změnila cesta nebo metrika. Když se objeví nový směr nebo když má být směr smazán, EIGRP posílá aktualizaci pouze o této síti a nikoliv celou směrovací tabulkou. Tato informace je vyslána pouze těm směrovačům, které ji potřebují.

Aktualizace EIGRP tedy jsou:

- **neperiodické**, protože nejsou vysíány pravidelně,
- **částečné** (*partial update*), protože se posílají informace pouze o změnách topologie, které mají vliv na směrování,
- **svázané** (*bounded*), protože se propagují částečné aktualizace automaticky svázané s určitými směrovači (s vytvořeným vztahem sousedství), takže jsou aktualizovány pouze ty směrovače, které ty informace potřebují.

Poznámka: podrobněji budou funkce EIGRP zmíněny v kapitole 9.

Událostí spouštěné aktualizace: RIPv1 i RIPv2

Aby se urychlila konvergence po změně topologie, používá RIP změnou topologie vyvolané **automaticky spouštěné aktualizace** (*triggered updates*). Tyto aktualizace jsou vyslány bezprostředně po změně směrovací tabulky a nečekají na vypršení aktualizačního směrovače. Směrovač detekující změnu ihned vyšle aktualizační zprávu na své sousedící směrovače (*adjacent routers*). Přijímající směrovače postupně generují spouštěné aktualizace, které informují zase jejich sousedy o této změně.

Spouštěné aktualizace se vysílají po výskytu jedné z následujících událostí:

- změna stavu rozhraní (zapnuto nebo vypnuto)
- směrovač vstoupil (nebo vystoupil) z/do „nedosažitelného“ stavu ("unreachable" state)
- do směrovací tabulky byl instalován (nový) směr (cesta)

Použití pouze spouštěných aktualizací by mohlo být dostačující, pokud by zde byla záruka, že vlna aktualizací dosáhne každý příslušný směrovač ihned. Přesto jsou se spouštěnými aktualizacemi dva problémy:

1. Pakety obsahující aktualizace mohou být na některých linkách v síti zahozeny nebo poškozeny.
2. Spouštěné aktualizace nejsou okamžité. Je tedy možné, že směrovač, který ještě nepřijal

spouštěnou aktualizaci vyšle pravidelnou periodickou aktualizaci právě ve špatný čas, což způsobí, že na souseda, který právě přijímá spouštěnou aktualizaci bude znovu vložen v této chvíli už špatný směr.

Náhodné kolísání (Random Jitter) aktualizačního časovače

Problémy se synchronizovanými aktualizacemi

Když více směrovačů přenáší směrovací aktualizace najednou ve stejnou chvíli (například v segmentu sítě LAN s vícenásobným přístupem a rozbočovačem), mohou pakety kolidovat a způsobit zpoždění nebo spotřebovávat příliš velkou šířku pásma. (Poznámka: kolize se vyskytnou pouze v případě použití rozbočovače a nikoliv přepínače.)

Vysílání aktualizací ve stejný čas je známé jako **synchronizace aktualizací**. Synchronizace může způsobovat problémy spolu s protokoly typu vektor vzdálenosti, protože ty používají periodické aktualizace. Jak začne být několik směrovačů synchronizováno, objeví se v síti více a více kolizí aktualizací a prodlužující se zpoždění. Na počátku aktualizace nebudou synchronizovány, ale postupně se časovače v síti sesynchronizují.

Řešení

Aby se zabránilo synchronizaci aktualizací mezi směrovači, Cisco IOS používá náhodnou proměnnou nazývanou RIP_JITTER, která odečítá určitý časový úsek od aktualizačního intervalu na každém směrovači v síti. Toto náhodné kolísání (*random jitter*) se mění mezi 0 až 15 procenty doby periody aktualizací. Tímto způsobem potom implicitně 30-ti sekundový aktualizační časovač náhodně kolísá mezi 25 až 30 sekundami.

Směrovací smyčka

Směrovací smyčka (*Routing Loop*) je stav ve kterém je paket trvale přenášen uvnitř skupiny na sebe navazujících směrovačů aniž kdy dosáhne zamýšlené cílové sítě. Směrovací smyčka se může vyskytnout když dva nebo více směrovačů mají ve své směrovací tabulce zdánlivě platnou cestu do ve skutečnosti nedosažitelného cíle. Důsledkem je nedoručování některých paketů a zbytečná zátěž sítě.

Čtyři možné způsoby vzniku směrovací smyčky:

1. nesprávně nastavené statické cesty
2. nesprávně nastavený proces redistribuce cest (na hraničních směrovačích mezi různými směrovacími protokoly) (podrobněji se probírá až v kurzu CCNP)
3. nekonzistentní směrovací tabulky při pomalé konvergenci v měnící se síti
4. nesprávně nastavené nebo nainstalované vyřazené cesty (po výpadku linky)

Jaké jsou dopady vzniku směrovacích smyček? Pokles výkonosti sítě nebo přímo její odstávka.

Směrovací smyčky mohou způsobit následující potíže:

- Spotřeba přenosové kapacity linky pakety ve smyčce.

- Spotřeba strojového času směrovače pakety ve smyčce.
- Směrovače jsou zavaleny pakety ve smyčce, což má negativní dopad na konvergenci sítě.
- Může dojít ke ztrátě směrovacích aktualizací či k jejich nesprávnému načasování. To vede ke vzniku dalších směrovacích smyček a situace se stává ještě horší.
- Pakety se mohou ztrácat v „černých dírách“.

Jak vidíte, směrovací smyčky konzumují přenosovou kapacitu, systémové zdroje směrovačů a výsledkem je pomalá nebo dokonce nereagující síť.

Existují různé **mechanismy pro eliminaci směrovacích smyček**, například:

1. definování maximální metriky pro eliminaci počítání do nekonečna (*count to infinity*)
2. zadržovací časovače (*holddown timers*)
3. rozložený horizont (*split horizon rule*)
4. otrávení/znehodnocení cest (*route poisoning*) nebo otrávené/znehodnocené zpětné informace (*poison reverse*)
5. aktualizace vyvolané událostí (= změnou topologie, přijetím aktualizace) (*triggered updates*)

Počítání do nekonečna

Počítání do nekonečna nastane, když nesprávné směrovací aktualizace postupně zvětšuje hodnotu metriky do „nekonečna“ (*infinity*) pro síť, která už nadále není dostupná.

Nastavení maximální hodnoty metriky

Pro zastavení postupné inkrementace metriky, je **definováno „nekonečno“ maximální hodnotou metriky**. Například: RIP má „nekonečno“ definováno jako 16 přeskoků. Jakmile směrovače „dopocítají do nekonečna“, a dál už tuto hodnotu metriky neinkrementují, je příslušná trasa označena jako nedostupná.

Prevence směrovací smyčky pomocí zadržovacího časovače

Již dříve jsme se dozvěděli o automaticky, po změně topologie, spouštěných aktualizacích (*triggered update*) určených pro urychlení konvergence. Nezapomeňte, že kromě těchto změnou topologie vyvolaných aktualizací posílají směrovací protokoly typu vektor vzdálenosti také ještě i periodické aktualizace. Teď si představme, že určitá síť je nestabilní. Její rozhraní se rychle za sebou zapíná a opět vypíná.

Trasa je střídavě dostupná a nedostupná, vypadává, kolísá (*flapping*). Použití automaticky spouštěných aktualizací nutí směrovač reagovat příliš rychle a nevědomky tak vytvořit směrovací smyčku. Smyčka může být vytvořena periodickou aktualizací, kterou směrovač pošle během nestability. Zadržovací časovač (*holddown timer*) za těchto podmínek potlačuje vznik směrovací smyčky a zároveň předchází počítání do nekonečna.

Zadržovací časovače se používají k potlačení nesprávného nainstalování tras, které by mohly být vadné, z pravidelných aktualizací. Zadržovací časovač instruuje směrovač, aby po určité době zadržel všechny aktualizace, které by mohly ovlivňovat trasy. Jestliže je trasa identifikována jako vy-

pnutá nebo pravděpodobně vypnutá, jsou všechny informace o této trase obsahující stejný nebo horší stav ignorovány po předem určenou tlumící dobu. To znamená, že směrovač ponechá trasu označenou jako nedostupnou v tomto stavu po dostatečně dlouhou dobu, aby se v aktualizacích propagovaly co nejaktuálnější informace.

Zadržovací (tlumící) časovače pracují následujícím způsobem:

1. Směrovač přijme od souseda aktualizaci indikující, že síť, která byla předtím dostupná, už dále není dostupná.
2. Směrovač tuto síť označí jako pravděpodobně vypnutou a spustí zadržovací směrovač.
3. Jestliže je během tlumící doby pro tuto síť přijata z libovolného souseda aktualizace s lepší metrikou, je síť uvedena zpět a zadržovací časovač je vymazán.
4. Jestliže je během tlumící doby pro tuto síť přijata z libovolného souseda aktualizace se stejnou nebo horší metrikou, je aktualizace ignorována. A tak je umožněna delší doba pro šíření informace o změně.
5. Směrovače přitom stále posílají pakety do cílových sítí, které jsou označeny jako pravděpodobně vypnuté. To směrovači umožní překonat každý problém s přerušovanou konektivitou. Jestliže cílová síť skutečně není k dispozici a zároveň jsou pakety předávány, směruje se do „černé díry“ až dokud zadržovací časovač nevyprší.

Rozložený horizont

Pravidlo **rozložený horizont** (*split horizon*) předchází vzniku směrovacích smyček tak, že směrovač neinzeruje síť prostřednictvím toho rozhraní, ze kterého se o této síti dozvěděl (ze kterého přišla původní aktualizace).

Rozložený horizont s otrávenou zpětnou informací neboli otrávení cest

Otrávení/znehodnocení cest

Otrávení/znehodnocení cest (*route poisoning*) slouží k označení cesty jako nedostupné ve směrovací aktualizaci, která je zasílána na jiné směrovače. Jako nedostupná je interpretována cesta, která má nastavenou metriku větší než maximální možnou (= nekonečnou). Pro RIP je otrávena cesta s metrikou 16.

Rozložený horizont s otrávenou/znehodnocenou zpětnou informací

Pravidlo **rozložený horizont s otrávenou/znehodnocenou zpětnou informací** (*Split Horizon with Poison Reverse*) označuje, když vysílá směrovací aktualizace z určitého rozhraní, všechny sítě, které byly naučeny z tohoto rozhraní, jako nedostupné. Obecně je vždy lepší říci směrovači, že určité cesty (směry) jsou nedostupné a že je má ignorovat, než mu o nich neříci vůbec nic (a směrovač se potom může snažit použít implicitní cestu).

Životnost IP paketu

Životnost IP paketu (*Time to Live (TTL)*) je 8-bitové pole v záhlaví IP, které omezuje počet skoků (směrovačů), přes které může paket cestovat přes síť před tím, než je zahozen. Účelem pole TTL je

předcházet situaci, kdy by nedoručitelný paket cirkuloval v síti nekonečně dlouho. Hodnota TTL je v paketu nastavena zdrojovým zařízením paketu. **Na každém směrovači na cestě do cíle je TTL snížováno o jedničku.** Pokud TTL dosáhne na nějakém směrovači hodnotu nula ještě před svým doručením do cíle, paket je zahoden a tento směrovač zašle zpět zdrojovému zařízení chybovou zprávu v protokolu ICMP (*Internet Control Message Protocol*) o překročení životnosti (*TTL exceeded message*).

Porovnání směrovacích protokolů (typu vektor vzdálenosti):

	RIPv1	RIPv2	IGRP	EIGRP
Vektor vzdálenosti (<i>Distance Vector</i>)	Yes	Yes	Yes	Yes
VLSM	No	Yes	No	Yes
Autentizace zdroje aktualizací (<i>Authentication</i>)	No	Yes	No	Yes
Aktualizační časovač (<i>Update Timer</i>) (sec)	30	30	90	n/a
Časovač neplatné cesty (<i>Invalid Timer</i>) (sec)	180	180	270	n/a
Vyprazdňovací časovač (<i>Flush Timer</i>) (sec)	240	240	630	n/a
Zadržovací časovač (<i>Holddown Timer</i>) (sec)	180	180	280	n/a
Protokol/port (<i>Protocol/port</i>)	UDP 520	UDP 520	IP 9	IP 88
Administrativní vzdálenost (<i>Administrative Distance</i>)	120	120	100	90

Výpočet metriky u algoritmu typu vektor vzdálenosti

Pro jednu konkrétní cestu se vezme metrika inzerovaná v aktualizaci ze sousedního směrovače a k ní se **přičte** metrika cesty z rozhraní, na které aktualizace přišla, do směrovače, ze kterého aktualizace přišla. To znamená, že k metrice cesty do cílové sítě propagované ze sousedního směrovače se přičte metrika přilehlého segmentu fyzické sítě (linky), ze kterého tato propagace přišla. Tento algoritmus se také nazývá **Bellman-Ford**.

Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Uveďte čtyři pravdivá tvrzení o směrovacích protokolech typu vektor vzdálenosti:
 - a) jako metriku mohou používat počet skoků (RIP),
 - b) aktualizace rozesílají v pravidelných časových intervalech vše směrově (RIP),
 - c) EIGRP umí vyvažování zátěže i na různě nákladných cestách,
 - d) směrovač s RIP zasílá celou svoji směrovací tabulkou na všechny sousední směrovače.
- 2) Za jakých podmínek směrovací protokoly typu vektor vzdálenosti rozesílají svoje směrovací

- aktualizace? (uveďte tři)
- když se objeví změna topologie (volitelně nastavitelné automaticky spouštěné aktualizace, *triggered update*),
 - když vyprší aktualizační časovač (*update timer*) (pravidelné periodické aktualizace),
 - když přijde automaticky spuštěná aktualizace z jiného směrovače (dojde ke změně směrovací tabulky).
- 3) Dvě charakteristiky aktualizací z EIGRP:
- pouze automaticky spouštěné aktualizace (*triggered update*) při změně topologie (EIGRP nemá pravidelné aktualizace),
 - svázané aktualizace (*bounded update*) se zasaženými (*affected*) přilehlými směrovači dalšího přesoku (*next hop*)
- 4) Jaká funkce byla přidána do protokolu RIP, aby předcházela vzniku chyb při synchronizaci aktualizací (to znamená aktualizací probíhajících ve stejných časech paralelně)?
- RIP_JITTER (aktualizační časovače mají vloženou náhodnou časovou odchylku 0-15%, aby se předešlo synchronizaci aktualizací)
- 5) Časovače použité u RIP:
- invalid, flush, holddown a update (naopak není použit časovač *hello*, ten je u EIGRP a OSPF)
- 6) Co je pravda ohledně výhod použití protokolů typu vektor vzdálenosti:
- jednoduchá implementace a snadná konfigurace (to je také jejich jediná výhoda)
- 7) Jaký mechanismus lze použít pro předcházení vzniku směrovací smyčky typu počítání do nekonečna (*count to infinity loop*)?
- Zadržovací časovače (*holddown timers*).
- 8) Jak se jmenuje postup, kdy směrovací protokol typu vektor vzdálenosti předchází vzniku smyčky pomocí propagace směru s nastavenou nekonečnou metrikou?
- Otrávení/zneplatnění cest (*route poisonning*)
- 9) Které políčko v záhlaví paketu IP zabrání, aby paket necestoval v síti ve smyčce nekonečně dlouho?
- TTL
- 10) Spárování názvu a popisu jednotlivých metod pro předcházení vzniku směrovacích smyček:
- rozložený horizont = cesty zjištěné (naučené) z jednoho rozhraní nejsou z tohoto rozhraní propagovány
 - otrávení cest (otrávení zpětných informací) = cesty naučené z jednoho rozhraní jsou z tohoto rozhraní inzerovány zpět jako nedosažitelné (neplatné, s nekonečnou metrikou)
 - automaticky spouštěné aktualizace = změny topologie jsou ihned zasílány sousedícím směrovačům,

- d) zadržovací časovače = umožní časovou prodlevu, aby se informace o změně topologie mohly rozšířit po celé síti.

Kapitola 5 - Protokol RIP verze 1

V této kapitole se naučíme:

- Popsat funkce, charakteristiky a činnost protokolu RIPv1.
- Konfigurovat síťové zařízení s použitím RIPv1
- Ověřit správnou činnost RIPv1
- Popsat, jak RIPv1 provádí automatickou summarizaci
- Konfigurovat, ověřit a odstranit chyby propagace implicitní cesty v sítích směrovaných protokolem RIPv1
- Používat k řešení problémů souvisejících s RIPv1 doporučované techniky

RIPv1: třídní směrovací protokol typu vektor vzdálenost

Směrovací protokoly byly postupem času vyvinuty tak, aby uspokojily rostoucí požadavky po složitých sítích. První takový použitý směrovací protokol byl **Routing Information Protocol (RIP)**. RIP stále těší popularitě díky své jednoduchosti a široké podpoře.

Pochopení protokolu RIP je pro vaše studia sítí důležité z následujících dvou důvodů:

1. RIP je dnes stále ještě používaný. Můžete se setkat s implementací sítě, která je dostatečně velká, aby byl potřeba nějaký směrovací protokol, ale zároveň dostatečně jednoduchá, aby bylo použití RIP efektivní.
2. Obeznámenost se základními koncepty RIP vám pomůže porovnat RIP s jinými protokoly. Pochopení činnosti RIP a jeho implementace umožní snazší pochopení jiných směrovacích protokolů.

Tato kapitola se vztahuje na detaily první verze RIP, včetně trošky historie, vlastností, provozu, konfigurace, ověřování a řešení problémů. V průběhu této kapitoly, můžete používat aktivity Packet Traceru pro procvičení toho, co jste se naučili. Na konci této kapitoly jsou tři praktická laboratorní cvičení a aktivita Packet Traceru pro integraci dovedností, aby vám pomohly zařadit RIPv1 do rostoucí množiny vašich síťových znalostí a dovedností.

RIP vliv minulosti

RIP je nejstarší ze směrovacích protokolů typu vektor vzdálenosti. Přestože RIP postrádá propracovanost pokročilejších směrovacích protokolů jeho jednoduchost a pokračující široké využití je důkazem jeho životnosti. RIP není protokol "na odpis". Ve skutečnosti je nyní k dispozici forma RIP pro IPv6 tzv. RIPng (*Next Generation* = další generace).

Přehled historických souvislostí RIP

Vývoj síťových protokolů		Vývoj RIP
Počátek 70-tých let	Ranný vývoj TCP/IP	
Střed 70-tých let		Xerox PARC Universal Protocol (PUP)
Konec 70-tých let		Xerox Network System (XNS)
Počátek 80-tých let	Standardizován TCP/IP RFC 791, RFC 793	Berkeley Software Distribution (UNIX BSD 4.2)
1988		RFC 1058: RIP
1994		RFC 1723: RIPv2
1997		RFC 2080: RIPng

RIP se vyvinul z dřívějšího protokolu vyvinutého ve firmě Xerox, tzv. Gateway Information Protocol (GWINFO). S rozvojem společnosti Xerox Network System (XNS), se GWINFO vyvinul v RIP. Později získal popularitu, protože byl implementován v distribuci Berkeley Software Distribution (BSD) jako démon *routed* (anglicky se vyslovuje "route-dee", nikoli "rout-ed"). Množství dalších prodejců vytvořilo své vlastní mírně odlišné implementace RIP. Cítíc potřebu standardizace protokolu, napsal Charles Hedrick v roce 1988 RFC 1058, v němž dokumentoval stávající protokol a specifikoval některá vylepšení. Od té doby by RIP vylepšen s RIPv2 v roce 1994 a s RIPng v roce 1997.

Poznámka: První verze RIP je často nazývána RIPv1 pro odlišení od RIPv2. Nicméně, obě verze mají mnoho stejných vlastností. Při diskusi vlastností společných pro obě verze budeme odkazovat na RIP. Při diskusi jedinečných vlastností pro každou verzi budeme používat RIPv1 a RIPv2. RIPv2 je diskutován v pozdější kapitole.

Odkazy

RFC 1058: Routing Information Protocol, <http://www.ietf.org/rfc/rfc1058.txt>

Charakteristiky RIP

Jak je popsáno v kapitole 4, "Směrovací protokoly typu vektor vzdálenosti", má RIP tyto hlavní vlastnosti:

- RIP je směrovací protokol typu vektor vzdálenosti.
- RIP používá jako jeho jedinou metriku pro výběr cesty počet přeskoků.
- Inzerované trasy s počty skoků většími než 15 jsou nedosažitelné.
- Zprávy jsou všeobecně vysílány každých 30 sekund.

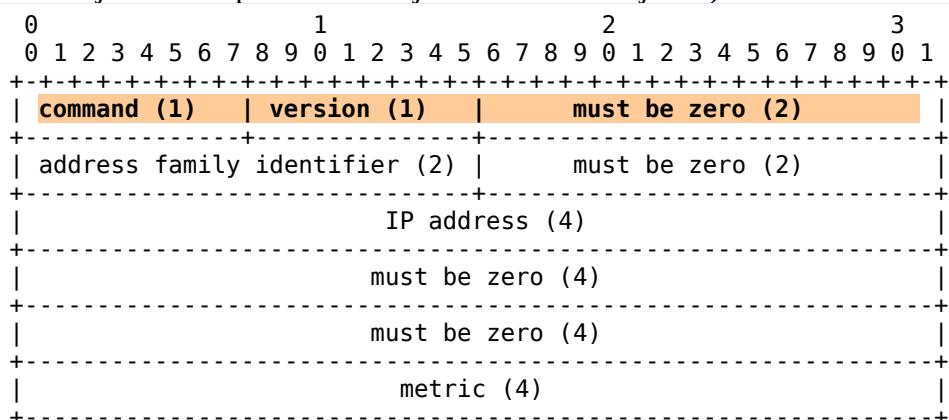
Datová část zprávy RIP je zapouzdřena do segmentu UDP, se zdrojovým i cílovým číslem portu nastaveným na 520. Než je zpráva rozeslána ze všech nakonfigurovaných rozhraní RIP jsou do záhlaví IP a linkové přidány všeobecné cílové adresy.

Zapouzdření zprávy protokolu RIPv1

Záhlaví linkové vrstvy	Záhlaví paketu IP	Záhlaví UDP Datagramu	Zpráva RIP (512 bajtů: až 25 tras)
Rámec linkové vrstvy			
Zdrojová MAC adresa = adresa vysílajícího rozhraní Cílová MAC adresa = Broadcast: FF-FF-FF-FF-FF-FF			
Paket IP Zdrojová IP adresa = adresa vysílajícího rozhraní Cílová IP adresa = Broadcast: 255.255.255.255 Protokol = 17 pro UDP			
		Segment (datagram) UDP Zdrojový port = 520 Cílový port = 520	
		Zpráva RIP Command: Request (1); Response (2) Version = 1 Address Family ID (AFI) = 2 pro IP Trasy: IP adresa sítě Metrika: počet přeskoků	

Formát zprávy RIP: Záhlaví RIP

(V závorkách je u názvu pole uvedena jeho velikost v bajtech.)



Ve čtyřbajtovém záhlaví (na předchozím obrázku zvýrazněném oranžově) jsou 3 pole:

- Pole Příkaz (*Command*) určuje typ zprávy, podrobněji se tím budeme zabývat v následující části.
- Pole Verze (*Version*) je nastaveno na 1 pro RIP verze 1.
- Třetí označené pole musí být nulové. Pole "Musí být nula" (*Must be zero*) poskytuje prostor pro budoucí rozšíření protokolu.

Formát zprávy RIP: Vstup trasy

Část zprávy se vstupem trasy se skládá ze tří polí s obsahem:

- **Identifikátor skupiny adres** (Address Family ID, AFI) (nastaven na **2 pro IP**; pokud požaduje úplné směrovací tabulky, je v takovém případě pole nastaveno na nulu),
- **IP adresa sítě**,
- **Metrika**.

Tato část Vstup trasy představuje jednu trasu do cíle a s ní spojenou metriku. Jedna aktualizace RIP

může obsahovat až 25 řádek tras. Maximální velikost datagramu je 504 bajtů, bez jednotlivých záhlaví IP nebo UDP.

Proč je zde tolik polí nastavených na nulu? RIP byla vyvinut ještě před protokolem IP a byl používán pro jiné síťové protokoly (jako XNS). Distribuce BSD měla také svůj vliv. Na počátku bylo do zprávy přidáno další volné místo s cílem podpořit větší adresové prostory v budoucnosti. Jak uvidíme v kapitole 7, RIPv2 nyní používá většinu z těchto prázdných polí.

Provoz RIP

Zpracování Poptávky/Odpovědi RIP

RIP používá dva typy zpráv uvedených v poli Příkaz (*Command*):

- Žádost (*Request*)
- Odpověď (*Response*).

Každé rozhraní s nastaveným RIP vysílá při startu zprávu typu Žádost požadující, aby všechny RIP sousedé zaslali své kompletní směrovací tabulky. Zpráva typu Odpověď je odeslána zpět sousedy se spuštěným protokolem RIP. Když žádající směrovač přijme odpovědi, vyhodnotí každou řádku s trasou. Je-li trasa nová, přijímající směrovač trasu přidá do směrovací tabulky. Je-li trasa již ve směrovací tabulce, je existující řádka nahrazena pouze pokud má nový vstup lepší (to jest menší) počet přeskoků (*hop count*). Směrovač po svém spuštění pošle automaticky spouštěnou aktualizaci, obsahující jeho vlastní směrovací tabulkou, ze všech rozhraní s povoleným protokolem RIP, takže sousedi (s běžícím RIP) mohou být informováni o všech nových trasách.

Třídy IP adresy a třídní směrování

Možná si vzpomínáte z předchozích studií, že IP adresy přidělené počítačům byly původně rozděleny do 3 tříd: třídy A, třídy B a třídy C. Každá třída měla přidělenou implicitní masku podsítě, jak je uvedeno na obrázku. Znát výchozí masku podsítě pro každou třídu je důležité pro pochopení toho, jak funguje RIP.

RIP je (plno)třídní směrovací protokol. Jak jste možná pochopili z předchozí diskuse formátu zprávy, RIPv1 neposílá v aktualizaci masku podsítě. Proto směrovač použije buď masku podsítě nastavenou na lokálním rozhraní nebo použije výchozí masku podsítě na základě třídy adresy. Kvůli tomuto omezení nesmí být sítě RIPv1 nesouvislé ani nemohou implementovat VLSM.

IP adresace je dále probírána v kapitole 6, "VLSM a CIDR." Můžete také navštívit níže uvedené odkazy pro zopakování tříd sítí a tvorbu podsítí.

Odkazy:

"Internet Protocol," <http://www.ietf.org/rfc/rfc791.txt>

"IP adresace a tvorba podsítí pro nové uživatele" http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a00800a67f5.shtml

Administrativní vzdálenost

Jak víte z kapitoly 3, "Úvod do dynamických směrovacích protokolů," administrativní vzdálenost (AD) je důvěryhodnost (neboli preference) zdroje tras. RIP má standardní administrativní vzdálenost 120. Ve srovnání s jinými vnitřními směrovacími protokoly je RIP nejméně preferovaný smě-

rovací protokol. Protokoly IS-IS, OSPF, IGRP, EIGRP mají všechny nižší implicitní hodnoty AD (než RIP).

Pamatujte si, že administrativní vzdálenosti můžete zkontrolovat pomocí příkazů "show ip route" nebo "show ip protocols".

R2#sh ip protocols

```

Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 10 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 1, receive any version
    Interface      Send   Recv   Triggered RIP  Key-chain
    Serial0/1/0        1       2     1
  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    172.16.0.0
  Passive Interface(s):
    FastEthernet0/0
  Routing Information Sources:
    Gateway          Distance      Last Update
    172.16.2.253        120          00:00:06
  Distance: (default is 120)
R2#

```

Základní konfigurace RIPv1

Zapnutí RIP: příkazem "router rip"

Chcete-li zapnout dynamický směrovací protokol, vstupte do globálního konfiguračního módu a použijte příkaz router. Jak je patrné z obrázku, pokud zadáte mezerník následovaný otazníkem, IOS zobrazí seznam všech dostupných směrovacích protokolů.

Pro vstup do konfigurace směrovacího protokolu RIP zadejte v globálním konfiguračním režimu "router rip". Všimněte si, že se změnila systémová výzva z globálního konfiguračního režimu na následující:

```
R1 (config-router) #
```

Tento příkaz nespouští přímo proces RIP. Místo toho poskytuje přístup ke konfiguraci nastavení směrovacích protokolů. Nejsou odeslány žádné směrovací aktualizace.

Pokud potřebujete ze zařízení zcela odstranit proces směrování RIP, negujte příkaz s "no" - "no router rip". Tento příkaz zastaví proces RIP a vymaže všechny stávající konfigurace RIP.

Zadání sítí

```
Router(config-router) #network directly-connected-classful-network-address
```

Příkaz network:

- Zapíná protokol RIP na všech rozhraních, které patří do specifikované sítě. Přidružená rozhraní budou nyní vysílat i přijímat aktualizace RIP.
- Inzeruje specifikované sítě ve směrovacích aktualizacích RIP na ostatní směrovače každých 30 sekund. (Zde **nastavte všechny přilehlé sítě k příslušnému směrovači v plné třídě** (odmaskované implicitní maskou příslušné třídy).)

Klíčové charakteristiky RIPv1:

- **Protokol typu vektor vzdálenosti (Distance-vector protocol)**
- Používá UDP port 520.
- **Třídní protokol (Classful protocol)** (nepodporuje VLSM a nebo CIDR²¹).
- **Metrika (metric) = počet skoků (router hop count)**
- Maximální počet skoků je 15., nedosažitelné cesty mají metriku 16. Cesty s počtem skoků větším než 15 jsou inzerované jako neplatné, nedostupné (*unreachable*)
- Periodické směrovací aktualizace jsou vysílány všesměrově (**broadcast**) každých 30 sekund na adresu 255.255.255.255.
- 25 cest v jedné zprávě RIP
- Nepodporuje autentizaci (*authentication*)
- Implementuje rozložený horizont s otrávenými zpětnými informacemi (*split horizon with poison reverse*)
- Implementuje automaticky spouštěné aktualizace (*triggered updates*) při změně přímo připojené sítě
- V aktualizaci není obsažena maska podsítě (*subnet mask*)
- Administrativní vzdálenost (*administrative distance*) RIPv1 (i verze 2) je 120
- Použití: pouze v malých, plochých sítích nebo na okraji velkých sítí.

Základní nastavení RIPv1

```
Router(config)#router rip
```

```
Router(config-router)#network <přímo připojená síť v plné třídě>
```

21 CIDR lze v RIPv1 použít **pouze na souvislé tranzitní sítě**.

... pro **všechny přímo připojené sítě v plné třídě**, které a do kterých chceme propagovat směrovací aktualizace

Příkaz **network <sít>**:

- Zapíná protokol RIP na všech rozhraních spadajících pod uvedenou síť. Přidružená rozhraní budou jak vysílat, tak i přijímat směrovací aktualizace z protokolu RIP.
- Inzeruje uvedenou síť ve směrovacích aktualizacích protokolu RIP vysílaných na ostatní směrovače každých 30 sekund.

Poznámka: Pokud vložíte adresu podsítě, IOS ji automaticky převede na adresu sítě v plné třídě. Například: jestliže vložíte příkaz *network 192.168.1.32*, směrovač ho převede na *network 192.168.1.0* odmaskováním implicitní maskou příslušné třídy.

Propagace implicitní cesty:

```
Router(config)#router rip
Router(config-router)#default-information originate
```

Nastavení implicitní cesty manuálně administrátorem je u dynamického směrování obvyklé pouze na hraničním směrovači.

Ověření a hledání chyb konfigurace RIPv1

Příkazy:

show ip route

Zobrazená řádka směrovací tabulky

```
R      192.168.1.0/24 [120/1] via 172.16.0.1, 00:00:04, Serial0/0/1
```

Interpretace cesty:

Výstup	Popis
R	Identifikuje jako zdroj cesty RIP.
192.168.1.0	Indikuje adresu vzdálené sítě.
/24	Zobrazuje masku podsítě použitou na tuto síť.
[120/1]	Zobrazuje administrativní vzdálenost (120) a metriku (1 hop).
via 172.16.0.1,	Specifikuje next-hop IP adresu směrovače (adresu dalšího skoku), přes kterou se posílá provoz do uvedené vzdálené sítě.
00:00:04,	Specifikuje dobu uběhlou od poslední aktualizace (zde 4 sekundy). Další aktualizace přijde za 26 sekund.
Serial0/0/1	Specifikuje lokální rozhraní, přes které lze dosáhnout uvedenou vzdálenou sítě.

show ip rip database

Zobrazí všechny nastavené trasy v **interní databázi** protokolu RIP (včetně nastavených cest směrem na, v této chvíli, nedostupné přilehlé sítě, které proto nejsou ve směrovací tabulce):

```
R2#show ip rip database
0.0.0.0/0
[0] via 0.0.0.0, 00:00:19
172.16.1.0/24
[1] via 172.16.2.253, 00:00:13, Serial0/1/0
172.16.2.0/24      directly connected, Serial0/1/0
172.16.3.0/24      directly connected, FastEthernet0/0
R2#
```

V uvedeném příkladě je implicitní cesta nastavena na dosud nezprovozněnou přilehlou síť a je v databázi RIP, ale není ve směrovací tabulce.

```
R2#show ip route
<vynecháno>
Gateway of last resort is not set

172.16.0.0/24 is subnetted, 3 subnets
R    172.16.1.0 [120/1] via 172.16.2.253, 00:00:14, Serial0/1/0
C    172.16.2.0 is directly connected, Serial0/1/0
C    172.16.3.0 is directly connected, FastEthernet0/0
R2#
```

show ip protocols

Zobrazí aktuální konfiguraci všech směrovacích protokolů na spuštěných na směrovači, na kterém tento příkaz vložíte.

Z výstupu tohoto příkazu zjistíte (pro RIP) následující informace:

1. název směrovacího protokolu,
2. nastavené hodnoty časovačů a kdy bude další periodická aktualizace,
3. nastavenou filtrace vysílaných aktualizací a nastavenou redistribuci do jiných směrovacích protokolů,
4. jednotlivá rozhraní, která vysílají a přijímají aktualizace a ve které verzi RIP (1 nebo 2),
5. zda je v činnosti automatická summarizace (ta je vždy na hranici plné třídy) a maximální počet cest RIP (*Maximum Path*) se stejnou cenou (*equal-cost*) do jedné konkrétní sítě,
6. směrování do uvedených sítí v plné třídě nastavené při konfiguraci směrovacího protokolu,
7. zdroje směrovacích informací – tj. sousední směrovače, ze kterých směrovač přijímá aktualizace; včetně informace o adrese dalšího skoku, administrativní vzdálenosti a času, kdy byla

Cisco NetAcad: CCNA Exploration - Routing Protocols and Concepts – studijní materiál

přijata poslední aktualizace; poslední řádka výpisu zobrazuje administrativní vzdálenost tohoto směrovače.

```
R1#show ip protocols
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 25 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 1, receive any version
    Interface          Send   Recv Triggered RIP  Key-chain
    FastEthernet0/0     1       2 1
    FastEthernet1/0     1       2 1
    FastEthernet1/1     1       2 1
  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    172.16.0.0
    192.168.1.0
  Passive Interface(s):
  Routing Information Sources:
    Gateway          Distance      Last Update
    172.16.3.254      120          00:00:00
    172.16.1.254      120          00:00:04
  Distance: (default is 120)
R1#sh ip protocols
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 23 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 1, receive any version
    Interface          Send   Recv Triggered RIP  Key-chain
    FastEthernet0/0     1       2 1
    FastEthernet1/0     1       2 1
    FastEthernet1/1     1       2 1
  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    172.16.0.0
    192.168.1.0
  Passive Interface(s):
  Routing Information Sources:
    Gateway          Distance      Last Update
    172.16.3.254      120          00:00:03
    172.16.1.254      120          00:00:06
  Distance: (default is 120)
R1#
```

debug ip rip

Ladění (*debugging*) směrovacího protokolu RIP. On-line okamžitý výpis všech událostí (aktualizací, změn ve směrovací tabulce, ...) o příslušném směrovacím protokolu.

POZOR: ladění (*debugging*) velice zatěžuje CPU směrovače, zapínejte ho proto pouze na nezbytnou dobu pro dohledávání chyb (a nikoliv při běžném provozu).

```
R1#debug ip rip
```

```
RIP protocol debugging is on
RIP: received v1 update from 172.16.1.6 on Serial0/0
    172.16.1.8 in 1 hops
    192.168.2.0 in 1 hops
    192.168.3.0 in 2 hops
RIP: sending v1 update to 255.255.255.255 via Loopback1 (10.0.0.1)
RIP: build update entries
    network 0.0.0.0 metric 1
    network 172.16.0.0 metric 1
    network 192.168.1.0 metric 1
    network 192.168.2.0 metric 2
    network 192.168.3.0 metric 2
RIP: sending v1 update to 255.255.255.255 via Serial0/0 (172.16.1.5)
RIP: build update entries
    network 0.0.0.0 metric 1
    network 10.0.0.0 metric 1
    network 172.16.1.12 metric 1
    network 192.168.1.0 metric 1
    network 192.168.3.0 metric 2
RIP: sending v1 update to 255.255.255.255 via Serial0/1 (172.16.1.14)
< ... >
R1#no debug all
All possible debugging has been turned off
```

Automatická summarizace na hraničním směrovači

Jak víte, RIP (verze 1) je třídní směrovací protokol, který automaticky summarizuje podsítě do třídní sítě (to znamená odmaskuje implicitní maskou) na hraničních hlavních (třídních) sítí (*major (classful) network*) (na tzv. hraničním směrovači (*boundary router*)).

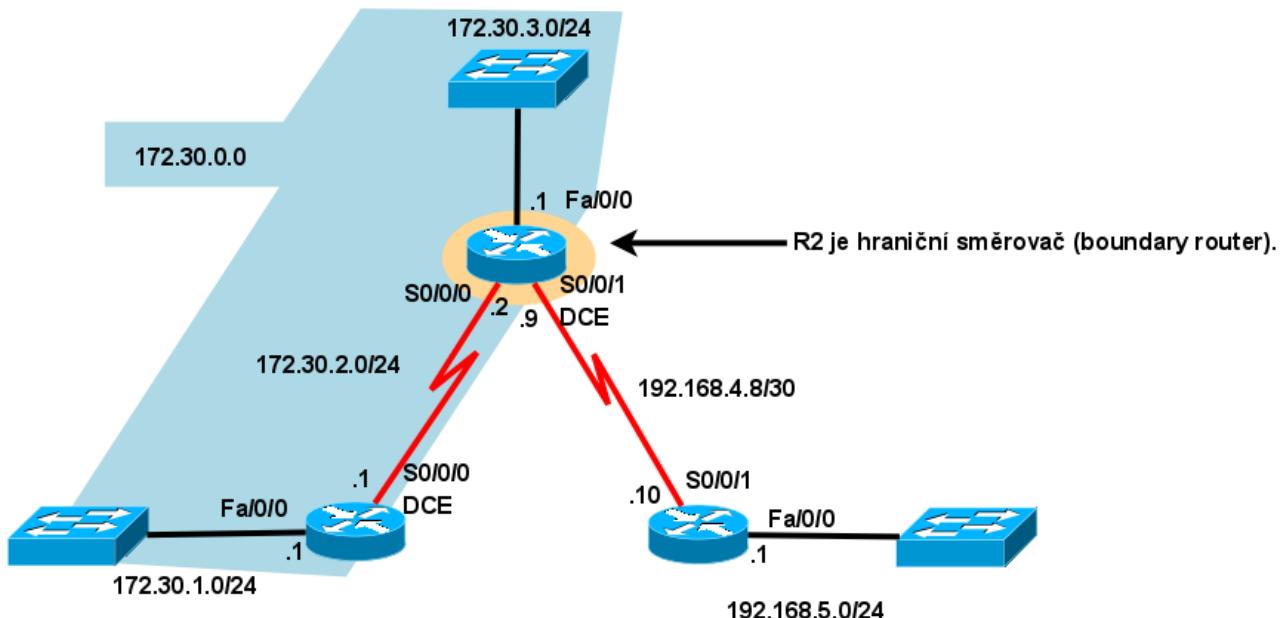
Pro aktualizace protokolu RIPv1 platí následující dvě pravidla:

- Jestliže síť ve směrovací aktualizaci a rozhraní, na které je ta aktualizace přijatá, spadají do stejné plnotřídní sítě, je na síť v řádce směrovací aktualizaci aplikována maska podsítě z tohoto síťového rozhraní.
- Jestliže síť ve směrovací aktualizaci a rozhraní, na které je ta aktualizace přijatá, spadají do různých plnotřídních sítí, je na síť v řádce směrovací aktualizace aplikována implicitní třídní síťová maska.

Směrovače, na kterých běží RIPv1 jsou omezeny používat tu samou masku podsítě pro všechny podsítě ve stejné třídní síti (CIDR).

Beztřídní směrovací protokoly (jako je RIPv2), dovolují stejné hlavní (třídní) sítě používat různé masky podsítě v různých podsítích (VLSM).

Hraniční směrovač u RIP (RIP boundary Router)



Příkazy pro kapitolu 5, RIPv1

Příkaz (Command)	Popis (Description)
Router(config)# router rip	Spouští směrování RIP
Router(config-router)# network 10.0.0.0	Umožní aktualizace RIP na rozhraních spadajících pod síť (<u>v plné třídě</u>) 10.0.0.0. Nastavte všechny přilehlé sítě v plné třídě .
Router(config-router)#passive-interface fa0/0	Ukončuje vysílání aktualizací z rozhraní FastEthernet 0/0 (<u>příjem ale trvá</u>).
Router(config-router)#default-information originate	Tento směrovač propaguje ve svých aktualizacích RIP implicitní cestu. Nastaví se pouze na tom směrovači, kde je nastavena implicitní cesta (= hraniční směrovač (<i>boundary router</i>) s ISP).
Router# show ip protocols	Zobrazí detailní informace o všech procesech dynamického směrování na tomto směrovači.
Router# debug ip rip	Umožní monitoring aktualizací RIP, jak jsou vysílány a přijímány.
Router# no debug all	Vypíná veškeré ladění

Odstaňování chyb RIP

Router#debug ip rip	Zobrazí všechny aktivity RIP v reálném čase.
Router#show ip rip database	Zobrazí obsah databáze RIP.

Další příkazy pro RIP viz [souhrn](#) v kapitole 7.

Komplexní praktické laboratorní cvičení – dynamické směrování RIPv1 a summarizace

Mějme 4 směrovače:

- R1, R2 a R3 (Cisco 2620XM se zásuvným modulem NM-2FE2W), které jsou zapojené do kruhu, a ke každému z nich (s výjimkou R3) je připojen jeden přepínač (2950-24) a za přepínačem jedno PC.
- Za R3 je připojen směrovač R4 (1841) a teprve za ním přepínač (2950) s PC.
- Směrovače jsou propojeny do kruhu přes rozhraní FastEthernet. Použita je privátní adresa sítě ve třídě B podsíťovaná implicitní maskou třídy C (/24).
- Lokální sítě (s přepínači a hostitelskými počítači) jsou připojeny prostřednictvím privátních sítí v rozsahu třídy C s implicitní maskou.

Pro směrování použijte vnitřní beztrídní směrovací protokol RIPv1. Do sítí s pouze koncovými stanicemi zakažte propagaci RIP, ale tyto sítě samotné propagujte do ostatních sítí.

Postup práce:

1. Nejprve si nakreslete topologické schéma zapojení včetně adres sítí a názvů portů.
2. Vyplňte (doplňte) následující tabulku adres sítiových rozhraní:

Zařízení	Rozhraní	IP adresa	Maska	Brána
R1	Fa0/0	192.168.1.0	255.255.255.0	
	Fa1/0	172.16.1.0	255.255.255.0	
	Fa1/1	172.16.3.0	255.255.255.0	
R2	...			
...				

3. Nastavte a ověřte směrování, **automaticky summarizovanou cestu v plné třídě na R4** a konvergenci po změně topologie (výpadku linky).
4. Na směrovači R4 nastavte implicitní cestu na odchozí rozhraní směrem na switch a PC. Tuto implicitní cestu pak propagujte na ostatní směrovače.
5. Na směrovačích s připojenými sítěmi, které obsahují pouze koncová zařízení (PC) (= netranzitní síť), zakažte propagaci RIP dovnitř téhoto sítí natavením příslušného rozhraní jako pasivního (*passive-interface*).

Příklad konfigurace RIP na R2:

```
!
router rip
  passive-interface FastEthernet0/0
  network 172.16.0.0
```

```
network 192.168.2.0
```

```
!
```

Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Co je pravda ohledně příkazu „**debug ip rip**“?
 - a) Vypisuje aktualizace RIP online, tak jak jsou aktuálně vysílány a přijímány
- 2) Jaký problém pomáhá řešit příkaz „**passive interface**“?
 - a) Pasivní rozhraní (= neposílá aktualizace, ale pouze je přijímá) předchází zbytečné záteži linky a procesoru nepotřebnými směrovacími aktualizacemi (do netrannitních sítí, *stub network*)
- 3) Co ze směrovače činí hraniční směrovač (*boundary router*) pro RIP?
 - a) Směrovač má několik rozhraní ve více než jedné hlavní třídní síti (=> probíhá na něm automatická summarizace směru do plné třídy, u RIPv2 lze summarize vypnout u RIPv1 summarizace nelze vypnout => je to vlastnost tohoto směrovacího protokolu)
- 4) Který příkaz je v RIP použit pro propagaci implicitní cesty?
 - a) **default-information originate**
- 5) Který příkaz vytvoří kandidáta na implicitní cestu v RIP?
 - a) Např.: **ip route 0.0.0.0 0.0.0.0 serial0/0/0**
- 6) Máte zapojené čtyři směrovače do kruhu pomocí čtyř propojovacích (= tranzitních) sítí: 192.168.8.0/30, 192.168.10.4/30, 192.168.11.12/30 a 192.168.9.0/30. Všechna rozhraní směrovačů jsou nastavená a zapnutá. L2 protokoly funkční. Na směrovačích je spuštěn RIPv1. Nelze se dopinknout mezi dvěma netrannitními sítěmi (v třídě A).
 - a) Tranzitní síť je tvořena nesouvislými sítěmi (podsítě různých sítí v plné třídě - vzhledem k implicitní masce třídy C /24).
- 7) Jak směrovač, na kterém běží RIPv1, masku podsítě, kterou přijal ze směrovací aktualizace?
 - a) Směrovač bud' použije implicitní masku, nebo pokud jde o podsítě plné třídy, ve které leží rozhraní, na které aktualizace přišla, použije masku nastavenou na přijímajícím rozhraní.
- 8) Jaký je účel příkazů **network** při konfiguraci směrovacího protokolu RIP?
 - a) Identifikují všechny přímo připojené sítě, které budou zahrnuté do směrovacích aktualizací
- 9) Spárování příkazů a jejich popisů (pro příkazy, které síťový administrátor používá pro ověření konfigurace směrovače):
 - a) výpis aktuální konfigurace rozhraní a směrovacích protokolů = show running-config
 - b) zobrazí zda jsou rozhraní zapnutá a protokoly funkční = show ip interfaces
 - c) online výpis směrovacích aktualizací, tak jak jsou aktuálně vysílány a přijímány = debug

ip rip

- d) vypíše všechny běžící směrovací protokoly na směrovači, a které sítě propagují = show ip protocols
- e) ověří, zda jsou všechny požadované směry (cílové sítě) nainstalované do směrovací tabulky = show ip route

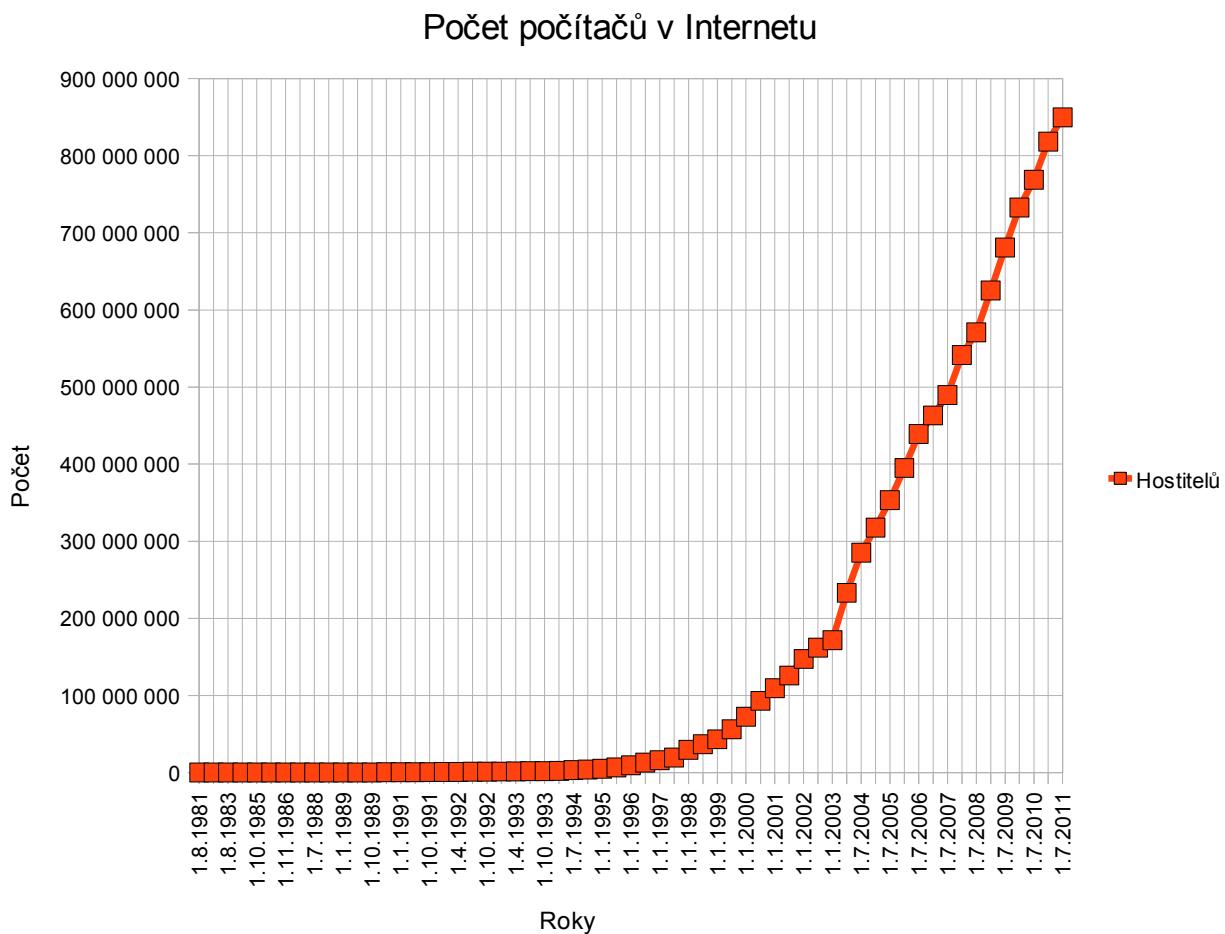
Kapitola 6 - VLSM a CIDR

V této kapitole se naučíme:

- Porovnat a zdůraznit rozdíly třídní (*classful*) a beztřídní (*classless*) IP adresace
- Přehled VLSM a vysvětlit přínosy beztřídní IP adresace
- Popsat roli beztřídního standardu CIDR (*Classless Inter-Domain Routing*) při efektivním použití nedostatkových adres IPv4

Adresní systémy pro IPv4

Před rokem 1981, používaly IP adresy k určení síťové části adresy pouze prvních 8 bitů, omezujících Internet, tehdy známý jako ARPANET, na pouhých 256 sítí. Brzo začalo být zřejmé, že to nebude stačit, aby byl dostatek volných adres.



(Zdroj dat: <https://www.isc.org/solutions/survey/history>)

V roce 1981 modifikoval dokument RFC 791 32-bitovou adresu protokolu IPv4 na tři různé třídy: A,B a C. Třída A používala 8 bitů pro síťovou část, třída B používala 16 bitů a třída C používala 24

bitů. Tento formát se stal známým jako třídní IP adresace.

Tento vývoj třídní adresace na čas řešil problém omezení na 256 sítí. O desetiletí později bylo ale jasné, že se adresní prostor rychle vyčerpává. Jako odpověď na to IETF zavedl adresní systémy CIDR (*Classless Inter-Domain Routing*) (RFC 1519 - 1993) a VLSM (*Variable Length Subnet Masking*).

Jednotliví poskytovatelé ISP nyní mohou přiřadit jednu část třídní sítě jednomu zákazníkovi a další část jinému zákazníkovi. Toto nesouvislé přiřazování adresy ze strany ISP bylo paralelně následováno vývojem beztřídních směrovacích protokolů. Pro srovnání: Třídní směrovací protokoly vždy sumarizují na hranici třídy a neobsahují masku v aktualizacích. Beztřídní směrovací protokoly obsahují ve směrovacích aktualizacích masku. Beztřídní směrovací protokoly diskutované v tomto kurzu jsou RIPv2, EIGRP a OSPF.

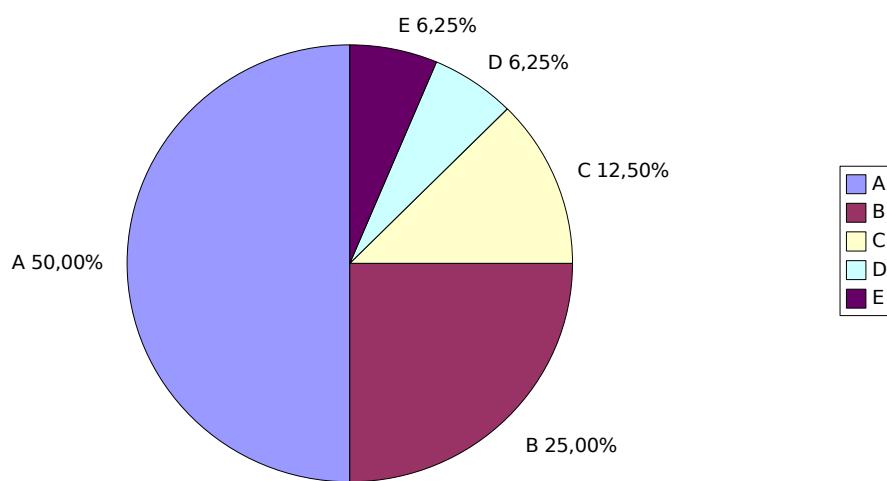
Se zavedením VLSM a CIDR musí ale síťoví administrátoři začít používat další znalosti z podsíťování. VLSM je jednoduše podsíťování podsítí. Podsítě mohou být podsíťovány v různých úrovních (s různými délками masek). A co více, stalo se možným sumarizovat celou sadu třídních sítí do agregovaného směru neboli nadsítě (*supernet*) s maskou kratší než je implicitní třídní maska.

Počet sítí a hostitelů v jedné síti podle tříd

Třída adres (Address Class)	Rozsah prvního oktetu (First Octet Range)	(Počet možných sítí) Number of Possible Networks	Počet hostitelů v jedné síti (Number of Hosts per Network)
A	0 až 127	128 (2 jsou rezervované)	16 777 214
B	128 až 191	16 384	65 534
C	192 až 223	2 097 152	254

Třída	Počet adres celkem ve třídě	Počet sítí	Počet adres v jedné síti
A	2 147 483 648	128	16 777 216
B	1 073 741 824	16 384	65 536
C	536 870 912	2 097 152	256
D	268 435 456	1	268 435 456
E	268 435 456	1	268 435 456
Celkem	4 294 967 296		

Podíl adres v jednotlivých třídách



Třídní a beztřídní adresace a směrování

Pojmy:

	Třídní (Classful)	Beztřídní (Classless)
IP adresace	Pouze sítě ve třídách A, B, C	<ul style="list-style-type: none"> • CIDR • VLSM
Směrování	V aktualizaci není maska	V aktualizaci je i maska

IP adresace

- **Třídní** - IP adresy pouze v celých třídách (plýtvá nedostatkovými IP adresami)
- **Beztřídní – CIDR** – IP adresy jsou v podsítích třídní adresy a všechny masky jsou stejné (adresní bloky jsou stejně velké) (mírně snižuje plýtvání adresami, při summarizaci snižuje velikost směrovací tabulky, a tím také snižuje aktualizační provoz) (RFC 1517)
- **Beztřídní – VLSM** – IP adresy jsou v podsítích jiné podsítě (masky jsou různé), je třeba dát pozor na překrývání adresních bloků (*overlapping*). Překrývání adres je detekováno a odmítnuto přímo operačním systémem směrovače (IOS). Použití VLSM snižuje plýtvání IP adresami, využívá adresní prostor ještě lépe než CIDR. Postup při vytváření: postupujte od největšího adresního bloku k nejmenšímu. Nezapomeňte, že prodloužení masky o jeden bit

zmenší adresní blok na polovinu.

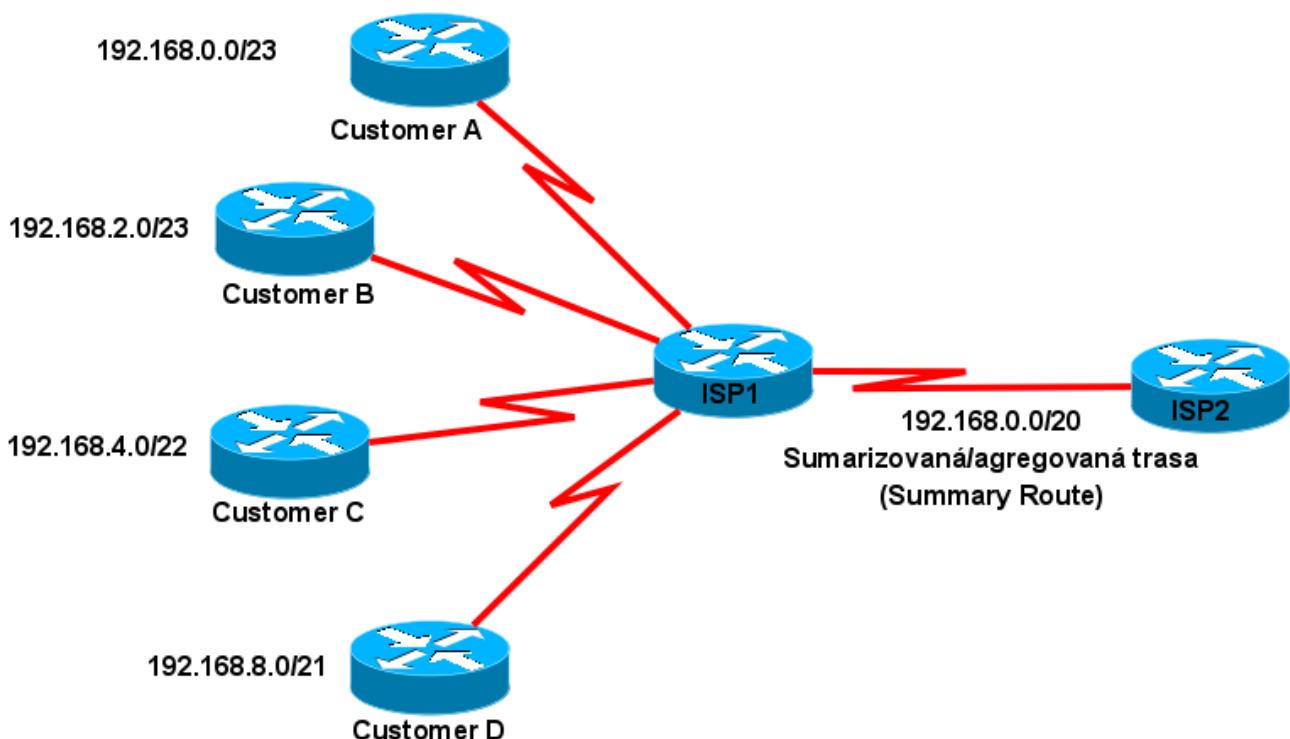
Směrování

- **Třídní** – nepřenáší se masky, jako maska je použita buď implicitní maska nebo, pokud cílová adresa leží ve stejné síti (vzhledem k implicitní masce) jako rozhraní směrovače (na které aktualizace přišla (*ingress interface*)), tak maska nastavená na tomto síťovém rozhraní. Tzn. Lze použít i jeden CIDR v souvislé (tranzitní) síti. **Souvislá, na sebe navazující, síť (contiguous network)** = všechny podsítě jsou v jedné síti v plné třídě (tj. vzhledem k implicitní masce třídy).
- **Beztrídní** – lze použít VLSM.

Zvláštní typy rozhraní směrovače

- **Loopback (zpětná smyčka)** – tato virtuální rozhraní mají v konfiguračním režimu pro rozhraní nastavenou IP adresu a masku, jsou implicitně administrativně zapnuté a mají nahozený L2 protokol, aniž je na nich vůbec něco připojeného. Na toto rozhraní lze pinknout a dostat odpověď. Používá se pro nastavení implicitní cesty, a simulaci (dosud) neexistujícího připojení na ISP. Nastavenou implicitní cesta lze dynamicky propagovat (*default-information originate*).
- **Null (neexistující prázdné rozhraní)** – tato rozhraní lze použít pro nastavení statických cest, které lze dynamicky propagovat (*redistribute static*).

CIDR a summarizace trasy (CIDR and Route Summarization)



Výpočet summarizované (agregované) cesty

CIDR:

192.168.64.0/22	11000000.10101000.01000000.00000000
192.168.68.0/22	11000000.10101000.01000100.00000000
192.168.72.0/22	11000000.10101000.01001000.00000000
192.168.76.0/22	11000000.10101000.01001100.00000000
Agregovaná cesta : 192.168.64.0/20	11000000.10101000.01000000.00000000

VLSM:

172.16.0.0/19	10101100.00010000.00000000.00000000
172.16.32.0/19	10101100.00010000.00100000.00000000
172.16.64.0/18	10101100.00010000.01000000.00000000
172.16.128.0/17	10101100.00010000.10000000.00000000
Agregovaná cesta : 172.16.0.0/16	10101100.00010000.00000000.00000000

Příklady summarizace

Příklad

Máte 4 následující třídní sítě: 172.20.0.0/16 , 172.21.0.0/16 , 172.22.0.0/16 a 172.23.0.0/16.

Spočtěte summarizaci do nadsítě (*supernet*) bez použití binárního tvaru.

Řešení:

B/C numericky nejvyšší síť mínus síťová adresa nejnižší síť, dvojkový doplněk z toho je roven masce nadsítě.

$172.23.255.255 - 172.20.0.0 = 0.3.255.255$ dvojkový doplněk je maska nadsítě $255.252.0.0 = /14$.

Sumarizovaná nadsíť je tedy 172.20.0.0/14.

Další podobný příklad

Máte 6 následujících třídních sítí: 172.16.0.0/16, 172.17.0.0/16, 172.18.0.0/16, 172.19.0.0/16 , 172.20.0.0/16 a 172.21.0.0/16.

Spočtěte summarizaci do nadsítě (*supernet*) bez použití binárního tvaru.

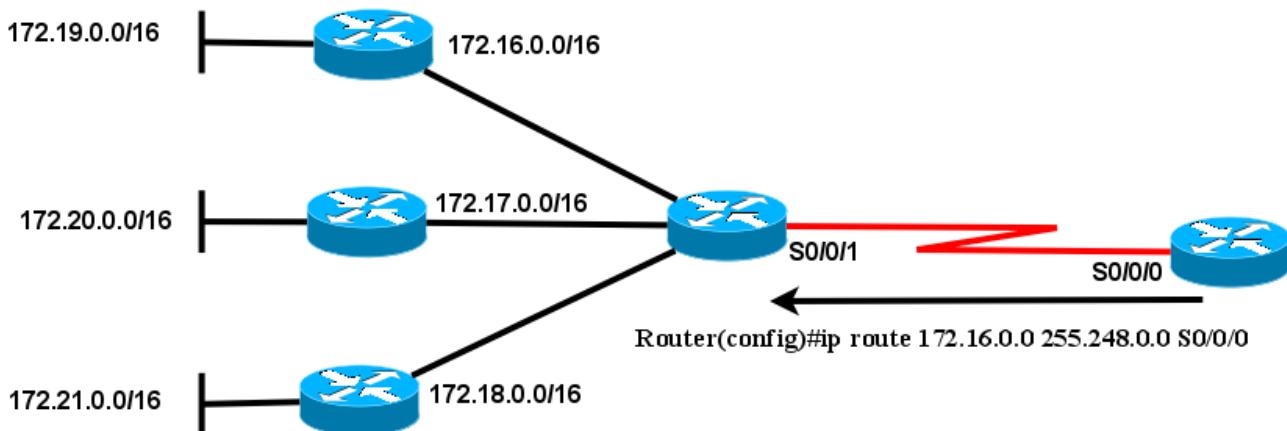
Řešení:

B/C numericky nejvyšší síť mínus síťová adresa nejnižší síť, dvojkový doplněk z toho je roven masce nadsítě.

$172.21.255.255 - 172.16.0.0 = 0.5.255.255 \rightarrow$ oprava na číslo, které má binárně zprava samé jedničky: 0.7.255.255, jeho dvojkový doplněk je maska nadsítě $255.248.0.0 = /13$.

Sumarizovaná nadsíť je tedy 172.20.0.0/14.

Sumarizace trasy (Route summarization)



Příklad VLSM

Máte přidělený adresní blok **172.16.128.0/17**. Tento rozsah máme v následujícím příkladu **chybně rozdelen** do v následující tabulce uvedených sítí (**viz originální Aktivita 6.4.3**). Zkontrolujte je a opravte nalezené chyby v chybně navržených adresách sítí. (Na portech směrovačů je třeba nastavit minimální adresy v příslušné síti.)

Kontrolujte vždy automaticky všechna zadání jednotlivých cvičení!

Kontrola předem dá vždy méně práce, než opravovat již jednou nastavenou chybnou konfiguraci.

Podsíť (Subnet)	Počet potřebných IP adres	Adresa sítě – původně navržená	Velikost bloku - Adresa sítě - opravená
HQ LAN1 (HQ = Centrála)	16000	172.16.128.0/19	0.0.64.0 - 172.16.128.0/18
HQ LAN2	8000	172.16.192.0/18	0.0.32.0 - 172.16.192.0/19
Branch1 LAN1 (Branch = Pobočka)	4000	172.16.224.0/20	0.0.16.0 - 172.16.224.0/20
Branch1 LAN2	2000	172.16.240.0/21	0.0.8.0 - 172.16.240.0/21
Branch2 LAN1	1000	172.16.244.0/24	0.0.4.0 - 172.16.248.0/22
Branch2 LAN2	500	172.16.252.0/23	0.0.2.0 - 172.16.252.0/23
Linka z HQ do Branch1	2	172.16.254.0/28	0.0.0.4 - 172.16.254.0/30
Linka z HQ do Branch2	2	172.16.254.6/30	0.0.0.4 - 172.16.254.4/30
Linka z Branch1 do Branch2	2	172.16.254.8/30	0.0.0.4 - 172.16.254.8/30

Laboratorní cvičení - příklad

1. Dva směrovače R1 a R2 propojte přes Ethernet v privátní síti třídy B, podsíťované implicitní maskou pro třídu C (/24).
2. Na R1 nakonfigurujte Loopback 0 (IP adresa a maska), na toto odchozí rozhraní Loopback 0 nastavte implicitní cestu, implicitní cestu dynamicky propagujte pomocí RIPv2.
3. Na R2 je nastavena statická cesta na neexistující rozhraní Null 0, nastavené statické cesty dynamicky propagujte pomocí RIPv2 se zvolenou metrikou = 12 (lze zvolit v rozsahu 0 až 16).

Částečná konfigurace R1:

```
!  
interface Loopback0  
ip address 172.16.100.254 255.255.255.0  
!  
<vynecháno>  
!  
router rip  
version 2  
network 172.16.0.0  
default-information originate  
!  
ip classless  
ip route 0.0.0.0 0.0.0.0 Loopback0  
!
```

Částečná konfigurace R2:

```
!  
router rip  
version 2  
redistribute static metric 12  
network 172.16.0.0  
!  
ip classless  
ip route 192.168.1.0 255.255.255.0 Null0  
!
```

Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Směrovací protokoly, které podporují a nepodporují VLSM:
 - a) podporují VLSM: RIPv2, EIGRP, IS-IS, OSPF
 - b) nepodporují VLSM: RIPv1, IGRP.
- 2) Přiřaďte k termínům odpovídající popisky:
 - a) VLSM:
 - i. schopnost pro jednu (pod)sítě vytvářet její podsítě s různými maskami,
 - ii. šetří adresy (lépe využívá daný adresní blok).
 - b) summarizace směru:
 - i. je také známa jako supernetting,
 - ii. snižuje počty řádek ve směrovací tabulce,
 - iii. v jedné IP adrese kombinuje více sítí.
- 3) Které dvě metody se používají, aby bylo možné stále používat IPv4, i když už jsou třídní a CIDR adresní bloky vyčerpané?
 - a) VLSM,
 - b) privátní adresy a jejich překlad na veřejné pomocí NAT (Network Address Translation) = IP maškaráda.
- 4) Pro síť 192.168.16.0 byly použity následující masky: 255.255.255.252, 255.255.255.240, 255.255.255.192. Popište nejefektivnější použití každé jednotlivé masky:
 - a) maska /30 pro spojení point-to-point k síti typu WAN,
 - b) maska /28 pro malé sítě do 14 hostitelů včetně,
 - c) maska /26 pro velké sítě do 62 hostitelů včetně.
- 5) Když použijete třídní adresu ve třídě A, kolik oktetů tvoří síťovou část IP adresy?
 - a) 1
- 6) Vyberte VLSM podsítě sítě 172.16.0.0, které poskytnou uvedený celkový počet hostitelů v každé jednotlivé podsíti.
 - a) 2 hostitelé = 172.16.16.64/30
 - b) 60 hostitelů = 172.16.5.128/26
 - c) 250 hostitelů = 172.16.18.0/24
 - d) 8000 hostitelů = 172.16.128.0/19
 - e) 16000 hostitelů = 172.16.64.0/18
- 7) Inženýr summarizuje na směrovači A dvě skupiny směrů (cest). Skupinu A: 192.168.0.0/30, 192.168.0.4/30, 192.168.0.8/30, 192.168.0.16/29 a Skupinu B: 192.168.4.0/30, 192.168.5.0/30, 192.168.6.0/30, 192.168.7.0/29. Jaká summarizace bude funkční pro všechny

uvedené podsítě?

- a) 192.168.0.0/21 (řešení: 192.168.7.0-192.168.0.0 = 0.0.7.0 => zleva 21 nul)
- 8) Kolik bitů je použito v adresním prostoru protokolu IPv4?
 - a) 32
- 9) Rozdělte adresy do tříd:
 - a) A: 123.90.78.45, 125.33.33.33, 126.0.0.0,
 - b) B: 128.44.30.1, 129.68.11.45, 191.254.45.0.
- 10) Do jednoho směrovače (R2) máte z levé strany připojeny tři směrovače na kterých jsou připojeny následující netranzitní sítě: 172.16.0.0/16, 172.17.0.0/16 a 172.18.0.0/16 (na každém směrovači jedna). Na tento R2 směrovač je potom z pravé strany připojen jeden směrovač R1. Co by měl administrátor této sítě aplikovat, aby snížil počet řádek ve směrovací tabulce na směrovači R1.
 - a) CIDR

Kapitola 7 - Protokol RIP verze 2

V této kapitole se naučíme:

- Střetnout se s nimi a popsat, jaká jsou omezení protokolu RIPv1
- Použít základní konfigurační příkazy pro RIPv2 (*Routing Information Protocol version 2*) a vyhodnotit aktualizace beztrídního směrování
- Analyzovat výstup ze směrovače, abychom poznali, jak RIPv2 podporuje beztrídní VLSM (*Variable Length Subnet Mask*) a CIDR (*Classless Interdomain Routing*)
- Určit příkazy pro ověření správné činnosti RIPv2 a identifikovat běžné problémy
- Konfigurovat, ověřit a odstranit chyby RIPv2 v praktickém laboratorním cvičení

RIP verze 2 a verze 1

Protokol RIP verze 2 (RIPv2) je definován v RFC 1723. Je to první beztrídní směrovací protokol popisovaný v tomto kurzu. Tabulka *Klasifikace dynamických směrovacích protokolů* v kapitole 3 dává protokol RIPv2 do správné souvislosti vzhledem k jiným směrovacím protokolům. Přestože je RIPv2 vhodným směrovacím protokolem pro některá prostředí, ztratil postupně popularitu ve srovnání s jinými směrovacími protokoly, jako jsou EIGRP, OSPF a IS-IS, které nabízejí více funkcí a jsou více škálovatelné.

I když mohou být méně populární než jiné směrovací protokoly, jsou obě verze RIP ještě vhodné v některých situacích. Přestože RIP postrádá schopnosti mnoha pozdějších protokolů, jeho jednoduchost a široké využití v různých operačních systémech z něj činí ideálního kandidáta pro menší, homogenní síť, kde je nezbytná podpora více dodavatelů - zejména v prostředích UNIX.

Protože je třeba, abyste pochopili RIPv2 - i když ho nebudete chtít používat - zaměří se tato kapitola na rozdíly mezi třídním směrovacím protokolem (RIPv1) a beztrídním směrovacím protokolem (RIPv2) spíše než na samotné detaily RIPv2. Hlavním omezením RIPv1 je to, že je třídním směrovacím protokolem. Jak víte, třídní směrovací protokoly nezahrnují se síťovou adresou do směrovacích aktualizací masku podsítě, což může způsobit problémy v nesouvislých podsítích nebo sítích, které používají adresní strukturu s proměnnou délkou masky (VLSM). Vzhledem k tomu, že RIPv2 je beztrídní směrovací protokol, jsou masky podsítě zahrnuty ve směrovacích aktualizacích, což činí RIPv2 více kompatibilní s moderními směrovacími prostředími.

RIPv2 je vlastně vylepšení funkcí a rozšíření RIPv1, spíše než zcela nový protokol. Některé z těchto rozšířených funkce obsahují:

- Adresu dalšího přeskoku (*Next-hop adres*) ve svých směrovacích aktualizacích
- Použití skupinových adres (*multicast*) v zasílaných aktualizacích
- Je k dispozici možnost ověřování (autentizace)

Stejně jako RIPv1, je RIPv2 směrovacím protokolem typu vektor vzdálenosti. Obě verze RIP sdílejí následují funkce a omezení:

- Použití zadržovacího (*holdown*) a dalších časovačů pro zabránění vzniku směrovacích smyček.

- Použití rozloženého horizontu (*split horizon*) nebo rozloženého horizontu s otrávenou zpětnou informací (*split horizon with poison reverse*) také pro zabránění směrovacích smyček.
- Použití automaticky spouštěné aktualizace (*triggered update*), když dojde ke změně v topologii, pro rychlejší konvergenci.
- Omezení maximálního počtu přeskoků (*hop counts*) na 15 přeskoků, přičemž počet přeskoků rovný 16 znamená nedosažitelnost sítě.

Omezení protokolu RIPv1

- Nepodporuje nesouvislé (tranzitní) sítě
- Nepodporuje VLSM
- Nepodporuje CIDR, které mají summarizované trasy s kratší maskou než je třídní (implicitní) maska této sítě.

Formát zpráv RIPv1 a RIPv2

(V závorkách je u názvu pole uvedena jeho velikost v bajtech.)

RIPv1 (RFC 1058)

0	1	2	3
0 1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
command (1) version (1)=1 must be zero (2)			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
address family identifier (2) must be zero (2)			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
IP address (4)			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
must be zero (4)			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
must be zero (4)			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
metric (4)			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+

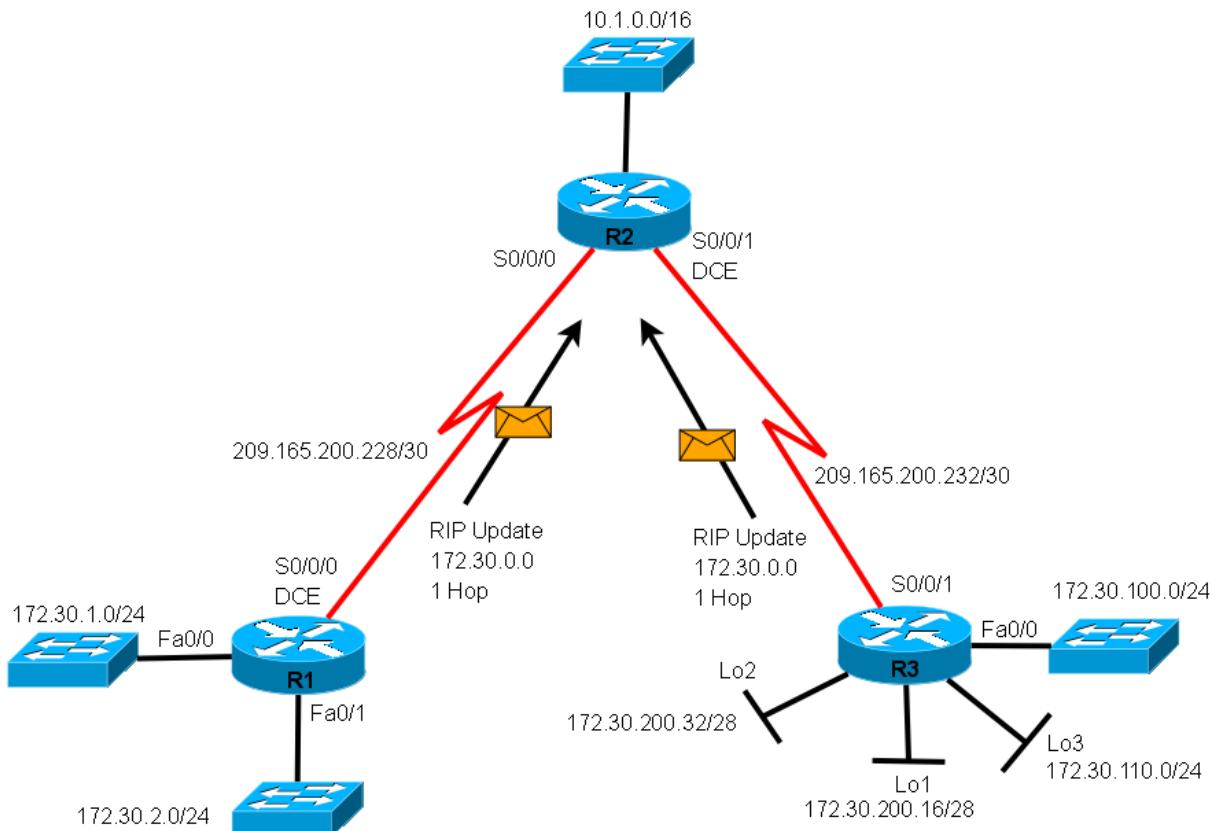
RIPv2 (RFC 1723)

0	1	2	3	3
0 1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1	1
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Command (1) Version (1)=2 unused				
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Address Family Identifier (2) Route Tag (2)				
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
IP Address (4)				
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Subnet Mask (4)				
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Next Hop (4)				
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Metric (4)				
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+

Automatická summarizace a RIPv2

Implicitně RIPv2 automaticky summarizuje síť na hranicích plnotřídní sítě (odmaskováním implicitní maskou třídy), přesně tak jako RIPv1. Zda je summarizace zapnuta zjistíte příkazem „**show ip protocols**“ a jeho odpověď "automatic summarization is in effect."

Automatická summarizace na hranici třídní sítě (Automatic Summarization)



Vypnutí automatické summarizace

K vypnutí automatické summarizace použijte příkaz „**no auto-summary**“ v konfiguračním režimu směrování (config-router) # . Pro RIPv1 je tento příkaz neplatný a ačkoliv ho můžete v IOSu pro RIPv1 zadat nemá žádný efekt.

Jakmile je automatická summarizace vypnuta, RIPv2 již dále na hranicích třídní sítě nesumarizuje do plné třídy. RIPv2 nyní do směrovacích aktualizací zahrne všechny podsítě a jejich masky. Pro ověření, že je automatická summarizace vypnuta použijte příkaz **show ip protocols**, který v tomto případě vrátí, že "automatic network summarization is not in effect."

POZNÁMKA: Uvědomte si, že mohou nastat případy konfigurace, kdy ani přepnutí RIP z verze 1 do verze 2 nestačí ke zprovoznění a je nutné ještě vypnout automatickou summarizaci. (Například: netranzitní podsíť a mezi nimi alespoň dvě třídní tranzitní sítě. Viz předchozí obrázek.) Takže je vhodné si vždy rozmyslet, zda je nutné summarizaci vypnout či nikoliv.

```
R2#show ip protocols
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 5 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 2, receive 2
    Interface          Send   Recv   Triggered RIP  Key-chain
    FastEthernet0/0      2       2
    Serial0/1/1         2       2
Automatic network summarization is in effect
Maximum path: 4
Routing for Networks:
  10.0.0.0
  172.16.0.0
Passive Interface(s):
Routing Information Sources:
  Gateway          Distance      Last Update
  10.0.0.253        120          00:00:13
Distance: (default is 120)
R2#
```

Charakteristiky RIPv2

- **Protokol typu vektor vzdálenosti (Distance-vector protocol)**
- Používá UDP port 520.
- **Beztrídní (classless) směrovací protokol** (podporuje CIDR i VLSM)
- **Metrika = počet skoků (router hop count)**
- **Maximální počet skoků = 15, nekonečné (nedosažitelné) (infinite (unreachable)) cesty mají metriku 16**
- **Periodické aktualizace jsou posílány každých 30 sekund na skupinovou (multicast) adresu 224.0.0.9.**
- 25 cest (routes) v jedné zprávě RIP (24 jestliže používáte autentizaci (*authentication*))
- Podporuje autentizaci (*authentication*)²² (formát zprávy pro toto viz RFC 1723)

²² Nebudeme procvičovat. Pro zájemce: jsou to příkazy: v globální konfiguraci: key chain ..., key 1, key-string ..., a v konfiguraci rozhraní: ip rip authentication key-chain

- Implementuje rozložený horizont s otrávenými zpětnými informacemi (*split horizon with poison reverse*)
- Implementuje automaticky spouštěné aktualizace (*triggered updates*) při změně přímo připojené sítě
- V aktualizaci je obsažena maska podsítě (*subnet mask*)
- Administrativní vzdálenost (*administrative distance*) RIPv2 je 120
- Použití: v malých, plochých sítích nebo na okraji velkých sítí.

Postup hledání chyb konfigurace

Je několik způsobů, jak hledat a odstraňovat chyby v konfiguraci RIPv2. Mnoho z těchto příkazů lze použít i u ostatních směrovacích protokolů.

Je vždy nejlepší začít od základů:

1. Přesvědčte se, že jsou rozhraní zapnutá a funkční. Příkazem show ip interface brief.
2. Ověrte kably. (show controllers ...)
3. Na každém síťovém rozhraní zkонтrolujte IP adresu a masku. (show ip interface brief)
4. Odstraňte v konfiguraci všechny nepotřebné příkazy, které buď nejsou již dále potřeba, nebo budou přepsány jinými příkazy.

Obvyklé problémy s RIPv2

Když odstraňujete chyby specifické pro RIPv2, je zde několik oblastí, kde je dobré začít hledat:

1. ověřit, zda je příkaz „version 2“ nastaven na všech směrovačích,
2. nesprávně vložené nebo chybějící příkazy „network“ – jsou příčinou, že aktualizace nejsou vysílány nebo přijímány na/z určitého rozhraní,
3. pokud není speciální důvod k posílání podsítí do třídní sítě, vypněte automatickou summarizaci příkazem „no auto-summary“,
4. též je dobré zjistit, zda vysílání aktualizací není bezděčně vypnuto příkazem „passive-interface“.

Autentizace

Bezpečnostním problémem jakéhokoliv směrovacího protokolu je možnost přijetí a použití neplatných směrovacích aktualizací. Zdrojem těchto neplatných směrovacích aktualizací může být útočník ve zlé vůli se pokoušející narušit síť a nebo zkoušející zachytávat pakety podvedením směrovače, aby posílal data do nesprávného cíle. Jiným zdrojem neplatných aktualizací může být nesprávně nastavený směrovač a nebo hostitelský počítač, na kterém běží směrovací protokol, o kterém jeho uživatel neví.

Ať už z jakéhokoliv důvodu, je dobré autentizovat směrovací informace. RIPv2, Enhanced IGRP (EIGRP), Open Shortest Path First (OSPF), Intermediate System-to-Intermediate System (IS-IS) a Border Gateway Protocol (BGP) mohou nastaveny tak, aby šifrovaly a autentizovaly směrovací

informace.

Tato praxe potom jednak utahuje obsah směrovacích informací (šifruje aktualizace ale samozřejmě nešifruje směrovací tabulku). Směrovače potom také akceptují informace z druhého směrovače pouze pokud je nastaven se stejným heslem nebo autentizačními informacemi.

Poznámka: Autentizace nebude v této kapitole dále diskutována. (Bude diskutováno později v souvislosti se zabezpečením – ve čtvrtém semestru.)

Příkazy pro kapitolu 7, RIPv2

<i>Příkaz (Command)</i>	<i>Popis (Description)</i>
Router(config)# router rip	Zapne směrování RIP.
Router(config-router)# version 2	Zapne verzi 2 protokolu RIP.
Router(config-router)# no version 2	Přepne zpět do verze 1 protokolu RIP. RIP přijímá aktualizace verze 2 i 1, vysílá pouze verzi 1.
Router(config-router)# network a.b.c.d	Nastavte všechny přímo připojené sítě v plných třídách , které chcete propagovat, a do kterých se bude propagovat.
Router(config-router)# no auto-summary	RIPv2 automaticky summarizuje sítě na směrovači na hranici plné třídy. (Sumarizace je zapnuta z důvodu zpětné kompatibility s RIPv1.) Tento příkaz vypne automatickou summarizaci (na hranici plné třídy). (V RIPv1 vypnout nelze (tam se summarizuje do plné třídy na hranici plné třídy vždy a bezpodmínečně).
Router# debug ip rip	Zobrazí všechny aktivity RIP v reálném čase (tak jak postupně přicházejí). Pozor zbytečně zatěžuje směrovač a proto použijte pouze při ladění problémů.
Router# undebug all	Vypne veškerá ladění na směrovači.

Souhrn příkazů pro obě verze RIPv1 i RIPv2

Povinné příkazy pro směrování protokolem RIP (pro obě verze 1 i 2)

Router(config)#router rip	Zapne směrovací protokol RIP.
Router(config-router)#network w.x.y.z	w.x.y.z je síťová adresa (číslo sítě) přímo připojené sítě, kterou chceme inzerovat (propagovat).

POZNÁMKA: Inzerovat v RIP lze pouze sítě v plné třídě nikoliv podsítě:

Router(config-router)#network 172.16.0.0

nikoliv

Router(config-router)#network 172.16.10.0

Jestliže inzerujete podsíť, nedostanete zpět chybovou zprávu, protože směrovač automaticky konvertuje podsíť na třídní adresu (odmaskuje implicitní maskou pro příslušnou třídu).

Volitelné příkazy pro RIP (souhrn pro obě verze 1 i 2)

Router(config)#no router rip	Vypne směrovací proces RIP.
Router(config-router)#no network w.x.y.z	Odstraní síť w.x.y.z ze směrovacího procesu RIP.
Router(config-router)#version 2	RIP nyní bude vysílat a přijímat pakety RIPv2.
Router(config-router)#version 1	RIP nyní bude vysílat a přijímat pouze pakety RIPv1.
Router(config-if)#ip rip send version 1	Toto rozhraní bude vysílat pouze pakety RIPv1.
Router(config-if)#ip rip send version 2	Toto rozhraní bude vysílat pouze pakety RIPv2.
Router(config-if)#ip rip send version 1 2	Toto rozhraní bude vysílat oboje pakety RIPv1 i RIPv2.
Router(config-if)#ip rip receive version 1	Toto rozhraní bude přijímat pouze pakety RIPv1.
Router(config-if)#ip rip receive version 2	Toto rozhraní bude přijímat pouze pakety RIPv2.
Router(config-if)#ip rip receive version 1 2	Toto rozhraní bude přijímat oboje pakety RIPv1 i RIPv2.
Router(config-router)#no auto-summary	RIPv2 sumarizuje sítě na hranici plné třídy – z důvodu kompatibility s RIPv1. Tento příkaz vypíná automatickou summarizaci ve verzi 2 .
Router(config-router)#passive-interface s0/0/0	Z tohoto rozhraní nebudou odesílány aktualizace RIP. (Aktualizace mohou ale být přijímány.)
Router(config-router)#neighbor a.b.c.d	Definuje konkrétní sousedy, se kterými si vyměňuje informace.
Router(config-router)#no ip split-horizon	Vypne rozložený horizont (<i>split horizon</i>) (implícitně je rozložený horizont zapnut).
Router(config-router)#ip split-horizon	Znovu zapne rozložený horizont.

Router(config-router)#timers basic 30 90 180 270 360	Změní časovače RIP: 30 = Aktualizační, Update timer (v sekundách) 90 = Neplatné cesty, Invalid timer (v sek.) 180 = Zadržovací, Hold-down timer (v sek.) 270 = Vyprazdňovací, Flush timer (v sekundách) 360 = Doba spánku, Sleep time (v milisek.)
Router(config-router)#maximum-paths N	Omezí počet cest pro vyrovnávání zátěže na N (4 = implicitní hodnota, 6 = maximum).
Router(config-router)#default-information originate	Propaguje statickou implicitní cestu v aktualizaci RIP. Jiný způsob je použít implicitní přilehlou síť ((config)#ip default-network), která je automaticky propagovaná protokolem RIP jako přilehlá síť'.
Router(config-router)#redistribute static Router(config-router)#redistribute static metrics	Propaguje všechny statické cesty v aktualizaci RIP. Propaguje všechny statické cesty v aktualizaci RIP se zvolenou metrikou.

Komplexní praktické laboratorní cvičení – RIPv2

Použijte příklad pro RIPv1.

1. Na jednotlivých směrovačích zapněte RIPv2.
2. Síť mezi směrovači R2 a R3 (172.16.2.0/24) rozdělte na dvě poloviny (172.16.2.0/25 a 172.16.2.128/25) vložte další směrovač R23 a zprovozněte s RIPv2.
3. Na směrovači na hranici plné třídy (= R3) vypněte automatickou summarizaci do plné třídy.
4. Nepropagujte RIPv2 do netranzitních sítí tam, kde jsou pouze koncová zařízení.

Příklad konfigurace RIPv2 na R2:

```
!  
router rip  
version 2  
passive-interface FastEthernet0/0  
network 172.16.0.0  
network 192.168.2.0  
!
```

Dokumentace nastavení

Po dokončení a ověření úlohy pro každý směrovač uložte do textového souboru výstupy následujících příkazů:

- show running-config
- show ip route
- show ip interface brief
- show ip protocols

Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Které dva příkazy identifikují, zda je u RIPv2 použita automatická summarizace?
 - a) show running-config
 - b) show ip protocols
- 2) Které tvrzení o RIPv2 je pravdivé?
 - a) RIPv2 bude provádět automatickou summarizaci na hranici hlavní sítě (*major network boundary*) (= hranici sítě v plné třídě)
- 3) Jaké je implicitní chování protokolu RIP při příjmu a vysílání aktualizací v jednotlivých verzích, pokud není specifikováno žádné číslo verze?
 - a) Vysílá aktualizace pouze ve verzi 1, přijímá aktualizace ve verzích 1 i 2.
- 4) Co by umožnil RIPv2 a nikoliv RIPv1?
 - a) Například síť 192.168.0.0/16 (síť má kratší masku než je implicitní pro danou třídu) (maximální počet skoků a možnosti redistribuce cest mají stejné)
- 5) Čím se liší RIPv2 od RIPv1?
 - a) RIPv2 obsahuje ve svých aktualizacích i masku podsítě
- 6) Na směrovači B, kterému jsou přímo připojené dvě sítě 192.168.1.0/30 a 192.168.1.4/30, se po zapnutí příkazu „**debug ip rip events**“ objevil následující výpis:

```
B#debug ip rip events
- vynecháno -
RIP ignored v2 packet from 192.168.1.1 (illegal version)
RIP ignored v2 packet from 192.168.1.6 (illegal version)
```

Co je pravděpodobnou příčinou tohoto hlášení?

- a) Tento směrovač B má spuštěnou jinou verzi RIP než oba jeho sousedi. (=> B má spuštěnou verzi 1 a sousední směrovače mají spuštěnou verzi 2 protokolu RIP)
- 7) Máte tři směrovače A, B a C zapojené v linii za sebou a k nim přilehlé následující čtyři sítě: 192.168.1.32/27, 192.168.1.64/30, 192.168.2.0/24 a 192.168.3.0/24. Na všech třech směrovačích je spuštěn směrovací protokol RIPv2. Proč na směrovači C vidíte pouze síť

192.168.1.0/24 a nikoliv jednotlivé dvě sítě 192.168.1.32/27 a 192.168.1.64/30?

- a) RIPv2 má implicitně spuštěnou automatickou summarizaci do plné třídy (z důvodu, aby byl v této věci kompatibilní s RIPv1)
- 8) Na směrovači máte následující výpis: Co z něj lze vyčíst?

```
R2#debug ip rip events
RIP event debugging is on
R2#RIP: received v2 update from 172.16.2.126 on FastEthernet1/0
    172.16.2.128/25 via 0.0.0.0 in 1 hops
    172.16.3.0/24 via 0.0.0.0 in 2 hops
    192.168.3.0/24 via 0.0.0.0 in 2 hops
    192.168.4.0/24 via 0.0.0.0 in 3 hops
```

- a) V aktualizacích směrovací protokol předává masky (přijímá RIP verzi 2).

Kapitola 8 - Směrovací tabulka – bližší pohled

V této kapitole se naučíme:

- Popsat jednotlivé typy cest, které je možné nalézt ve směrovací tabulce
- Popsat postup vyhledání cesty do cílové sítě
- Popsat chování procesu směrování ve směrovaných sítích

V předchozích kapitolách jsme prozkoumávali směrovací tabulku pomocí příkazu *show ip route*. Viděli jsme jak jsou přidávány a vymazávány ze směrovací tabulky přímo připojené statické i dynamické cesty.

Pro správce sítě je při odstraňování síťových problémů důležité znát směrovací tabulku do hloubky. Pochopení struktury směrovací tabulky i vyhledávacího procesu v ní vám pomohou diagnostikovat jakékoli problémy směrovací tabulky bez ohledu na Váš stupeň znalosti konkrétního směrovacího protokolu. Například se můžete setkat se situací, kdy jsou ve směrovací tabulce všechny trasy, které byste očekávali, že uvidíte, ale paket není očekávaným způsobem přeposílan. Znalost postupu vyhledávání cílové IP adresy pro paket vám umožní určit, zda je paket přeposílan dle očekávání, nebo zda a proč je přeposílan jinam, nebo zda byl zahozen.

V této kapitole se podíváme na směrovací tabulky trochu blíže. První část kapitoly se zaměřuje na strukturu směrovací tabulky Cisco pro IP. Budeme zkoumat formát směrovací tabulky a dozvíme se trasách úrovně 1 a úrovně 2. Druhá část kapitoly analyzuje proces prohledávání směrovací tabulky. Budeme diskutovat třídní směrovací chování, stejně tak jako beztřídní směrovací chování, která používají příkazy *no ip classless* a *ip classless*.

Mnoho podrobností o struktuře a vyhledávacím procesu ve směrovací tabulce IP Cisco bylo z této kapitoly vypuštěno. Máte-li zájem o četbu většího množství informací o tomto tématu a vnitřním fungování Cisco IOS, které se týká směrování, podívejte se do knihy *Cisco IP Routing* od Alexe Zinina. Poznámka: Uvedená kniha ale není knihou pro začátečníky ve směrovacích protokolech, je to důkladné prozkoumání protokolů, procesů a algoritmů používaných operačním systémem Cisco IOS.

Podrobnější pohled na směrování

Terminologie: *route* česky směr, cesta, trasa (překlad kolísá dle aktuálního kontextu).

Směrovací tabulka je databáze, která má hierarchickou strukturu. Důvodem je rychlý postup vyhledávání v ní (*speed lookup process*). Tato struktura má několik úrovní, pro jednoduchost budeme diskutovat pouze úroveň 1 a 2.

Její obsah na směrovači zobrazíme příkazem: **show ip route**

Poznámka: Hierarchie směrovací tabulky v Cisco IOS byla původně implementována s třídním směrovacím schématem. Přestože směrovací tabulka obsahuje oboje třídní i beztřídní adresy, je její celková struktura stále vybudována na tomto třídním schématu.

Ve směrovací tabulce mohou být směry trojího druhu (podle druhu zdroje informací pro tuto řádku):

- přilehlý, přímo připojený (*directly connected*) – kód C,
- statický (*static*) – kód S,
- dynamický (*dynamic*) – kódy R(RIP), D(EIGRP), O(OSPF),

Přidání přilehlého, přímo připojeného, směru, cesty (*connected route*) do směrovací tabulky:

- nastavte IP adresu a masku rozhraní,
- administrativně zapněte rozhraní příkazem „no shutdown“,
- přilehlý (přímo připojený) směr (cesta) je okamžitě přidán do směrovací tabulky,
- vyzkoušejte si příkaz „debug ip routing“, abyste to viděli přímo v akci.

Směry úrovně 1 (Level 1 routes)

Úrovně jedna (*Level 1*) je **směr s maskou podsítě (subnet mask) rovnou nebo menší než (equal to or less than) implicitní maska plné třídy (classful mask)**

Trasa úrovně 1 může mít 3 různé typy (funkce):

- 192.168.1.0/24 **síťový směr (network route)** (rozumí se směr do třídní sítě). /24 (= *classful mask*). Má implicitní třídní masku. Síťový směr může zároveň být i tzv. Rodičovský směr = úrovně 1 (viz dále).
- 192.168.128.0/20 **nadsíťový směr (supernet route)**. Má kratší než implicitní třídní masku.
- 0.0.0.0/0 **implicitní směr (Default route)**. Má masku /0.

Trasa první úrovně je do směrovací tabulky přidána okamžitě po zapnutí rozhraní do přilehlé (přímo připojené) sítě příkazem no shutdown.

Příklad: určete, zda je směr do uvedené sítě **úrovně 1**:

192.168.1.0/24	ano (network route)
192.168.1.32/27	ne
192.168.4.0/22	ano (supernet route)
0.0.0.0/0	ano (default route)

Trasa první úrovně může být ultimátní (ale také být nemusí).

Definitivní, ultimátní trasa (Ultimate Route)

Úroveň 1 může být dále navíc definována jako **definitivní, nepominutelná konečná trasa (ultimate route)**, což je směr, který zahrnuje buď:

- IP adresu dalšího přeskoku (*next-hop*) (odkaz na jinou cestu přes přilehlou síť),
- a/nebo výstupní, odchozí rozhraní (*exit interface*).

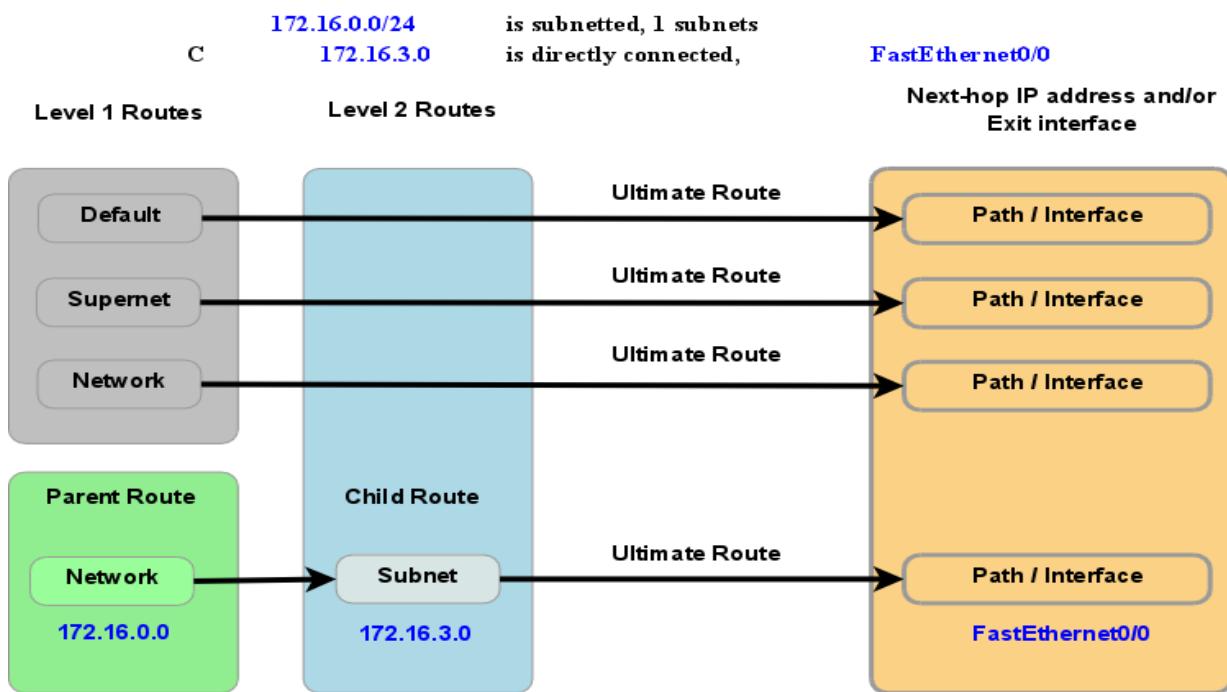
C 192.168.1.0/24 is directly connected, **Serial0/0/1**

Pokud je ve skupině sítí definována podsíť jsou řádky směrovací tabulky navíc ještě rozděleny na typ rodič a potomek.

Rodič a potomek (Parent and child)

- **Trasa typu rodič (parent route) = trasa úrovně 1 (level 1 route).** Nemá žádnou informaci o výstupním směru (*no exit information*), next-hop nebo odchozí rozhraní. Je automaticky přidán, když je do směrovací tabulky přidána podsíť, tj. když je přidán směr typu potomek (*added when child route is added*).
- **Trasa typu potomek (child route) = trasa úrovně 2 (level 2 route).** Je to podsíť sítě v plné třídě (*subnet of classful network*). Může být také považována za definitivní, konečnou, ultimátní, protože obsahuje odchozí rozhraní a/nebo next-hop.

Směrovací tabulka: vztah rodič/potomek
Routing Table: Parent/Child Relationship



Obsah směrovací tabulky

Příklad (podsíťování třídní sítě, adresní struktura CIDR)

```
10.0.0.0/30 is subnetted, 2 subnets
R      10.10.10.0 [120/1] via 10.10.10.5, 00:00:28, Serial0/0/0
C      10.10.10.4 is directly connected, Serial0/0/0
```

Směr (Route)	Rodič (Parent)	Potomek (Child)
10.0.0.0/16	X	
10.10.10.0/30		X
10.10.10.4/30		X

Poznámka: pamatujete, že hierarchie směrovací tabulky v Cisco IOS má třídní směrovací schéma. Směr úrovně 1 je třídní síťová adresa směru podsítě. V tomto případě dokonce i když je zdrojem směru do podsítě beztřídní směrovací protokol.

Výstup z příkazu (Command Output)	Popis (Description)
10.0.0.0	Třídní síť (rodičovská)
/30	Maska podsítě pro směry typu potomek
is subnetted, 2 subnets	Rodič se dvěma směry typu potomek.
R	Zdrojem směru je RIP
10.10.10.0	První cesta typu potomek
120	Hodnota administrativní vzdálenosti pro cesty jejichž zdrojem je RIP
1	Metrika RIP, 1 hop
via 10.10.10.5	IP adresa Next-hop pro tuto cestu typu potomek
00:00:28	Doba od poslední aktualizace ze souseda
Serial0/0/0	Odchozí rozhraní pro první cestu typu potomek (trasa je <i>ultimate</i>)
C	Zdrojem je připojená, přilehlá cesta (<i>route</i>)
10.10.10.4	Druhá cesta typu potomek
Serial0/0/0	Odchozí rozhraní pro druhou cestu typu potomek (<i>ultimate route</i>)

Příklad (beztřídní adresní struktura VLSM)

172.16.0.0/16 is variably subnetted, 3 subnets, 3 masks

```
R      172.16.0.0/18 [120/1] via 10.10.10.5, 00:00:28, Serial0/0/0
C      172.16.68.0/22 is directly connected, FastEthernet0/0
C      172.16.72.0/23 is directly connected, FastEthernet0/1
```

Směr (Route)	Rodič (Parent)	Potomek (Child)
172.16.0.0/16	X	
172.16.0.0/18		X
172.16.68.0/22		X
172.16.72.0/23		X

Bez ohledu na adresní schéma použité sítí (třídní nebo beztřídní), směrovací tabulka bude používat třídní schéma.

Výstup z příkazu (Command Output)	Popis (Description)
172.16.0.0	Třídní rodičovský směr (<i>Classful parent route</i>)
/16	Třídní maska (<i>Classful mask</i>)
is variably subnetted,	Směry potomků mají různé masky
3 subnets, 3 masks	Počet podsítí a masek pro tento rodičovský směr
R	Zdrojem směru je RIP
172.16.0.0	První směr typu potomek
/18	Maska pro první směr typu potomek
120	Hodnota AD směry jejichž zdrojem je RIP
1	Metrika RIP, 1 hop
via 10.10.10.5	IP adresa Next-hop pro první směr typu potomek
00:00:28	Doba od poslední aktualizace ze souseda
Serial0/0/0	Odchozí rozhraní pro první cestu typu potomek (je <i>ultimate</i>)
C	Zdrojem je přilehlá cesta (<i>source is connected route</i>)
172.16.68.0	Druhá cesta typu potomek
/22	Maska pro druhou cestu typu potomek
FastEthernet0/0	Odchozí rozhraní pro druhou cestu typu potomek

Rozdíl v obsahu směrovací tabulky podle typu sítě

Typ sítě	Pro trasu typu rodič je zobrazena maska v plné třídě	Pro trasu typu rodič je uveden termín „variably subnetted“	Pro trasu typu rodič je uveden počet různých masek pro podřízené řádky typu potomek	V každé řádce typu potomek je maska podsítě
Třídní	Ne	Ne	Ne	Ne
Beztrídní (VLSM)	Ano	Ano	Ano	Ano

Postup vyhledání nejlepšího směru

1. Směrovač prohledá řádky směrovací tabulky se směry úrovně 1, zahrnující třídní směry a nadsíťové směry, pro nejlepší spárování s cílovou adresou IP paketu.
 - 1.1. Jestliže je nejlepším spárováním ultimátní směr úrovně 1, => plná síť v plné třídě, nad-síť nebo implicitní cesta, je tento směr použit pro přeposlání paketu.
 - 1.2. Jestliže nejlepší spárování je rodičovský směr úrovně 1, provede se krok 2.
2. Směrovač prohledá směry typu potomek (podsítové směry) pro příslušný rodičovský směr pro nejlepší spárování.
 - 2.1. Jestliže je zde odpovídající směr typu potomek úrovně 2, bude tato podsíť použita pro přeposlání paketu.
 - 2.2. Jestliže zde není žádný odpovídající řádek úrovně 2, provede se krok 3.
3. Má směrovač implementované třídní nebo beztrídní směrování (směrovací chování = způsob prohledávání směrovací tabulky)?
 - 3.1. Třídní směrování (*classful routing behavior*): pokud je funkční třídní směrování, ukončí se proces vyhledávání a **odhodí paket (bez ohledu na případné nastavení implicitní cesty)**.
 - 3.2. Beztrídní směrování (*classless routing behavior*): pokud je funkční beztrídní směrování, pokračuje hledání nadsíťového směru úrovně 1 ve směrovací tabulce pro spárování, včetně implicitní cesty, pokud je nastavena.
4. Pokud je zde nyní kratší spárování s nadsíťovou nebo implicitní cestou úrovně 1, směrovač použije tento směr pro přeposlání paketu.
5. Jestliže zde **není spárování s žádnou cestou** ve směrovací tabulce, směrovač tento **paket odhodí**.

Později budeme třídní a beztrídní chování směrování diskutovat podrobněji.

Poznámka: směr, který odkazuje pouze na IP adresu následujícího skoku (next-hop) a nemá odchozí rozhraní, musí být převeden na směr s odchozím rozhraním, pomocí rekurzivního vyhledání ve směrovací tabulce.

Poznámka k terminologii: je třeba si uvědomit, že rozlišujeme následující kategorie:

1. systém adresace sítě

- 1.1. třídní – pouze implicitní masky,
- 1.2. beztřídní
 - 1.2.1.podsítě pouze sítě v plné třídě, všechny masky stejné - CIDR
 - 1.2.2.podsítě podsítí, masky mohou být různé – VLSM
- 2. směrovací protokol
 - 2.1. třídní – aktualizace neobsahuje masku
 - 2.2. beztřídní – aktualizace obsahuje masku
- 3. způsob prohledávání směrovací tabulky, směrovací chování
 - 3.1. třídní – v případě nenalezení přesného spárování u cesty do podsítě typu potomek úrovně 2, je paket odhozen
 - 3.2. beztřídní - v případě nenalezení přesného spárování u cesty do podsítě typu potomek úrovně 2 je dále prohledávána 1. úroveň na supersíť nebo implicitní cestu.

Nejdelší spárování

Nejlepší spárování (*the best match*) termín, který byl použit v předchozím popisu, je také někdy uváděn jako (*a.k.a = also known as*) **nejdelší spárování** (*longest match*).

Cílová IP adresa paketu	172.16.0.10	10101100.00010000.00000000.00001010
Route 1	172.16.0.0/12	10101100.00010000.00000000.00000000
Route 2	172.16.0.0/18	10101100.00010000.00000000.00000000
Route 3	172.16.0.0/26	10101100.00010000.00000000.00000000

Příklad

Použijte obsah části výpisu směrovací tabulky na směrovači C, na kterém je IOS verze 12.3. Směrovač přijal paket s cílovou IP adresou 172.16.1.130. Který nabídnutý směr (cestu) směrovač použije pro přeposlání paketu a proč?

```
C#show ip route
<vynecháno>
Gateway of last resort is 0.0.0.0 to network 0.0.0.0

S    172.16.0.0/13 is directly connected, FastEthernet0/0
     172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
R      172.16.0.0/24 [120/3] via 172.16.1.1, 00:00:03, FastEthernet0/0
C      172.16.1.0/25 is directly connected, FastEthernet0/0
     172.17.0.0/25 is subnetted, 1 subnets
C      172.17.1.0 is directly connected, FastEthernet0/1
S*   0.0.0.0/0 is directly connected, FastEthernet0/0
C#
```

- a) 172.16.1.0/25
- b) 172.16.0.0/16
- c) 172.16.0.0/24
- d) 172.16.0.0/13
- e) implicitní směr (cesta)
- f) nic, paket bude zahozen

Jak postupovat při řešení:

Protože je použita verze IOS 12.3, je tam implicitní nastavení beztrídního způsobu prohledávání směrovací tabulky *ip classless*. Prohledávají se nejprve rodičovské směry a to podle délky masky sestupně. Nejprve se prohledávají potomci rodičovského směru 172.16.0.0/16 a protože žádný nevyhovuje, začnou se prohledávat další rodičovské směry s kratší maskou. Vyhovuje potom ultimátní cesta 172.16.0.0/13 (d).

Pokud by bylo dodatečně nastaven třídní způsob prohledávání směrovací tabulky *no ip classless*, tak by byl daný paket zahozen (protože by se v rámci rodičovského směru 172.16.0.0/16 nenašel odpovídající potomek) a prohledávání by se v tomto případě již nevracelo na rodičovskou úroveň.

Příkazy pro kapitolu 8, Směrovací tabulka – bližší pohled

Příkaz (Command)	Popis (Description)
Router(config)# no ip classless	Nastavuje chování směrovače při prohledávání směrovací tabulky jako třídní (<i>classful</i>). Bylo implicitní v IOS ve verzi před Release 11.3.
Router(config)# ip classless	Nastavuje chování směrovače při prohledávání směrovací tabulky jako beztrídní (<i>classless</i>). Je implicitní v IOS ve verzi Release 11.3 a později.

Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Které charakteristiky mohou být použity k určení, zda je směr ultimátní?
 - a) Směr obsahuje odchozí rozhraní (*exit interface*).
- 2) Směrovač R1 je nakonfigurován s příkazy: R1(config)#**ip classless** a R1(config)#**ip route 0.0.0.0 0.0.0.0 serial0/0/0**. Co udělá R1 s paketem, který bude spárovaný s rodičovským směrem, ale v rámci něho už nebude spárovaný se žádným směrem typu potomek?
 - a) Pošle paket přes implicitní cestu (implicitním směrem).
- 3) Která akce na směrovači umožní třídní chování při směrování (způsob prohledávání směrovací tabulky)?
 - a) Vydání příkazu **no ip classless**.

- 4) Během procesu hledání směru při směrování, co ustavuje preferovaný směr?
 - a) Nejdelší spárování bitů zleva.
- 5) Jestliže je paket spárovaný s (= odpovídá) cestou 1. úrovně typu rodič, co je v procesu směrování druhý krok?
 - a) Směrovač bude hledat směr 2. úrovně typu potomek s odchozím rozhraním.
- 6) Co dělají příkazy **ip classless** a **no ip classless**?
 - a) Určují beztřídní nebo třídní způsob prohledávání směrovací tabulky během směrování.
- 7) Směrovač R1 je nakonfigurován s příkazy: R1(config)#**no ip classless** a R1(config)#**ip route 0.0.0.0 0.0.0.0 serial0/0/0**. Co udělá R1 s paketem, který bude spárovaný s rodičovským směrem, ale v rámci něho už nebude spárovaný se žádným směrem typu potomek?
 - a) Zahodí paket.

Kapitola 9 - Protokol EIGRP

V této kapitole se naučíme:

- Popsat předchůdce a historii EIGRP
- Popsat funkce a činnost EIGRP
- Prozkoumat základní konfigurační příkazy EIGRP a určit jejich účel
- Vypočítat složenou metriku používanou EIGRP
- Popsat koncept a činnost konvergenčního algoritmu DUAL
- Popsat použití dalších konfiguračních příkazů EIGRP

EIGRP (*Enhanced Interior Gateway Routing Protocol*) je směrovací protokol typu vektor vzdálenosti, beztrídní směrovací protokol, který byl uvolněn v roce 1992 spolu s IOS 9.21. Jak už jeho název napovídá, EIGRP, je vylepšení protokolu Cisco IGRP (*Interior Gateway Routing Protocol*). Oba dva jsou proprietární protokoly Cisco a pracují pouze na směrovačích Cisco.

Hlavním cílem společnosti Cisco při vývoji EIGRP bylo vytvořit beztrídní verzi IGRP. EIGRP obsahuje několik funkcí, které se běžně nevyskytují v jiných směrovacích protokolech typu vektor vzdálenosti jako jsou RIP (RIPv1 a RIPv2) a IGRP. Mezi tyto funkce patří:

- spolehlivý transportní (L4) protokol RTP (*Reliable Transport Protocol*),
- omezené aktualizace,
- konvergentní algoritmus DUAL (*Diffusing Update Algorithm*),
- vytváření vztahů sousedství (*adjacencies*),
- tabulky Sousedů (*neighbor*) a Topologickou (*topology*).

Přestože EIGRP může působit jako směrovací protokol typu stavu linky, je to stále ještě směrovací protokol typu vektor vzdálenosti.

Poznámka: Pro definici EIGRP je někdy používáno termínu *hybridní směrovací protokol*. Nicméně tento termín je zavádějící, protože EIGRP není kříženec mezi směrovacími protokoly typu vektor vzdálenosti a typu stav linky - je to je pouze směrovací protokol typu vektor vzdálenosti. Proto společnost Cisco při odkazu na EIGRP již tento termín nadále nepoužívá.

V této kapitole se dozvítíte, jak nastavit EIGRP a ověřit si konfiguraci s novými příkazy show. Naučíte se také vzorec, který EIGRP používá pro výpočet složené (kompozitní) metriky.

Jedinečný pro EIGRP je jeho spolehlivý transportní protokol RTP (*Reliable Transport Protocol*), který poskytuje spolehlivé i nespolehlivé doručování paketů EIGRP. Kromě toho, EIGRP vytváří vztahy sousedství (*adjacency*) s přímo připojenými směrovači, které mají též spuštěný EIGRP. Součeské vztahy se používají ke sledování stavu těchto sousedů. RTP a sledování vztahů sousedství (*adjacencies*) připravují půdu pro tahouna EIGRP – algoritmus DUAL (*Diffusing Update Algorithm*).

Vzhledem k tomu, že výpočetní motor, který pohání EIGRP, DUAL sídlí v samotném centru směrovacího protokolu, zaručuje to v celé směrovací doméně cesty bez smyček a záložní cesty. Naučíte se, jak přesně DUAL zvolí trasy k instalaci do směrovací tabulky, a to, co DUAL dělá s potenciální-

mi záložními trasami.

Stejně jako RIPv2, EIGRP může pracovat s třídním nebo beztřídním chováním směrování. Naučíte se, jak vypnout automatické summarizace a pak, jak ručně summarizovat síť, aby se zmenšila velikost směrovacích tabulek. Nakonec se naučíte, jak používat implicitní směrování s EIGRP.

Úvod do EIGRP

EIGRP – vylepšený protokol typu vektor vzdálenosti

Ačkoli je EIGRP popisován jako vylepšení směrovacího protokolu typu vektor vzdálenosti, je to stále ještě směrovací protokol typu vektor vzdálenosti. To někdy může být zdrojem nejasnosti. Abychom ocenili vylepšení EIGRP a odstranili jakékoli nedorozumění, musíme se nejprve podívat na jeho předchůdce, IGRP.

Kořeny EIGRP: IGRP

Společnost Cisco vyvinula svůj proprietární protokol IGRP v roce 1985 v reakci na některá omezení RIPv1 zahrnující použití počtu přeskoků jako metriky a maximální velikosti sítě 15 přeskoků.

Místo počtu přeskoků používají jak IGRP tak i EIGRP jako (složenou, kompozitní) metriku šířku pásma, zpoždění, spolehlivost a zatížení. Ve se výchozím nastavení oba směrovací protokoly používají pouze šířky pásma a zpoždění. Nicméně, protože IGRP je třídní směrovací protokol, který používá Bellman-Fordův algoritmus a periodické aktualizace, je jeho využitelnost v mnoha dnešních sítích omezená.

Proto společnost Cisco vylepšila IGRP s novým algoritmem DUAL a dalšími funkcemi. Příkazy pro IGRP i EIGRP jsou podobné a v mnoha případech totožné. To umožňuje snadnou migraci z IGRP na EIGRP. Společnost Cisco přerušila podporu IGRP počínaje IOS verze 12.2(13)T a 12.2(R1s4)S.

Přestože budou podrobněji popsány v celé této kapitole, dovolte probrat některé z rozdílů mezi tradičním směrovacím protokolem typu vektor vzdálenosti jako je RIP i IGRP a mezi vylepšeným směrovacím protokolem typu vektor vzdálenosti EIGRP.

Následující tabulka shrnuje hlavní rozdíly mezi tradičním směrovacím protokolem typu vektor vzdálenosti jako je RIP a mezi vylepšeným směrovacím protokolem typu vektor vzdálenosti EIGRP.

Přehled činnosti směrovacích protokolů	
Tradiční protokol typu vektor vzdálenosti	Vylepšený protokol - EIGRP
Používá algoritmus Bellman-Ford neboli Ford-Fulkerson	Používá rozprostřený aktualizační algoritmus <i>Diffusing Update Algorithm</i> (DUAL)
Sleduje stáří záznamů ve směrovací tabulce a používá periodické aktualizace	Nesleduje stáří záznamů (<i>not age out</i>) ve směrovací tabulce a nepoužívá periodické aktualizace
Eviduje pouze nejlepší trasy, nejlepší cesty do cílové sítě	Odděleně udržuje tabulku topologie (<i>topology table</i>) obsahující nejlepší trasu a všechny záložní cesty neobsahující smyčky (<i>loop free backup</i>)

	<i>paths), nezávisle na směrovací tabulce</i>
Když se směr stane nedostupným, musí směrovač počkat na novou aktualizaci	Když se směr stane nedostupným, DUAL použije záložní cestu pokud existuje v topologické tabulce
Pomalejší konvergence z důvodu použití zadržovacích časovačů	Rychlejší konvergence z důvodu nepoužití zadržovacích časovačů a použití systému koordinovaných výpočtů tras (<i>coordinated route calculations</i>)

Algoritmus

Všechny tradiční směrovací protokoly typu vektor vzdálenosti používají některou variantu algoritmu Bellman-Ford či Ford-Fulkerson. Tyto protokoly, jako jsou RIP a IGRP, sledují stáří jednotlivých řádek směrovací tabulky a proto je nutné pravidelně posílat aktualizace směrovací tabulky.

EIGRP používá aktualizační algoritmus **DUAL (Diffusing Update Algorithm)**. Ačkoli je EIGRP stále ještě směrovacím protokolem typu vektor vzdálenosti, implementuje s algoritmem DUAL funkce, které nejsou v tradičních směrovacích protokolech typu vektor vzdálenosti. EIGRP neposílá pravidelné aktualizace a nesleduje stáří řádky tras ve směrovací tabulce. Místo toho EIGRP používá jednoduchý protokol Hello pro monitorování stavu spojení se svými sousedy. Pouze změny ve směrovací informace, jako je nová linka nebo že se linka stala nedostupnou, způsobí, že nastane aktualizace. Směrovací aktualizace EIGRP jsou stále vektory vzdáleností předávané přímo připojeným sousedům.

Stanovení cesty

Tradiční směrovací protokoly typu vektor vzdálenosti jako RIP a IGRP sledují pouze preferované trasy, nejlepší cestu k cílové síti. Pokud přestane být tato trasa k dispozici, směrovač čeká na další směrovací aktualizaci s cestou k této vzdálené síti.

Algoritmus DUAL v EIGRP udržuje oddeleně od směrovací tabulky tabulku topologie, která obsahuje jak nejlepší cestu k cílové síti tak všechny záložní cesty, které DUAL určil jako neobsahující smyčky (*loop-free*). *Loop-free* znamená, že soused nemá cestu do cílové sítě, která prochází přes tento router.

Později v této kapitole uvidíte, že trasa, která bude algoritmem DUAL považována za platnou záložní cestu bez smyček, musí splňovat požadavek známý jako podmínka proveditelnosti. Jakákoli záložní cesta, která splňuje tuto podmínu má zaručeno, že je bez smyček (*loop-free*). Vzhledem k tomu, že EIGRP je směrovací protokol typu vektor vzdálenosti, je možné, že mohou existovat záložní cesty k cílové síti neobsahující smyčky, které nesplňují podmínu proveditelnosti. Tyto cesty proto nejsou zahrnuty v tabulce topologie jako platná záložní cesta bez smyček určená algoritmem DUAL.

Jestliže se trasa stane nedostupnou, bude DUAL hledat ve své topologické tabulce platnou záložní cestu. Pokud existuje, tak se tato trasa okamžitě zapíše do směrovací tabulky. V případě že neexistuje, DUAL provádí proces zjišťování sítí, zda tam náhodou není záložní cesta, která nesplňuje požadavek podmínky proveditelnosti. Tento proces se diskutuje důkladněji později v této kapitole.

Konvergence

Tradiční směrovací protokoly typu vektor vzdálenosti jako RIP a IGRP používají periodické aktualizace. Vzhledem k nespolehlivé povaze periodických aktualizací, jsou tradiční směrovací protokoly typu vektor vzdálenosti náchylné ke směrovacím smyčkám a počítání do nekonečna. RIP a IGRP využívají několik mechanismů, které pomáhají vyhnout se těmto problémům, včetně zadržovacích časovačů, které způsobují dlouhé doby konvergence.

EIGRP nepoužívá zadržovací časovače. Místo toho je cest bez smyček dosaženo prostřednictvím systému výpočtu tras (rozptylové výpočty), které jsou vykonávány koordinovaným způsobem mezi směrovači. Detail toho, jak se to provádí, je nad rámec tohoto kurzu, ale výsledkem je rychlejší konvergence než u tradičních směrovacích protokolů typu vektor vzdálenosti.

Formát zprávy EIGRP

Poznámka: V následující diskusi zpráv EIGRP je mnoho políček jdoucích nad rámcem tohoto kurzu. Jsou zobrazena všechna pole, aby se poskytl přesný obraz formátu zprávy EIGRP. Avšak jsou diskutována pouze pole relevantní pro uchazeče CCNA.

Každá zpráva EIGRP obsahuje záhlaví. Důležitá pro naši diskusi jsou políčko Opcode a políčko číslo autonomního systému. Opcode specifikuje typ paketu EIGRP:

- Aktualizace
- Dotaz
- Odpověď na dotaz
- Kontaktní paket

Číslo Autonomního systému (AS) určuje proces směrování EIGRP. Na rozdíl od RIP mohou směrovače Cisco provozovat více instancí EIGRP. Číslo AS slouží k odlišení vícero instancí EIGRP od sebe.

Zapouzdření zprávy protokolu EIGRP

Záhlaví linkové vrstvy	Záhlaví paketu IP	Záhlaví paketu EIGRP	Typy Type/Length/Value (TLV)
Rámec linkové vrstvy			
Zdrojová MAC adresa = adresa vysílajícího rozhraní Cílová MAC adresa = Multicast: 01-00-5E-00-00-0A			
	Paket IP Zdrojová IP adresa = adresa vysílajícího rozhraní Cílová IP adresa = Multicast: 224.0.0.10 Protokol = 88 pro EIGRP	Záhlaví paketu EIGRP Opcode pro typ paketu EIGRP Číslo Autonomního systému (AS)	
			Typy TVL (pouze výběrový seznam): 0x0001 - Parametry EIGRP 0x0102 - IP trasy interní 0x0103 - IP trasy externí

EIGRP obsahuje několik funkcí, které běžně nejsou k nalezení u jiných směrovacích protokolů **typu vektor vzdálenosti** jako je RIP (RIPv1 a RIPv2) a IGRP. Tyto funkce zahrnují:

- Spolehlivý transportní protokol (L4) *Reliable Transport Protocol (RTP)* – potvrzovaná i ne-potvrzovaná (datagramová) služba na transportní vrstvě.
- Částečné omezené aktualizace (*Partial Bounded Updates*) – aktualizace obsahují pouze změny topologie a jsou zasílány pouze směrovačům, kterých se týkají,
- Difuzní algoritmus aktualizací - *Diffusing Update Algorithm (DUAL)* – umožňuje mít připravenou předem vypočtenou záložní cestu při výpadku linky bez čekání na další aktualizaci => **rychlá konvergence (= synchronizace směrovacích tabulek do konzistentního stavu)**
- Vytváření vztahů sousedství (*Establishing Adjacencies*) mezi přilehlými směrovači ve stejné směrovací doméně (AS).
- Tabulka sousedů a tabulka topologie (*Neighbor and Topology Tables*). Tabulka topologie obsahuje tzv. přípustné následníky (*feasible successors*) = záložní cesty. Tabulka sousedů obsahuje přilehlé směrovače ve stavu sousedství.

Přestože EIGRP může působit dojmem jako směrovací protokol typ stav linky, je to stále směrovací protokol typu vektor vzdálenosti.

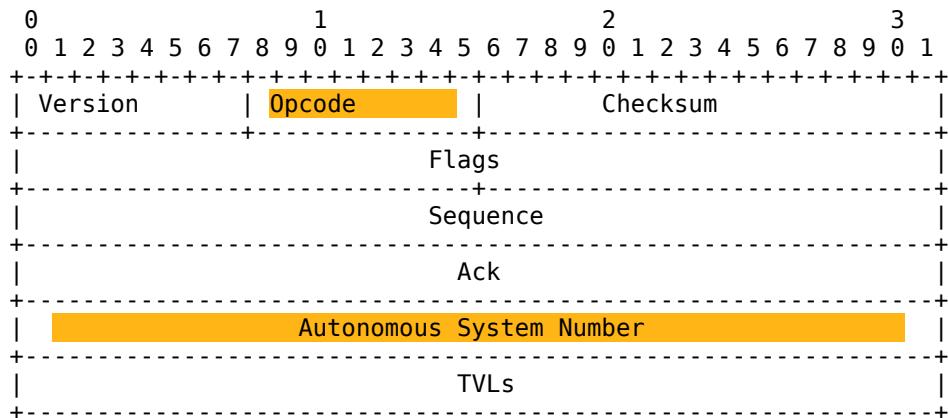
Poznámka: Pro definici EIGRP býval někdy použit termín hybridní směrovací protokol. Nicméně tento termín je matoucí, protože EIGRP je výhradně protokol typu vektor vzdálenosti. Z tohoto důvodu Cisco již nadále nepoužívá tento termín v odkazu na EIGRP.

EIGRP používá a udržuje pro svoji činnost **3 tabulky**:

- **směrovací (routing)** – obsahuje pouze nejlepší cesty (*successor*) (jednu nebo několik se stejnou nejnižší metrikou (*feasible distance*)) do cílové sítě použité pro směrování => v algoritmu DUAL jsou tzv. *Successor route* - primární cesty (*primary route*) vybrané pomocí DUAL pro směrování – zařazení do směrovací tabulky. Její obsah je určen pomocí DUAL z následujících dvou tabulek:
- **topologie (topology)** – obsahuje všechny zjištěné (naučené) směry (nejlepší směr (*successor route*), záložní směr (*feasible successor route*) i všechny ostatní) do všech cílových sítí (obsahuje tedy celou topologii sítě ve stejné směrovací doméně),
- **sousedů (neighbor)** – obsahuje sousední směrovače, kteří si vzájemně vyměňují aktualizace v EIGRP (směrovače jsou **ve vztahu přilehlého sousedství (adjacent routers)** na přímo připojené (přilehlé) síti ve stejném autonomním systému (AS)). (Směrovač má své informace včetně hodnoty metriky trasy pouze od přilehlých sousedů, proto je to směrovací protokolu používající algoritmus vektoru vzdálenosti.)

Přesný formát těchto tabulek je závislý na **směrovaném protokolu** a je včetně jejich obsahu veden odděleně (pro směrované protokoly L3: IP, IPX, AppleTalk) = tzv. modul závislý na protokolu (*Protocol Dependent Module, PDM*).

EIGRP



Opcode: EIGRP Packet Type: Update (1), Query (3), Reply (4), Hello (5).
 Autonomous System Number: ID for EIGRP routing process

Typy paketů EIGRP (typ je určen hodnotou pole *Opcode*):

- Aktualizace (*Update*) – obsahují pouze změny, nejsou periodické, vysílané unicast/multicast (podle počtu adresátů), potvrzované. Aktualizace jsou:
 - vázané, omezené (*bounded*) – aktualizace jsou posílány (propagovány) pouze na směrovače, na které má tato změna vliv,
 - částečné (*partial*) – aktualizace obsahují pouze změny topologie (týká se to též změny metriky).
- Dotaz (*Query*) - hledání sítí, další úkoly, unicast nebo multicast, potvrzovaná,
- Odpověď na dotaz (*Reply*) - odpověď, vždy unicast, potvrzovaná,
- Kontaktní paket (*Hello*) - hledání, identifikace a verifikace sousedních směrovačů (EIGRP ve stejném autonomním systému), multicast, datagram (periodické – 5 sekund u Ethernetu). (Protože aktualizace nejsou úplné (= nikoliv celá směrovací tabulka) a neposílají se všem směrovačům, musí být pro kontrolu toho že všechny směrovače jsou „naživu“ vytvořen a udržován **vztah sousedství mezi směrovači** (*adjacency*), které si vyměňují informace, vztah sousedství se vytváří a udržuje právě pomocí těchto kontaktních paketů.)

Administrativní vzdálenosti

- Interní EIGRP = 90,
- EIGRP agregovaný směr (*summary route*) = 5,
- External EIGRP (redistribuce z jiných směrovacích protokolů nebo z EIGRP v jiném autonomním systému) = 170.

Metrika

Kompozitní (*composite*), složená, metrika u EIGRP:

Default metric = [K1*bandwidth²³ + K3*delay] (implicitní formule)

Metric = [K1*bandwidth + (K2*bandwidth)/(256 – load) + K3*delay] * [K5/(reliability + K4)] (kompletní formule)

Při vypočtu číselné hodnoty metriky se použijí následující hodnoty:

referenční šířka pásma = bandwidth = $256 * (10\ 000\ 000 / \text{nejnižší šířka pásma na trase do cíle})$,

delay = $256 * (\text{součet zpoždění na cestě do cíle}) / 10$.

Nejlepší cesta (s nejmenší metrikou, *feasible distance*) je ta s největší šírkou pásma a s nejmenším zpožděním.

Implicitní hodnoty K:

1. K1 (*bandwidth*) = 1
2. K2 (*load*) = 0
3. K3 (*delay*) = 1
4. K4 (*reliability*) = 0
5. K5 (*reliability*) = 0

Aktuální hodnoty K zobrazí příkaz:

show ip protocols

Změna hodnot K:

Router(config-router) #metric weights tos k1 k2 k3 k4 k5

tos = type of service je vždy nastavena na 0.

```
R2#show ip protocols
```

```
Routing Protocol is "eigrp 100 "
```

```
Outgoing update filter list for all interfaces is not set
```

```
Incoming update filter list for all interfaces is not set
```

```
Default networks flagged in outgoing updates
```

```
Default networks accepted from incoming updates
```

```
EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
```

```
EIGRP maximum hopcount 10024
```

23 Do vzorce se automaticky použije referenční šířka pásma nejpomalejší linky na trase do cílové sítě.
Bandwidth = $256 * 10\ 000\ 000 / \text{bandwidth}$. (Nejlepší cesta je cesta s nejmenší hodnotou metriky.)

24 Implicitní maximální počet přeskoků v EIGRP je roven 100 a lze ho nastavit až na maximálně 224 přeskoků.

Aktuální hodnoty vah metrik EIGRP na konkrétním rozhraní zobrazí:

show interface

```
R2#sh int fa0/0
FastEthernet0/0 is up, line protocol is up (connected)
  Hardware is Lance, address is 0060.2f37.725b (bia 0060.2f37.725b)
  Internet address is 192.168.2.254/24
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set
```

- Metrika **přenosová rychlosť (přenosová kapacita)** (*bandwidth*) je zobrazena v Kbit (kilobitech). Většina sériových rozhraní používá implicitní hodnotu 1,544,000 bps, což je hodnota pro připojení typu T1. Nastavená hodnota může a také nemusí odrážet skutečnou přenosovou rychlosť rozhraní. Můžete ji nastavit v konfiguračním režimu rozhraní.
- **Zpoždění (delay)** je měřítkem doby potřebné pro cestu paketu přes daný směr (*route*). Je to statická hodnota vyjádřená v mikrosekundách (μsec ve výpisech usec). Pro FastEtherenet je to 100 μsec. Pro T1 je to 20 000 μsec.
- **Spolehlivost (reliability, rely)** je měřítkem pravděpodobnosti (*probability*), že linka selže, nebo jak často se na lince vyskytují chyby. Na rozdíl od zpoždění je spolehlivost měřena dynamicky s hodnotou mezi 0 a 255, kde 1 je minimálně spolehlivá linka a 255 je 100% spolehlivá. Je počítána jako průměr za 5 minut, aby se předešlo vlivům náhlých změn četnosti chyb.
- **Zatížení (load)** odráží využití linky síťovým provozem. Zatížení je měřeno dynamicky s hodnotami mezi 0 a 255. Je žádanější nižší hodnota, která indikuje méně zatíženou linku.

Konvergenční algoritmus DUAL

Koncepce algoritmu DUAL

DUAL (*Diffusing Update Algorithm*) je algoritmus používaný EIGRP pro dosažení (primární) nejlepší cesty neobsahující smyčky a dalších záložních cest neobsahujících smyčky (*the best loop-free path and loop-free backup paths*), má rychlou konvergenci – protože záložní cesty má napočítány dopředu a potřebuje malou šířku pásma – používá omezené a částečné aktualizace.

DUAL používá několik termínů:

- Následník (*Successor*) - sousední směrovač na cestě, přes který bude přeposílan (*forward*) paket (nejnižší metrika)
- Přípustná, možná, vzdálenost (*Feasible Distance (FD)*) - nejnižší metrika do cílové sítě (je ve směrovací tabulce aktuálního směrovače i v tabulce síťové topologie)
- Přípustný, možný, následník (*Feasible Successor (FS)*) - soused, který má cestu k cíli neobsahující smyčky (*loop-free*), musí splnit podmínu přípustnosti (*feasibility condition*),

- Inzerovaná vzdálenost - *Reported Distance (RD)* neboli *Advertised Distance (AD)* - vzdálenost souseda k cíli, kterou hlásí soused aktuálnímu směrovači
- Podmínka přípustnosti (*Feasible Condition* neboli *Feasibility Condition (FC)*) - je splněna, pokud sousedova *reported distance* (tj. vzdálenost souseda k cíli, kterou mi hlásí) je menší než moje *Feasible Distance*. Pokud není k dispozici *Feasible Successor* (nesplňuje podmínu přípustnosti), musí se přepočítat celý DUAL

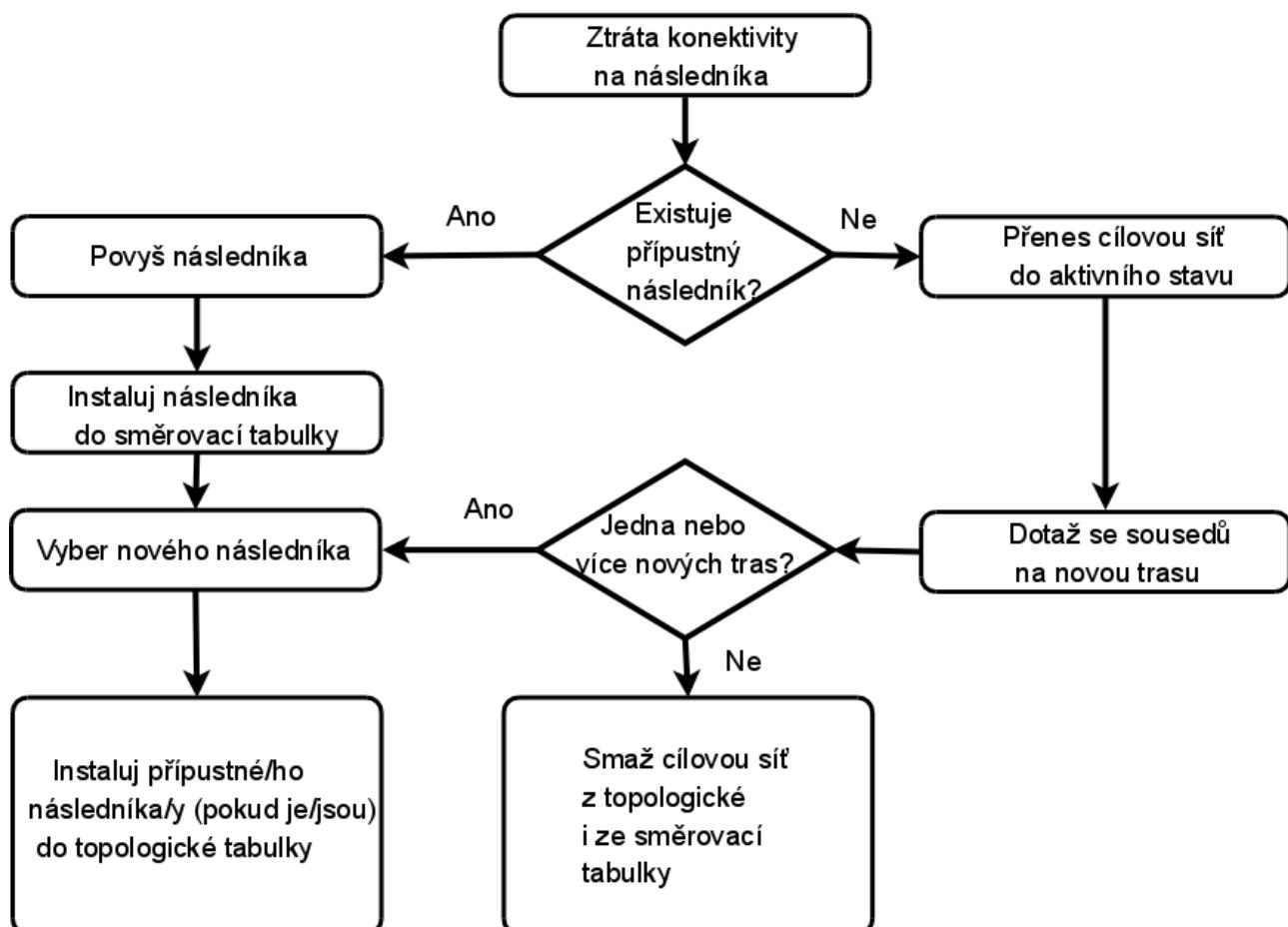
Tyto termíny a koncepty jsou centrem mechanismu předcházení směrovacím smyčkám.

Konečný automat

Konečný automat (*Finite State Machine, FSM*) – je abstraktní automat, nikoliv mechanické zařízení s pohyblivými součástmi. Konečný automat (*FSM*) definuje množinu možných stavů, které někdy mohou nastat, a jaké události jsou příčinou těchto stavů a jaké události jsou důsledkem těchto stavů. (Na rozdíl od logického obvodu, kde výstupní stav záleží na okamžitém vstupním stavu, výstup konečného automatu závisí na celé posloupnosti vstupních stavů.)

Vývojáři používají konečné automaty k popisu jak budou zařízení, počítačové programy nebo směrovací algoritmy, reagovat na určitou konkrétní sadu vstupních událostí.

Konečný automat algoritmu DUAL (DUAL Finite State Machine)



=> U EIGRP (a stejně potom i u OSPF) stav směrování závisí i na postupně provedených změnách nastavení (protože výpočetní algoritmus je konečný automat (FSM). Někdy je tedy nutné, po změnách konfigurace, vymazat tabulky ukládající průběžné stavy. (Resetovat procesy příslušného směrovacího protokolu nebo restartovat směrovače.)

Autonomní systém

Autonomní systém (*Autonomous System, AS*) neboli směrovací doména je oblast, ve které jsou nastaveny stejné zásady směrování do Internetu. Je mu přiděleno 16-ti bitové číslo (1 – 65535). Při konfiguraci EIGRP musí být číslo AS zadáno. Pokud mají směrovací procesy EIGRP jiné číslo AS nekomunikují spolu (pokud není mezi nimi nastavena redistribuce cest). Z tohoto pohledu je tedy vlastně AS číslo, identifikátor, procesu (*process ID*).

Příkazy pro kapitolu 9, EIGRP

Konfigurace EIGRP

Router(config)#router eigrp 100	Zapne proces EIGRP. 100 je číslo autonomního systému, což může být číslo mezi 1 a 65 535.
	Všechny směrovače v tom samém autonomním systému musí používat stejné číslo autonomního systému.
Router(config-router)#network 10.0.0.0	Specifikuje, která síť je inzerována pomocí EIGRP.
Router(config-if)#bandwidth x	Nastaví šířku pásmá (přenosovou rychlosť, kapacitu) tohoto rozhraní na x kilobitů, což EIGRP umožní lepší kalkulaci metriky.
	TIP: Příkaz bandwidth je použit pouze pro výpočet metriky. Nemění skutečný výkon rozhraní.
Router(config-router)#no network 10.0.0.0	Vymaže zadanou síť ze zpracování EIGRP.
Router(config)#no router eigrp 100	Vypne směrovací proces 100.
Router(config-router)#network 10.0.0.0 0.255.255.255	Identifikuje, která rozhraní nebo síť jsou zahrnuty do EIGRP. Rozhraní musí být nakonfigurována a adresami, které spadají do rozsahu určeného pseudomaskou v příkazu network. Maska sítě zde lze také použít.
Router(config-router)#metric weights tos k1 k2 k3 k4 k5	Změní implicitní hodnoty <i>k</i> , použité při výpočtu metriky. Toto jsou implicitní hodnoty: tos=0,

	k1=1, k2=0, k3=1, k4=0, k5=0
--	------------------------------

POZNÁMKA: Klíčové slovo *tos* (*type of service*) je odkaz na původní protokol IGRP, zamýšlející směrování podle typu služby. Protože to ale nebylo nikdy zavedeno do praxe, je pole *tos* v tomto příkaze **vždy** nastaveno na nulu (0).

POZNÁMKA: S implicitním nastavením je metrika EIGRP redukována na nejpomalejší šířku pásmo plus součet všech zpoždění odchozích rozhraní z lokálního směrovače do cílové sítě.

TIP: Aby mohly dva směrovače zformovat vztah sousedství v EIGRP, musí jim vzájemně souhlasit hodnoty *k*.

UPOZORNĚNÍ: Bez toho, aniž byste byli opravdu velmi dobře obeznámeni s tím, co se děje ve vaší síti, **doporučuje se neměnit hodnoty koeficientů *k***.

Automatická a manuální summarizace v EIGRP

Router(config-router)#auto-summary	Zapne automatickou summarizaci v EIGRP. POZNÁMKA: Implicitní chování automatické summarizace je změněno ze zapnuto na vypnuto - od verze IOS 12.2(8)T.
Router(config-router)#no auto-summary	Vypne automatickou summarizaci.
	POZNÁMKA: Chování automatické summarizace je implicitně vypnuto, počínaje od IOS 12.2. (8)T. To znamená, že IOS nyní posílá směrovací informace o podsítích i mimo hranice sítě v plné třídě (nadsítě).
Router(config)#interface fastethernet 0/0	Vstup do konfiguračního režimu rozhraní.
Router(config-if)#ip summary-address eigrp 100 10.10.0.0 255.255.0.0 75	Zapne manuální summarizaci pro autonomní systém 100 v EIGRP na tomto konkrétním rozhraní pro zadanou síť a masku. Administrativní vzdálenost pro tento summarizovaný směr je nastavena na 75.
	POZNÁMKA: Argument administrativní vzdálenost je v tomto případě nepovinný. Bez něho je na summarizovaný směr automaticky použita hodnota 5.

VAROVÁNÍ: EIGRP automaticky summarizuje síť na hranicích plné třídy. Špatně navržená síť s ne-souvislými podsítěmi může mít problémy s konektivitou, jestliže je funkce summarizace ponechána zapnutá. Například: jestliže by dva směrovače inzerovaly stejnou síť 172.16.0.0/16, když by ve skutečnosti bylo třeba, aby inzerovaly dvě různé sítě 172.16.10.0/24 a 172.16.20.0/24. Doporučená praxe je, abyste vypnuli automatickou summarizaci a použili příkaz *ip summary-address* a summarizovali manuálně to, co je potřeba.

Vyvažovaní zátěže: variance (variace)

Router(config)#router eigrp 100	Vytvoří směrovací proces 100.
Router(config-router)#network 10.0.0.0	Určuje, která síť je inzerována v EIGRP.
Router(config-router)#variance n	Dá pokyn směrovači, aby zahrnul směry s metrikou menší nebo rovnou n-krát minimální metrice směru pro daný cíl. N je číslo specifikované pomocí příkazu variance.

POZNÁMKA: Jestliže cesta není přípustný následník (*feasible successor*), není použita ve vyvažování zátěže.

POZNÁMKA: EIGRP podporuje vyvažování zátěže až šesti cest s nestejnou cenou (metrikou).

Použití příkazu Bandwidth

Router(config)#interface serial 0/0	Vstup do konfiguračního režimu rozhraní.
Router(config-if)#bandwidth 256	Nastaví šířku pásma, přenosovou kapacitu (<i>bandwidth</i>) na 256 kilobitů, aby tak umožnilo lepší kalkulaci metriky v EIGRP.
Router(config-if)#ip bandwidth-percent eigrp 50 100	Nastaví procento přenosové kapacity - šířky pásma (<i>bandwidth</i>), které může být EIGRP použito na tomto rozhraní pro výměnu směrovacích informací. 50 je číslo autonomního systému EIGRP. 100 je hodnota procenta. $100\% * 256 = 256 \text{ kb/s}$.

POZNÁMKA: Implicitně je EIGRP nastaveno pouze na 50 procent šířky pásma rozhraní pro výměnu směrovacích informací. Mohou být nastaveny větší hodnoty než je 100 procent. Takové nastavení může být užitečné jestliže je *bandwidth* z jiných důvodů nastaven uměle nízký (jako je manipulace se směrovací metrikou).

POZNÁMKA: Příkaz *ip bandwidth-percent* se spoléhá na hodnotu nastavenou příkazem *bandwidth*.

Autentizace

Router(config)#interface serial 0/0	Vstup do konfiguračního režimu rozhraní.
Router(config-if)#ip authentication mode eigrp 100 md5	Zapne na tomto rozhraní v autentizaci EIGRP paketů hašovací algoritmus MD5 (<i>Message Digest 5</i>).
Router(config-if)#ip authentication key-chaine-	Zapne na tomto rozhraní autentizaci EIGRP pa-

igrp 100 romeo	ketů. Romeo je jméno pojmenované skupiny klíčů (<i>key chain</i>).
Router(config-if)#exit	Návrat do globálního konfiguračního režimu.
Router(config)#key chain romeo	Určuje pojmenovanou skupinu klíčů (<i>key chain</i>). Jméno musí souhlasit se jménem nastaveným ve výše uvedené konfiguraci rozhraní.
Router(config-keychain)#key 1	Určuje číslo klíče.
	POZNÁMKA: Rozsah klíčů je od 0 do 2147483647. Identifikační čísla klíčů nemusí být na sebe navazující. V řetězci musí být definován nejméně jeden klíč.
Router(config-keychain-key)#key-string shakespeare	Určuje heslo klíče (<i>key string</i>).
	POZNÁMKA: Řetězec klíče (heslo) může obsahovat od 1 do 80 alfanumerických znaků (malá i velká písmena), s výjimkou prvního znaku, který nemůže být číslicí.
Router(config-keychain-key)#accept-lifetime start-time {infinite end-time duration seconds}	Volitelně (nepovinně) určuje periodu, během které mohou být klíče přijímány.
	POZNÁMKA: Implicitní počátek periody a nejranější akceptovatelný datum je 1.1.1993. Implicitní konec periody je nekonečno.
Router(config-keychain-key)#send-lifetime start-time {infinite end-time duration seconds}	Volitelně (nepovinně) určuje periodu, během které mohou být klíče vysílány.
	POZNÁMKA: Implicitní počátek periody a nejranější akceptovatelný datum je 1.1.1993. Implicitní konec periody je nekonečno.

POZNÁMKA: Pro zajištění relevantních údajů pro počátek a konec periody se ujistěte, že má směrovač nastavený správný čas. Doporučovaná praxe je spustit protokol NTP (*Network Time Protocol*) nebo použít jinou metodu pro synchronizaci času, pokud zamýšlite použít nastavení životnosti klíčů.

Verifikace, ověření funkce EIGRP

Router#show ip eigrp neighbors	Zobrazí tabulku sousedů.
--------------------------------	--------------------------

Router#show ip eigrp neighbors detail	Zobrazí tabulkou sousedů detailně.
	TIP: Příkaz <i>show ip eigrp neighbors detail</i> ověřuje zde je soused nastaven jako hraniční směrovač (<i>stub router</i>).
Router#show ip eigrp interfaces	Zobrazí informace pro každé rozhraní.
Router#show ip eigrp interfaces serial 0/0	Zobrazí informace pro konkrétní rozhraní.
Router#show ip eigrp interfaces 100	Zobrazí informace pro rozhraní, na kterém běží proces 100.
Router#show ip eigrp topology	Zobrazí tabulkou topologie.
	TIP: Příkaz <i>show ip eigrp topology</i> zobrazuje, kde jsou Vaši přípustní následníci (<i>feasible successors</i>).
Router#show ip eigrp traffic	Zobrazí počet a typ vyslaných a přijatých paketů.
Router#show ip route eigrp	Zobrazí směrovací tabulkou pouze s řádky od EIGRP.

Odstraňování závad EIGRP

Router#debug eigrp fsm	Zobrazí události/akce související s EIGRP metrikou přípustných následníků (<i>feasible successor metrics (FSM)</i>).
Router#debug eigrp packet	Zobrazí události/akce související s EIGRP pakety.
Router#debug eigrp neighbor	Zobrazí události/akce související s Vašimi EIGRP sousedy.
Router#debug ip eigrp neighbor	Zobrazí události/akce související s Vašimi EIGRP sousedy (pro protokol IP).
Router#debug ip eigrp notifications	Zobrazí oznámení událostí EIGRP.

Příkazy pro kapitolu 9, EIGRP

Příkaz (Command)	Popis (Description)
Router(config)# router eigrp 100	Zapíná EIGRP. 100 je číslo autonomního systému (<i>autonomous system AS</i>), které může být

	mezi 1 a 65535. všechny směrovače ve stejném AS musí mít stejné číslo AS.
Router(config-router)# network 192.168.1.32 0.0.0.31	Umožňuje směrování pro podsíť 192.168.1.32/27. (V případě, že jde o podsíť (= není použita implicitní (= třídní) maska pro danou třídu sítě), je nutné v klauzuli network použít pseudomasku, zástupnou masku (<i>wild-card mask</i>). <u>Pokud by byla uvedena agregovaná třídní adresa, nemusí se žádná pseudomaska používat.</u>)
Router# show ip eigrp neighbors	Zobrazí tabulku sousedů.
Router# show interface serial 0/0/0	Lze ověřit aktuální metriku použitou EIGRP pro rozhraní Serial 0/0/0.
Router(config-if)# bandwidth 128	Mění přenosovou rychlosť (<i>bandwidth</i>) rozhraní na 128 kb/s.
Router# show ip eigrp topology	Zobrazí tabulku topologie. Tento příkaz Vám ukáže kdo jsou Vaši přípustní následníci (= zástupci) (<i>feasible successors</i>), splňují podmínu přípustnosti.
Router# show ip eigrp topology all-links	Zobrazí tabulku topologie včetně cest, které nesplňují podmínu přípustnosti (<i>feasibility condition</i>). Zobrazuje všechny možné cesty do cílové sítě.
Router# debug eigrp fsm	Zobrazí události/akce vztahující se k algoritmu DUAL FSM.
Router(config)# ip classless	Umožní beztřídní směrování. (V IOS od Release 11.3 výše je implicitně zapnuto.)
Router(config-router)# no auto-summary	Vypne automatickou summarizaci sítí na hranicích plné třídy.
Router(config-router)# eigrp log-neighbor-changes	Loguje všechny změny ve vztazích přilehlého sousedství EIGRP (<i>neighbor adjacency</i>).
Router(config-if)# ip summary-address eigrp 100 10.10.0.0 255.255.0.0	Umožní manuální summarizaci na tomto určitém rozhraní pro zadaný adresní prostor 10.10.0.0/16.
Router(config-route)# redistribute static metric	Nastaví EIGRP tak, aby zahrnoval ve svých aktualizacích statické cesty. Je třeba nastavit

	hodnoty EIGRP metrik.
--	-----------------------

Komplexní praktické laboratorní cvičení – EIGRP

Použijte příklad pro RIPv2.

1. Vypněte RIP (*no router rip*).
2. Směrujte pomocí EIGRP v **autonomním systému** 100. Správně určete pseudomasky pro podsítě. Privátní sítě ve třídě C lze vložit do konfigurace EIGRP bez pseudomasky (= je použita implicitní maska třídy C.)
3. Zakažte propagaci EIGRP **do sítí obsahujících pouze koncová zařízení (zde netransitních (stub) sítí) (passive-interface)**.
4. Na hraničním směrovači plné třídy vypněte automatickou summarizaci.
5. Změňte přenosovou rychlosť (*bandwidth*) na jednotlivých rozhraních. Nastavte na obou koncích jednoho média stejnou hodnotu. => Chování se změní. Od původního, kdy se chovalo stejně jako RIP, tzn. Nejlepší je nejkratší cesta (s nejmenším počtem skoků), nyní je délka ovlivněna i přenosovou rychlostí. (Nastavíme na lince mezi R1 a R3 hodnotu 1 000.). Potom do sítě 192.168.2.0 se dostaneme pouze spodní cestou (původně tam byly dvě cesty se stejnou cenou / metrikou).
6. Dále nastavte na směrovači R4 statickou (implicitní) cestu pro síť 10.2.2.0/24 na virtuální rozhraní typu loopback a redistribujte ji na ostatní směrovače. Vypněte automatickou summarizaci.
7. Zobrazte si na R3:
 - směrovací tabulkou (*sh ip route*),
 - tabulkou sousedů (*sh ip eigrp neighbors*),
 - tabulkou topologie (bez a včetně cest, které nesplňují podmínu přípustnosti): *sh ip eigrp topology*, *sh ip topology all-links*,
 - ladící výpis algoritmu DUAL FCM: *debug eigrp fsm*.

Konfigurace EIGRP na R2:

```
!
router eigrp 100
  passive-interface FastEthernet0/0
  network 172.16.2.0 0.0.0.127
  network 172.16.1.0 0.0.0.255
  network 192.168.2.0
  auto-summary
!
ip classless
!
```

Redistribuce statické cesty na R4:

```
<vynecháno>
interface Loopback0
  ip address 10.1.1.1 255.255.255.0
!
<vynecháno>
!
router eigrp 100
  redistribute static metric 100 10 255 255 1500
    passive-interface FastEthernet0/1
    network 192.168.4.0
    network 192.168.3.0
    no auto-summary
!
ip classless
ip route 10.2.2.0 255.255.255.0 Loopback0
!
```

<vynecháno>!

Směrovač R3:

Směrovací tabulka R3:

```
10.0.0.0/24 is subnetted, 1 subnets
D EX 10.2.2.0 [170/25605120] via 192.168.3.253, 00:03:32, FastEthernet0/0
      172.16.0.0/16 is variably subnetted, 5 subnets, 3 masks
D     172.16.0.0/16 is a summary, 00:12:04, Null0
D     172.16.1.0/24 [90/33280] via 172.16.2.253, 00:19:46, FastEthernet1/1
D     172.16.2.0/25 [90/30720] via 172.16.2.253, 00:19:46, FastEthernet1/1
C     172.16.2.128/25 is directly connected, FastEthernet1/1
C     172.16.3.0/24 is directly connected, FastEthernet1/0
D     192.168.1.0/24 [90/35840] via 172.16.2.253, 00:12:05, FastEthernet1/1
D     192.168.2.0/24 [90/33280] via 172.16.2.253, 00:19:46, FastEthernet1/1
C     192.168.3.0/24 is directly connected, FastEthernet0/0
D     192.168.4.0/24 [90/30720] via 192.168.3.253, 00:03:30, FastEthernet0/0
D     192.168.5.0/24 [90/284160] via 172.16.2.253, 00:19:46, FastEthernet1/1
```

Tabulka sousedů R3:

```
R3#sh ip eigrp nei
```

Cisco NetAcad: CCNA Exploration - Routing Protocols and Concepts – studijní materiál

IP-EIGRP neighbors for process 100

H	Address	Interface	Hold (sec)	Uptime 00:21:49	SRTT (ms)	RTO 1000	Q 0	Seq 74	Cnt	
									Num	
0	172.16.2.253	Fa1/1	11	00:21:49	40	1000	0	74		
1	172.16.3.253	Fa1/0	14	00:14:07	40	1000	0	96		
2	192.168.3.253	Fa0/0	14	00:05:38	40	1000	0	35		

Tabulka topologie (bez cest, které nesplňují podmínučku přípustnosti) R3:

R3#sh ip eigrp topology

IP-EIGRP Topology Table for AS 100

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
r - Reply status

```
P 172.16.3.0/24, 1 successors, FD is 2562560
    via Connected, FastEthernet1/0
P 192.168.3.0/24, 1 successors, FD is 28160
    via Connected, FastEthernet0/0
P 172.16.0.0/16, 1 successors, FD is 28160
    via Summary (28160/0), Null0
P 172.16.2.128/25, 1 successors, FD is 28160
    via Connected, FastEthernet1/1
P 172.16.2.0/25, 1 successors, FD is 30720
    via 172.16.2.253 (30720/28160), FastEthernet1/1
    via 172.16.3.253 (2567680/30720), FastEthernet1/0
```

< vynescháno >

R3#

Poznámka:

1. **P**- tento směr je v **pasivním stavu** (*passive state*). Když algoritmus DUAL neprovádí svůj výpočet k určení cesty do sítě, směr, cesta je ve stabilním režimu (*stable mode*), který je známý jako pasivní stav (*passive state*). Jestliže DUAL přepočítává nebo hledá novou cestu, směr, cesta je v aktivním stavu (*active state*). Všechna směrovače v topologické tabulce by měly být ve stabilním stavu pro stabilní směrovací doménu. DUAL zobrazí stav A, jestliže je směrovač „*Stuck in Active*“ (= uvázlý, přilepený v aktivním stavu), což je problém pro výuku hledání a odstraňování chyb na úrovni kurzu CCNP).
2. **FD is 30720 – přípustná vzdálenost následníka** (**Feasible Distance (FD) of Successor**) = metrika příslušné cesty ve směrovací tabulce
3. .../**30720** – inzerovaná vzdálenost záložní cesty (*reported distance of feasible successor*).

Tabulka topologie (včetně cest, které nesplňují podmínu přípustnosti, tzn. všechny cesty) R3:

Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Co je účelem EIGRP PDM (= *Protocol Dependent Module*, modul závislý na protokolu)?
 - a) PDM poskytuje modulární podporu pro L3 protokoly.
- 2) Spárujte termíny EIGRP a jejich popisy:
 - a) obsahuje směry EIGRP určené pro přenosílání paketů = směrovací tabulka,
 - b) primární směr, který má být použit, vybraný algoritmem DUAL = následník (*successor route*)
 - c) nejdůležitější datový zdroj EIGRP, obsahuje seznam směrovačů s vytvořeným sousedstvím (*adjacency*) = tabulka sousedů (*neighbor table*)
 - d) záložní cesta do cílové sítě = přípustný následník (*feasible successor route*)
 - e) obsahuje všechny naučené (zjištěné) směry do všech cílových sítí = topologická tabulka (*topology table*)
- 3) Který typ paketů EIGRP je použit pro objevování, verifikaci a znova objevování sousedních směrovačů?
 - a) Kontaktní paket hello
- 4) Jestliže směr EIGRP spadne a v topologické tabulce není pověřený následník (= záložní směr), jakým návěstím (flag) DUAL označí tento směr, který selhal?
 - a) Aktivní
- 5) Které tři tabulky EIGRP spravuje (= udržuje)?
 - a) Směrovací
 - b) topologická
 - c) sousedů
- 6) Jaký je účel tabulky sousedů a topologické tabulky u EIGRP?
 - a) Jsou použity algoritmem DUAL pro vytvoření (naplnění) směrovací tabulky.
- 7) Co znamená číslo 255/255 ve následujícím výpisu?

```
R1#sh int fa1/0
FastEthernet1/0 is up, line protocol is up (connected)
  Hardware is Lance, address is 0030.a309.4001 (bia 0030.a309.4001)
  Internet address is 172.16.1.253/24
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, rely 255/255, load 1/255
  Encapsulation ARPA, loopback not set
```

- a) Pravděpodobnost, že linka bude dále funkční (= spolehlivost).

- 8) Spárujte termíny DUAL s jejich popisy:
- funkční záložní cesta do cíle = přípustný následník (*feasible successor*)
 - směr, který je použit pro přeposílání paketů do cíle a zároveň směr s nejmenšími náklady = následník (*successor*)
 - nejnižší vypočtená metrika pro dosažení cílové sítě = přípustná vzdálenost (*feasible distance*)
 - tabulka, která obsahuje následníky i přípustné následníky = topologická tabulka
 - tabulka, která obsahuje pouze následníky = směrovací tabulka
- 9) Administrátor hledá a odstraňuje závady směrování EIGRP. Který příkaz vypíše všechny možné cesty do cíle?
- show ip eigrp topology all-link
- 10) Jaká je inzerovaná (oznamovaná) vzdálenost (*reported distance*) v inzerovaném přípustném následníkovi do sítě 172.16.2.128/25?

```
R1#show ip eigrp topology
IP-EIGRP Topology Table for AS 100

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - Reply status

P 192.168.1.0/24, 1 successors, FD is 28160
    via Connected, FastEthernet0/0
P 172.16.2.128/25, 1 successors, FD is 33280
    via 172.16.1.254 (33280/30720), FastEthernet1/0
    via 172.16.3.254 (4294967295/28160), FastEthernet1/1
```

- a) 28160

Kapitola 10 - Směrovací protokoly typu stav linky (Link-State)

V této kapitole se naučíme:

- Popsat základní koncepty a funkce směrovacích protokolů používajících algoritmus stavu linky
- Popsat výhody a požadavky kladené na protokoly typu stav linky

Směrování typu stav linky

Směrovací protokoly typu vektor vzdálenosti (*distance vector*) si můžeme představit jako směrové dopravní značky na silnici (*road signs*), protože směrovače musí rozhodnout o preferovaném směru na základě vzdálenosti neboli metriky do cílových sítí. Právě tak jako cestovatel důvěřuje dopravnímu značení, že ukazuje správnou vzdálenost do dalšího města, směrovače s vektorem vzdálenosti důvěřují, že ostatní směrovače inzerují pravdivou vzdálenost do cílové sítě.

Směrovací protokoly typu stav linky (*link-state*) volí jiný přístup. Směrovací protokoly typu stav linky (*link-state*) jsou, pro představu, spíše jako silniční mapy, protože vytvářejí mapu topologie sítě a každý směrovač tuto mapu používá k určení nejkratší cesty do každé sítě. Tak, jako se vy podíváte do mapy, abyste našli směr do jiného města, směrovače se stavem linky používají topologickou mapu k určení preferované cesty k dosažení dalšího cíle.

Směrovací protokoly typu **stav linky** (*Link-State Routing Protocols*) jsou také známy jako **protokoly typu nejkratší cesta jako první** (*shortest path first protocols, SPF*) a jsou postaveny na **algoritmu SPF Dijkstru**²⁵.

Pro IP jsou nejznámějšími protokoly stavu linky:

- Open Shortest Path First (OSPF)
- Intermediate System-to-Intermediate System (IS-IS)

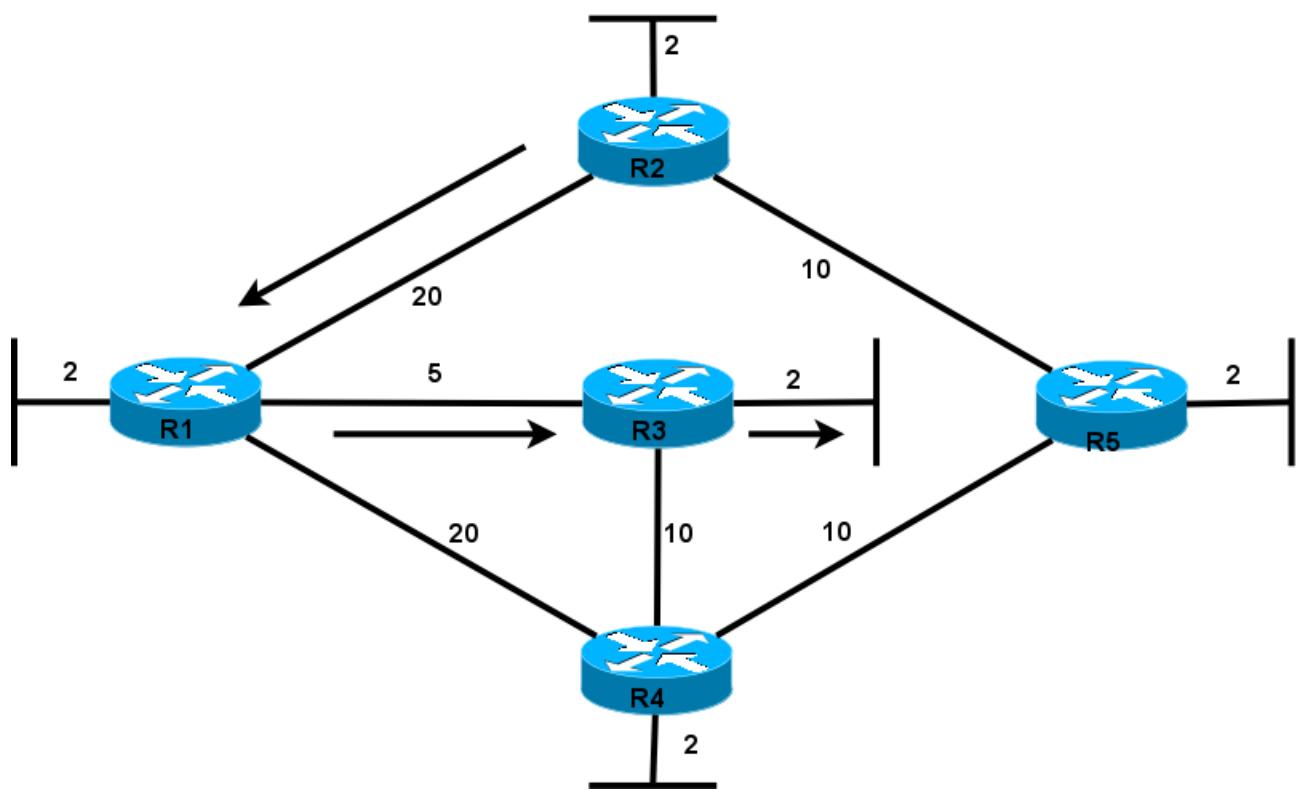
Poznámka: OSPF bude diskutován v kapitole 11 a IS-IS v kurzu CCNP. Existují také směrovací protokoly typu stav linky pro sítě nepoužívající protokol IP. Například DNA Phase V od firmy DEC, Netware Link Service Protocol (NLSP) od firmy Novell, ty se nebudou probírat ani v CCNA ani v CCNP.

25 Edsger Wybe Dijkstra (1930 – 2002) byl holandský vědec v oboru počítačů.

Úvod do algoritmu SPF

Algoritmus „nejkratší cesta první“ (Shortest Path First, SPF) akumuluje ceny (costs) podél každé cesty od zdroje do cíle. Každý směrovač vypočítává algoritmus SPF a určuje metriku = **cenu** (cost) ze své vlastní perspektivy (sčítá jednotlivé ceny jednotlivých segmentů sítě (linek) podél každé možné cesty do cíle včetně ceny segmentu cílové sítě (ze směrovače do cílového hostitele) a s výjimkou ceny segmentu zdrojové sítě (od zdrojového hostitele do bránového směrovače zdrojové sítě)). Přestože algoritmus Dijkstra je znám jako algoritmus nejkratší cesta první, je to ve skutečnosti smysl každého směrovacího protokolu.²⁶

Dijkstrův algoritmus nejkratší cesta jako první (Dijkstra's Shortest Path First Algorithm)



Postup zpracování algoritmu SPF na směrovači:

1. Každý směrovač se dozví o každé k sobě přímo připojené síti,
2. Každý směrovač je zodpovědný, že řekne „hello“ (= pošle kontaktní pakety hello) každému sousedovi v přímo připojené síti
 - podobně, jako v EIGRP, se tak vytvoří vztah přilehlosti, sousedství (*adjacency*) (v dané oblasti),

²⁶ Ve směrovací tabulce je vždy pouze „nejlepší“ tj. nejkratší, nejrychlejší cesta (směr) do cíle.

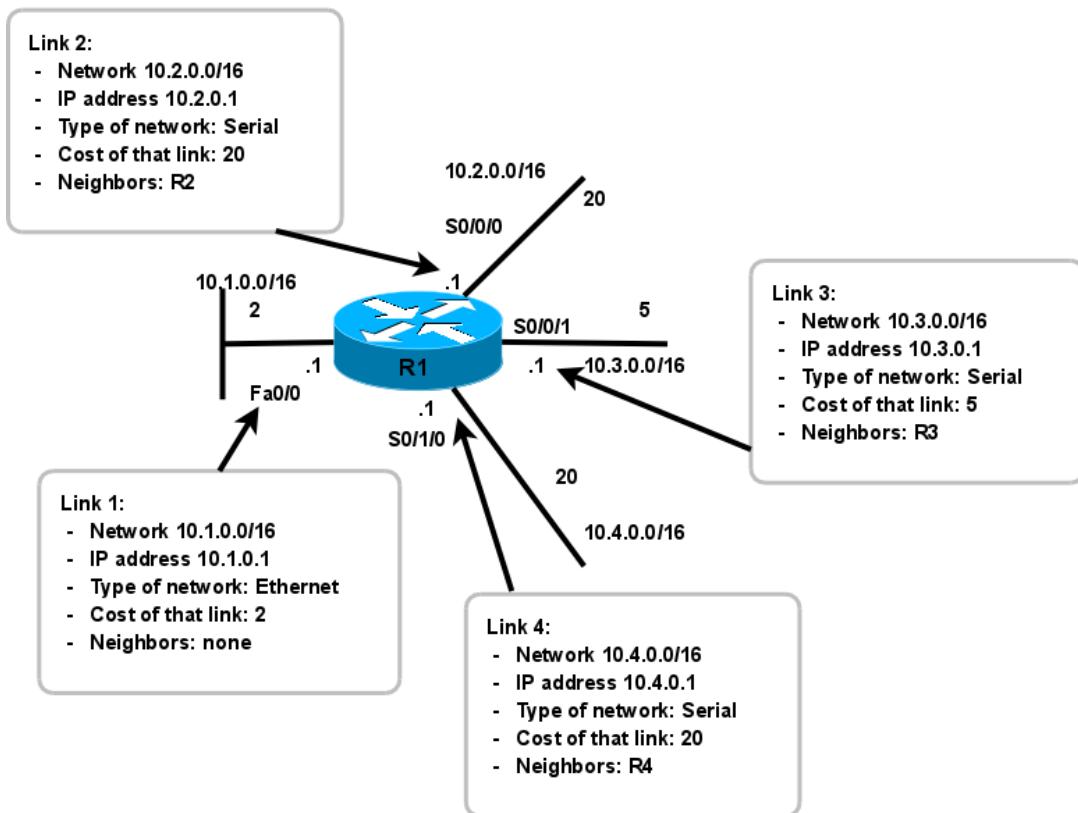
3. Každý směrovač sestavuje **pakety stavu linky** (*Link-State Packet, LSP*), které obsahují stavy každé přilehlé (přímo připojené) linky
 - o LSP obsahuje:
 - údaje o lince mezi dvěma směrovači: směrovač 1 – směrovač 2, ID souseda, typ linky, adresa sítě, maska, přenosová kapacita, cena,
 - nebo informace o netranzitní síti.
4. Kdykoliv při změně topologie, zapnutí/vypnutí linky, nebo zapnutí směrovače nebo směrovacího protokolu (vytvoření vztahu sousedství), každý směrovač zaplavuje (*flood*) pakety stavu linky (*Link-State Packet, LSP*) všechny sousedy v přímo připojených, přilehlých, síťích ve směrovací oblasti, kteří potom ukládají všechny přijaté pakety stavu linky (LSP) do své **databáze stavu linky** (*link-state database, LSDB*).
 - o Nezapomeňte: LSP není posílán periodicky!
 - o Každý směrovač **ve směrovací oblasti** (*area*) bude mít LSP ze všech směrovačů v této oblasti,
5. Jednotlivé směrovače si vytvářejí úplnou a synchronizovanou **mapu topologie sítě** a nezávisle počítají **nejlepší cestu do každé cílové sítě** (s celkovou nejnižší cenou celé trasy).
 - o Vytváří si **strom sítě** (*Link State Tree*) – mapu neobsahující smyčky.

Informace o stavu linky

Informace o stavu linek směrovače je známa jako stavy linky (*Link States*). Obsahuje:

- IP adresu sítě a masku podsítě přilehlé sítě,
- IP adresu rozhraní směrovače,
- typ sítě (Ethernet (*broadcast*) nebo sériové dvoubodové připojení (*point-to-point link*)),
- cenu této linky,
- všechny sousedící (přilehlé) směrovače této linky.

Informace o stavu linky pro směrovač R1 (Link State Information for R1)



Výhody algoritmu Link-State

Následuje několik výhod směrovacích protokolů typu stav linky proti protokolům typu vzdálenosti:

- Každý směrovač si vytváří vlastní topologickou mapu neboli strom SPF síťové topologie, ze kterého si sám počítá nejkratší cestu.
- Bezprostředním zaplavováním (*flooding*) sousedů pakety LSP se dosáhne rychlá konvergence.
- LSP jsou posílány pouze při změně topologie a obsahují pouze informace týkající se této změny – automaticky spouštěné aktualizace (*triggered update*).
- Hierarchický návrh, při použití více oblastí (*area*).

Systémové požadavky

Systémové požadavky na směrovač s protokolem typu stav linky jsou proti protokolům typu vektory vzdálenosti zvýšené o:

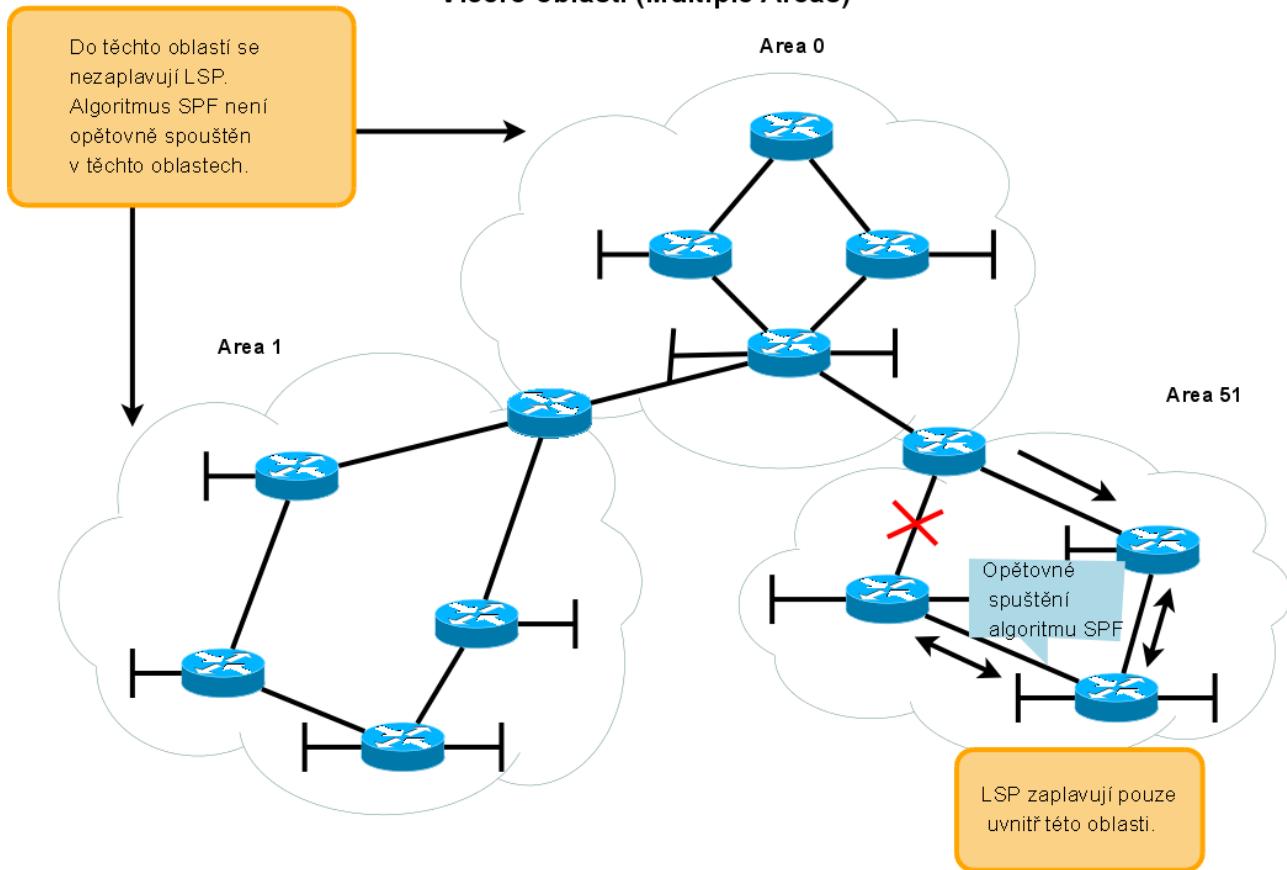
- Operační paměť pro databázi link-state.

- Procesorový čas pro výpočet algoritmu SPF.
- Přenosová kapacita (šířka pásma) pro záplavy paketů LSP (ta je ale čerpána převážně při startu směrovače, později obvykle nastávají již pouze malé změny topologie).

Vícero oblastí

Aby se zmenšila zátěž procesoru směrovače a požadavky na jeho paměť, je topologie pro směrování typu stav linky rozdělena do malých **oblastí** (area). Procesor je nejvíce zatížen při počáteční záplavě (flood) paketů stavu linky (Link State Packet, LSP), poté už přicházejí pouze změny topologie (tím je pro aktualizace potřeba nižší šířka pásma). Konvergenci sítě urychlují aktualizace spouštěné změnami v síti (triggered updates). Při více oblastech musí být jedna z těchto oblastí **páteřní oblast** (area 0).

Vícero oblastí (Multiple Areas)



Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Které tři mechanismy používají směrovací algoritmy typu stav linky (*link state*) k vytvoření a ke správě směrovacích tabulek?
 - a) Kontaktní pakety hello,
 - b) Oznamovače stavu linky LSA (*Link-State Acknowledgment*),

- c) Algoritmus SPF.
- 2) Porovnání vlastností směrovacích algoritmů:
 - a) stav linky (*link-state*):
 - i. používají algoritmus Dijkstra (SPF),
 - ii. vytvářejí kompletní topologii na každém směrovači,
 - iii. rychlá konvergence (= hlavní výhoda proti vektoru vzdálenosti),
 - iv. větší zatížení a požadavky na HW (= hlavní nevýhoda proti vektoru vzdálenosti).
 - b) Vektor vzdálenosti (*distance vector*):
 - i. používají algoritmus Bellman-Ford,
 - ii. závislé na cestách zjištěných od souseda,
 - iii. cesty jsou tedy známé „z doslechu“ (*by „rumor“*),
 - iv. používají periodicky se opakují aktualizace.
- 3) Co je obsaženo v LSP posílaných směrovači typu stav linky (*link-state*) na jejich sousedy?
 - a) Stav přímo připojených linek
- 4) Potom, co si dva směrovače OSPF vymění kontaktní pakety *hello* a vytvoří vztah sousedství (*adjacency*), je další krok?
 - a) Začnou si navzájem posílat pakety LSP.
- 5) Jak se směrovač dozví o přímo připojené síti?
 - a) Když administrátor přiřadí k rozhraní IP adresu a masku podsítě.

Kapitola 11 - Protokol OSPF

V této kapitole se naučíme:

- Popsat východiska a základní funkce OSPF
- Popsat a použít základní konfigurační příkazy OSPF
- Popsat, vypočítat a modifikovat metriku používanou OSPF
- Popsat proces volby pověřeného směrovače/záložního pověřeného směrovače (*Designated Router / Backup Designated Router - DR/BDR*) v síti s více přístupy (s více branami)
- Využít příkazu „**default-information originate**“ ke konfiguraci a k propagaci implicitní cesty v OSPF

Úvod do OSPF

Open Shortest Path First (OSPF) je **veřejný směrovací protokol typu stav linky**, který byl vyvinut jako náhrada směrovacího protokolu typu vektor vzdálenosti RIP. RIP byl přijatelný v počátcích sítí a Internetu, ale spoléhání se na počet přeskoků jako na jediný způsob určení nejlepší trasy rychle přestalo být ve velkých sítích akceptovatelné. OSPF je **beztrídní směrovací protokol**, který používá pro svoji rozšiřitelnost **konceprt oblastí (area)**. Metrika je definována jako libovolná hodnota nazývaná **cena (cost)** podle RFC 2328.

Hlavní výhodou OSPF proti RIP je jeho **rychlá konvergence** a jeho **rozšiřitelnost na mnohem větší síť**. V této závěrečné kapitole tohoto kurzu se naučíte implementaci a konfiguraci OSPF v jedné oblasti. Komplexnější konfigurace jsou v kurzu CCNP.

Historické pozadí

Počáteční vývoj OSPF začala pracovní skupina OSPF při Internet Engineering Task Force (IETF) v roce 1987. V té době byl Internet převážně v akademických a výzkumných sítích financovaných vládou USA.

V roce 1989 byla publikována specifikace OSPFv1 v RFC 1131. OSPFv1 byl experimentální směrovací protokol, který nebyl nikdy nasazen.

V roce 1991 uveden OSPFv2 v RFC 1247 (napsal John Moy). Ve stejné době pracovala ISO na svém vlastním směrovacím protokolu typu stav linky *Intermediate System-to-Intermediate System (IS-IS)*. IETF doporučila OSPF jako vnitřní směrovací protokol IGP (Interior Gateway Protocol).

V roce 1998 byla specifikace OSPFv2 aktualizována nyní platnou RFC 2328.

Poznámka: V roce 1999 byl publikován OSPFv3 pro IPv6 v RFC 2740 (napsali John Moy, Rob Co-ltun a Dennis Ferguson). OSPFv3 je probírána v CCNP.

Linky:

OSPFv2 <http://www.ietf.org/rfc/rfc2328.txt>

Zjednodušená činnost OSPF

1. Směrovač vysílá přes svá rozhraní kontaktní pakety (*Hello packet*). Pokud se dva navzájem propojené routery pomocí těchto paketů dohodnou na určitých společných parametrech, stávají se sousedy (*neighbors*)
2. Mezi některými ze sousedů se vytvářejí užší vazby sousedství. Tyto směrovače se pak označují jako přilehlé (*adjacent*).
3. Přilehlé směrovače si vzájemně vyměňují aktualizační pakety (*Link-State Update, LSU*) obsahující oznamovače LSA (*Link-State Advertisement*). Informace v oznamovačích popisují stav rozhraní směrovače nebo seznam směrovačů připojených k dané síti.
4. Všechny směrovače si ukládají přijaté LSA do své lokální topologické databáze (LSDB) a zároveň je přeposílají na ostatní přilehlé směrovače. Tím se informace postupně záplavově (*flood*) rozšíří mezi všechny směrovače v síti. Výsledkem bude shodná topologická databáze na všech směrovačích.
5. Po naplnění databáze (*Link-State DataBase, LSDB*) každý směrovač samostatně provede výpočet pomocí SPF (Dijkstrova) algoritmu. Jeho výsledkem bude nalezení nejkratší cesty do každé známé sítě v podobě stromu a tím odstranění smyček v topologii sítě.
6. Na základě vypočtených dat ve stromu SPF (*SPF tree*) je možné naplnit směrovací tabulkou směrovač nejlepšími cestami do cílových sítí.
7. Pokud dojde ke změně topologie sítě, směrovač, na kterém ke změně došlo, odešle přilehlým směrovačům informaci v podobě datových položek LSA v LSU paketu. Ty se postupně rozšíří po celé síti a každý směrovač upraví svou topologickou databázi a provede nový výpočet SPF algoritmu.

Zapouzdření zprávy protokolu OSPF

Záhlaví linkové vrstvy	Záhlaví paketu IP	Záhlaví paketu OSPF	Data specifická dle typu paketu OSPF
Rámcem linkové vrstvy			
Zdrojová MAC adresa = adresa vysílajícího rozhraní			
Cílová MAC adresa = Multicast: 01-00-5E-00-00-05 nebo 01-00-5E-00-00-06			
	Paket IP Zdrojová IP adresa = adresa vysílajícího rozhraní Cílová IP adresa = Multicast: 224.0.0.5 nebo 224.0.0.6 Protokol = 89 pro OSPF		
		Záhlaví paketu OSPF Kód typu pro Typ paketu OSPF ID směrovače a ID oblasti	
			Typ paketu OSPF 0x01 Hello 0x02 Database Description 0x03 Link State Request 0x04 Link State Update 0x05 Link State Acknowledgment

Typy paketů OSPF

1. **Hello** – kontaktní pakety *hello* objevují sousedy (*neighbor*) OSPF, vytvářejí a udržují vztah přilehlého sousedství (*adjacency*) s ostatními směrovači OSPF.
2. **DBD - The Database Description** – zkrácený výpis link-state databáze vysílajícího směrovače, určen k ověření a synchronizaci lokální databáze link-state na přijímajícím směrovači.
3. **LSR - Link-State Request** – žádost o další informace pro řádku DBD.
4. **LSU - Link-State Update** – odpověď na LSR, který žádal nové informace. LSU může obsahovat až 11 (7) různých typů oznamovačů *Link-State Advertisements* (LSA) (někdy se jako synonymum pro LSA používá termín *Link-State Update* (LSU), ve skutečnosti LSU ob-sahuje jeden nebo více LSA). Jednotlivé LSA obsahují směrovací informace do cílové sítě.
Typy LSA:
 - 4.1. Směrovač – Router,
 - 4.2. Síť – Network,
 - 4.3. Agregace – Summary,
 - 4.4. Agregace – Summary,
 - 4.5. Externí autonomní systém,
 - 4.6. Multicast OSPF
 - 4.7. Definované pro tranzitní oblasti (not-so-stubby areas),
 - 4.8. Externí atributy pro protokol BGP,
 - 4.9. nejasný LSA,
 - 4.10. nejasný LSA,
 - 4.11. nejasný LSA.
5. **LSAck - Link-State Acknowledgement (LSAck)** – potvrzení přijetí LSU.

Záhlaví OSPF paketu (v2):

0	1	2	3
0 1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1
Version # Type Packet length			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Router ID			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Area ID			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Checksum AuType			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Authentication			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Authentication			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+

Kontaktní paket Hello:

0	1	2	3
0 1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1 2 3 4 5 6 7 8 9 0	1
Version # 1 Packet length			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Router ID			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Area ID			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Checksum AuType			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Authentication			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Authentication			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Network Mask			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
HelloInterval Options Rtr Pri			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
RouterDeadInterval			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Designated Router			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Backup Designated Router			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
Neighbor			
+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+	+-----+-----+-----+-----+
...			

Zdroj: <http://www.ietf.org/rfc/rfc2328>

Kontaktní pakety Hello

- Objevují sousedy (*neighbor*) OSPF, vytvářejí a udržují vztah přilehlosti, sousedství (*adjacency*) s ostatními směrovači OSPF.
- Inzerují parametry, na kterých se dva směrovače musí shodnout, aby vytvořily vztah sousedství.
- Volí pověřený směrovač (*Designated Router (DR)*) a záložní pověřený směrovač (*Designated Router (BDR)*) v sítích s více přístupy (s více branami) (*multiaccess networks*) jako jsou

Ethernet nebo Frame Relay.

- Nejčastěji je zasílán na skupinovou adresu *ALLSPFRouters* 224.0.0.5.

Aby bylo možné vytvořit vztah přilehlého sousedství mezi dvěma směrovači, musí mít rozhraní těchto směrovačů stejné hodnoty pro následující proměnné:

- **Hello interval** – indikuje jak často směrovač vysílá hello pakety (v sekundách)
 - 10 sekund - implicitně v segmentech broadcastových sítí s vícenásobnými přístupy (*broadcast multiaccess (BMA)*) (Ethernet) a dvoubodová spojení (point-to-point).
 - 30 sekund v segmentech *non-broadcast multiaccess (NBMA)* sítí (Frame Relay, X.25, ATM).
- **Dead interval** – perioda v sekundách, po kterou bude směrovač čekat na příjem hello paketu, než označí sousedství za „mrtvé“ a zruší ho. Jestliže vyprší dead interval před tím, než směrovač přijme hello paket, OSPF smaže souseda ze své link-state databáze LSDB. Směrovač zaplaví (*floods*) informacemi, že sousedství je vypnuté, všechna rozhraní, na kterých je spuštěný OSPF. Je obvykle nastaven na čtyřnásobek intervalu hello.
 - 40 sekund - segmenty multiaccess a point-to-point,
 - 120 sekund – sítě NBMA.
- **Network type** – typ sítě:
 - Broadcast sítě - ty sítě, které jsou schopny vzájemně propojit více než dva počítače a navíc zajišťují, že jeden vyslaný paket mohou přijmout současně všechny počítače. Typickými představiteli broadcast sítí jsou sítě typu Ethernet nebo FDDI.
 - Point to point sítě (dvoubodové spoje) - sítě spojující pouze dva směrovače. Jejich typickým příkladem jsou sériové linky. Na těchto sítích se nevolí DR/BDR a směrovače na point to point sítích se vždy stávají přilehlými. Pro komunikaci mezi nimi se používá pouze multicast adresa 224.0.0.5.
 - NBMA sítě - *Non Broadcast Multi Access*. Sítě tohoto typu může propojit více než dva směrovače, není však schopna posílat broadcasty. Není tedy možné vyslat paket, který by byl přijat všemi směrovači současně. Jako příklad NBMA sítě můžeme uvést síť Frame Relay, ATM nebo X.25. Na NBMA síti se volí DR a BDR a veškerá komunikace probíhá pomocí unicastů.

Volba DR a BDR

Aby zmenšil objem provozu OSPF **v sítích s více přístupy**, s více branami (*multiaccess network*), OSPF volí pověřený směrovač (*Designated Router (DR)*) a záložní pověřený směrovač (*Backup Designated Router (BDR)*). DR (směrovač s nejvyšší prioritou) je zodpovědný za aktualizace všech ostatních směrovačů OSPF (nazývaných *DROther*), když nastane změna topologie v síti s více přístupy (*multiaccess network*). BDR monitoruje DB a převeze funkci DR, pokud aktuální DR selže.

Jak je volen DR a BDR?

1. DR: směrovač s nejvyšší prioritou OSPF rozhraní
2. BDR: směrovač s druhou nejvyšší prioritou OSPF rozhraní

3. Jestliže jsou OSPF priority shodné, rozetne nerozhodný výsledek nejvyšší ID směrovače.

Konfigurace priority rozhraní:

```
Router(config-if)#ip ospf priority 255
```

(tímto se nastaví rozhraní nejvyšší možná priorita => bude zvoleno DR)

Zobrazení aktuální priority daného rozhraní a ID routeru:

```
Router# show ip ospf interface jméno_rozhraní
```

Implicitní priorita pro rozhraní směrovače je jednička (1). **Pokud mají všechny směrovače nastavenou implicitní prioritu rozhraní, bude jako DR zvolen směrovač s nejvyšším identifikátorem směrovače (Router ID, RID).**

Jednotlivé DROther (= jiné směrovače než DR nebo BDR (*DR other*)) budou formovat sousedství typu **FULL** pouze s DR a BDR, ale budou stále formovat přilehlé sousedství s jakýmkoliv jiným směrovačem DROther, který je připojený v síti. To znamená, že všechny směrovače DROther v síti s více přístupy (*multiaccess*) stále přijímají kontaktní pakety hello ze všech ostatních směrovačů DROther. Tímto způsobem jsou si vědomy všech směrovačů v síti. Když dva směrovač typu DROther zformují přilehlé sousedství, je stav sousedství zobrazen jako typ **2WAY**. Další stavы sousedství jsou diskutovány v kurzu CCNP.

Algoritmus OSPF

Každý OSPF směrovač spravuje svoji databázi stavů linek (*link-state database*), která obsahuje jednotlivé LSA přijaté ze všech ostatních směrovačů. Jakmile směrovač přijal všechny oznamovače v aktualizaci a sestavil svoji lokální databázi, OSPF použije Dijsktrův algoritmus SPF k vytvoření stromu SPF (*SPF tree*). SPF strom je potom použit k naplnění směrovací tabulky nejlepšími směry do každé sítě.

Autentizace

OSPF pakety jsou šifrované a autentizované.

Je dobrou praxí autentizovat přenášené směrovací informace. RIPv2, EIGRP, OSPF, IS-IS a BGP mohou všechny být nakonfigurované, aby šifrovaly a autentizovaly jejich směrovací informace (aktualizace, nikoliv směrovací tabulky). Tato praxe zajišťuje, že směrovače akceptují pouze ty směrovací informace z druhých směrovačů, které byly nastavené se stejným heslem nebo autentizační informací.

Poznámka: Autentizace nešifruje směrovací tabulku.

Identifikátor směrovače

Identifikátor směrovače (*Router ID*) je unikátní identifikace směrovače v OSPF doméně. Router ID je jednoduše IP adresa. Směrovače Cisco odvozují hodnotu Router ID na základě tří kritérií a následující nadřazenosti:

1. směrovač použije IP adresu nastavenou příkazem „**router-id**“,
2. jestliže není nastaven příkaz „**router-id**“, směrovač si zvolí **nejvyšší adresu ze všech svých rozhraní typu loopback**,

3. jestliže nejsou nastavená žádná rozhraní typu loopback, směrovač si vybere **nejvyšší aktivní adresu ze všech svých fyzických rozhraní**.

Verifikace identifikátoru směrovače (*Router ID*): **show ip protocols**. Pokud některá verze IOS nevrací Router ID v tomto příkazu, použijte: **show ip ospf** nebo **show ip ospf interface**.

Rozhraní typu loopback:

```
Router(config)#interface loopback number
Router(config-if)#ip address ip-address subnet-mask
```

Nastavení příkazem router-id:

```
Router(config)#router ospf process-id
Router(config-router)#router-id ip-address
```

Při dodatečných změnách v příkazu *network* nebo *router-id* je vhodné restartovat směrovač (*Router#reload*) nebo vymazat proces OSPF:

```
Router#clear ip ospf process
```

Duplikace identifikátorů směrovače:

IOS duplicitu detekuje a oznámí:

```
%OSPF-4-DUP_RTRID1: Detected router with duplicate router ID
```

Ověření funkčnosti OSPF

Ověření vztahu přilehlosti

```
show ip ospf neighbor
```

Výstupy příkazu show ip ospf neighbor:

- **Neighbor ID** – identifikátor sousedícího směrovače.
- **Pri** - OSPF priorita rozhraní.
- **State** – stav rozhraní. Stav FULL znamená, že směrovač a jeho soused mají identické databáze LSDB. Stavy (state) jsou podrobněji diskutovány v kurzu CCNP.
- **Dead Time** – zbývající čas, který bude směrovač čekat na přijmutí kontaktního paketu hello od souseda před tím, než prohlásí sousedství za zrušené (mrtvé). Tato hodnota je resetována, když rozhraní přijme kontaktní hello paket.
- **Address** - IP adresa sousedova rozhraní, kterému je tento směrovač přímo připojen.
- **Interface** – rozhraní, na kterém tento směrovač zformoval sousedství / přilehlost se sousedem.

Poznámka: na multiaccess sítích²⁷ - sítích s vícenásobným přístupem, jako je Ethernet, mohou mít dva přilehlé směrovače zobrazen jejich stav jako 2WAY. Viz Volba DR a BDR.

```
R3#show ip ospf neighbor
```

27 Sítě s více branami.

Neighbor ID	Pri	State	Dead Time	Address	Interface	
10.1.1.1	0	FULL/	-	00:00:36	192.168.10.5	Serial0/1/0
10.2.2.2	0	FULL/	-	00:00:36	192.168.10.9	Serial0/1/1
R3#						

Dva směrovače nemusejí mít vytvořený vztah přilehlosti, sousedství, jestliže:

- Vzájemně nesouhlasí (*do not match*) **masky podsítě**, to má za příčinu, že směrovače jsou v různých sítích,
- Vzájemně nesouhlasí OSPF **Hello nebo Dead intervaly**,
- Vzájemně nesouhlasí OSPF **Network Type**.
- Chybějící nebo nesprávný OSPF příkaz **network** (například různá oblast (*area*)).

Nastavení rozhraní, časovačů, typ sítě, cenu linky a vznik sousedství v příslušném směru na konkrétním rozhraní ověříte pomocí *show ip ospf interface*:

```
R3#show ip ospf interface serial 0/1/1
Serial0/1/1 is up, line protocol is up
  Internet address is 192.168.10.10/30, Area 0
    Process ID 1, Router ID 10.3.3.3, Network Type POINT-TO-POINT, Cost: 64
    Transmit Delay is 1 sec, State POINT-TO-POINT, Priority 0
    No designated router on this network
    No backup designated router on this network
    Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
      Hello due in 00:00:00
    Index 3/3, flood queue length 0
    Next 0x0(0)/0x0(0)
    Last flood scan length is 1, maximum is 1
    Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1 , Adjacent neighbor count is 1
    Adjacent with neighbor 10.2.2.2
  Suppress hello for 0 neighbor(s)
R3#
```

V případě, že není sousedství vytvořeno je možno nesoulad nastavených hodnot časovačů zjistit též z výpisu ladícího příkazu: *debug ip ospf events*

```
R3#
00:04:06: OSPF: Rcv hello from 10.2.2.2 area 0 from Serial0/1/1 192.168.10.9
00:04:06: OSPF: Mismatched hello parameters from 192.168.10.9
00:04:06: OSPF: Dead R 40 C 40 Hello R 10 C 50 Mask R 255.255.255.252 C
255.255.255.252
R3#
```

Standardní ověřovací příkazy pro OSPF

- **show ip protocols**
- **show ip ospf**
- **show ip ospf interface ...**
- **show ip route**
- **debug ip ospf events**

Administrativní vzdálenost

Viz tabulka *Administrativní vzdálenosti pro jednotlivé směrovací protokoly* v kapitole 3.

Pro OSPF je implicitní administrativní vzdálenost (*distance, AD*) = 110.

Zjistíte ji příkazem *show ip protocols*

```
R3#sh ip protocols

Routing Protocol is "ospf 1"
<vynecháno>
  Distance: (default is 110)

R3#
```

Metrika OSPF

Metrika OSPF se nazývá cena (*cost*). [Citát z RFC 2328](#):

„Cena je přiřazena k odchozí straně každého rozhraní směrovače. Tato cena je nastavitevná systémovým administrátorem. Čím nižší cena, tím více žádoucí je toto rozhraní pro přeposlání datového provozu.“

Pamatujte, že RFC 2328 nespecifikuje, jaké hodnoty by měly být použity k určení ceny. My se budeme zabývat dvěma metodami nastavení cen na směrovačích Cisco.

Cost = $10^8/\text{bandwidth}$ (v bps, b/s)

Příklad:

Jaká je OSPF cena linky FastEthernet? $10^8/100\ 000\ 000\ \text{b/s} = 1$

Jaká je OSPF cena linky T1 ? $10^8/1\ 544\ 000\ \text{b/s} = 64.7$, která je systémem IOS zaokrouhlená na cenu 64.

Jaká je OSPF cena vytáčené linky 56K (*dial up*)? $10^8/56000 = 1785.71$, která je systémem IOS zaokrouhlená na cenu 1785.

1) Konfigurace metriky pomocí šířky pásmo (rychlosti) (*bandwidth*):

```
R2(config)# interface serial0/0/0
R2(config-if)# bandwidth 64
```

2) Konfigurace metriky přímo nastavením ceny (*cost*):

```
R3(config)# interface serial0/0/0
R3(config-if)# ip ospf cost 390
```

OSPF 2 (aktualizace 1998) RFC: <http://www.ietf.org/rfc/rfc2328.txt>

=> U OSPF (a stejně tak i u EIGRP) stav směrování závisí i na postupně provedených změnách nastavení (algoritmus je konečný automat (FSM)). Někdy je tedy nutné, po změnách konfigurace, vymazat tabulky ukládající průběžné stavy. Tzn. resetovat procesy příslušného směrovacího protokolu (Router# clear ip ospf process) nebo restartovat směrováče.

Příkazy pro kapitolu 11, OSPF

Konfigurace OSPF: Mandatorní (povinné) příkazy

Router(config)#router ospf 123	Nastartuje proces OSPF 123. Identifikátor (ID) procesu je jakékoli kladné celé číslo mezi 1 a 65 535. Identifikátor procesu se nevztahuje k oblasti OSPF (<i>OSPF area</i>). Identifikátor procesu nesouvisí s oblastí OSPF. Identifikátor procesu pouze odlišuje jeden proces od jiného na jednom zařízení.
Router(config-router)#network 0.0.0.255 area 0	OSPF inzeruje rozhraní nikoliv síť. Používá pseudomasku (<i>wildcard mask</i>) k určení, která rozhraní inzerovat. Tato příkazová řádky říká: „Všechna rozhraní s adresou 172.16.10.x mají být vložena do OSPF oblasti 0“. Oblast 0 (area 0) je páteřní oblast . Při více oblastech musí jed-

	na z nich být páteřní.
	POZNÁMKA: Číslo identifikátoru procesu na jednom směrovači nemusí souhlasit s číslem identifikátoru na jakémkoliv jiném směrovači. Na rozdíl od EIGRP, rovnost tohoto čísla na všech směrovačích nezajistí, že se vytvoří sousedství.
Router(config-router)#log adjacency-changes detail	Nastaví směrovač tak, aby posílal systémové logovací zprávy (<i>syslog message</i>), když nastane změna stavu mezi OSPF sousedy.
	TIP: Ačkoliv je příkaz <i>log adjacency-changes</i> implicitně zapnutý, je bez použití klíčového slova <i>detail</i> oznamována (report) pouze událost zapnuto/vypnuto.

Použití pseudomasky v oblastech OSPF

Router(config-router)#network 0.0.0.0 area 0	172.16.10.1	Tuto řádku čtete jako: „Každé rozhraní s přesnou adresou 172.16.10.1 má být dán do oblasti 0.“
Router(config-router)#network 0.0.255.255 area 0	172.16.10.0	Tuto řádku čtete jako: „Každé rozhraní s přesnou adresou 172.16.X.X má být dán do oblasti 0.“
Router(config-router)#network 255.255.255.255 area 0	0.0.0.0	Tuto řádku čtete jako: „Každé rozhraní s jakoukoliv adresou má být dán do oblasti 0.“

Konfigurace OSPF: Nepovinné (volitelné) příkazy**Virtuální rozhraní zpětná smyčka (Loopback)**

Router(config)#interface loopback 0	Vytvoří virtuální rozhraní pojmenované <i>loopback 0</i> a potom směrovač přepne do konfiguračního režimu rozhraní.
Router(config-if)#ip address 192.168.100.1 255.255.255.255	Přiřadí k rozhraní IP adresu. (Všimněte si zadané masky. Kolik je v této síti adres? ²⁸)
	POZNÁMKA: rozhraní typu zpětná smyčka má vždy stav „up and up“ = rozhraní administrativně zapnuté a běžící protokol linkové vrstvy.

28 Je tam právě jenom jedna adresa a to adresa sítě.

	Nevypne se bez toho aniž by se ručně zadal příkaz <i>shutdown</i> . To je výborné pro použití rozhraní <i>loopback</i> jako identifikátorů směrovačů v OSPF (OSPF router ID).
--	---

Router ID

Router(config)#router ospf 1	Spustí OSPF proces číslo 1.
Router(config-router)#router-id 10.1.1.1	Nastaví identifikátor směrovače (<i>Router ID</i>) na 10.1.1.1. Jestliže je tento příkaz použit na OSPF proces, který je již aktivní (má sousedy), je nový identifikátor směrovače použit až po příštím znovuzavedení systému (<i>reload</i>) nebo po manuálním restartu procesu OSPF.
Router(config-router)#no router-id 10.1.1.1	Odstraní z konfigurace statický identifikátor směrovače. Jestliže je tento příkaz použit na OSPF proces, který je již aktivní (má sousedy), je staré chování ID směrovače použito až při příštím znovuzavedení systému (<i>reload</i>) nebo při manuálním restartu procesu OSPF.

Volby pověřeného a záložního pověřeného směrovače (DR/BDR)

Router(config)#interface serial 0/0	Změní režim směrovače na režim konfigurace rozhraní.
Router(config-if)#ip ospf priority 50	Změní OSPF prioritu rozhraní na 50.
	POZNÁMKA: Přiřazená priorita může být mezi 0 a 255. Priorita 0 činí tento směrovač nezpůsobilý stát se pověřeným směrovačem (<i>designated router (DR)</i>) nebo záložním pověřeným směrovačem (<i>backup designated router (BDR)</i>). Nejvyšší priorita vyhrává volbu. Priorita 255 zaručuje nerozhodný výsledek volby. Jestliže mají všechny směrovače stejnou prioritu, bez ohledu na číslo priority, je výsledek volby nerozhodný. Nerozhodnost je prolomena nejvyšším ID směrovače.

Modifikace ceny metriky (cost)

Router(config)#interface serial 0/0	Změní režim směrovače na režim konfigurace rozhraní.
-------------------------------------	--

Router(config-if)#bandwidth 128	Pokud změníte <i>bandwidth</i> , OSPF přeypočte cenu (<i>cost</i>) linky.
Nebo	
Router(config-if)#ip ospf cost 1564	Změní cenu (<i>cost</i>) na hodnotu 1564. POZNÁMKA: Cena linky je určena vydělením referenční šířky pásma (<i>reference bandwidth</i>) šířkou pásma tohoto rozhraní. Šířka pásma rozhraní je číslo mezi 1 a 10 000 000. Měrná jednotka (<i>unit of measurement</i>) je kilobit (kb). Cena je číslo mezi 1 a 65 535. Cena nemá měrnou jednotku - je to jenom číslo.

Autentizace: jednoduchá

Router(config)#router ospf 1	Spustí OSPF proces 1.
Router(config-router)#area 0 authentication	Umožní jednoduchou autentizaci, heslo bude posíláno v čistém textu.
Router(config-router)#exit	Návrat do globálního konfiguračního režimu.
Router(config)#interface fastethernet 0/0	Přesun do konfiguračního režimu rozhraní.
Router(config-if)#ip ospf authentication-key fred	Nastaví klíč (<i>key</i>) tj. heslo (<i>password</i>) na hodnotu <i>fred</i> . POZNÁMKA: Heslo může být libovolný řetězec znaků vložených z klávesnice do délky 8 bajtů. Aby byly schopny si vyměňovat OSPF informace, musí mít všechny sousedící směrovače ve stejné síti stejně heslo.

Autentizace: použití šifrování MD5

Router(config)#router ospf 1	Spustí OSPF proces 1.
Router(config-router)#area 0 authentication message-digest	Umožní autentizaci, kdy heslo bude zašifrováno MD5.
Router(config-router)#exit	Návrat do globálního konfiguračního režimu.
Router(config)#interface fastethernet 0/0	Přesun do konfiguračního režimu rozhraní.

Router(config-if)#ip ospf message-digest-key 1 md5 fred	1 je identifikátor klíče (<i>key-id</i>). Tato hodnota musí být stejná jako na sousedícím směrovači. Md5 indikuje použití hašovacího algoritmu MD5. <i>fred</i> je klíč (heslo) a musí být stejné jako na sousedícím směrovači.
	POZNÁMKA: Jestliže není použit příkaz <i>service password-encryption</i> , když je implementována MD5 OSPF autentizace, je tajné heslo MD5 uloženo v konfiguraci v NVRAM jako čistý text (<i>plain text</i>).

Časovače

Router(config-if)#ip ospf hello-interval timer 20	Změní časovač <i>Hello Interval</i> na 20 sekund.
Router(config-if)#ip ospf dead-interval 80	Změní časovač <i>Dead Interval</i> na 80 sekund.
	POZNÁMKA: Časovače Hello a Dead Interval musí být na směrovačích stejné, aby se mohly stát sousedy.

Propagace implicitní cesty

Router(config)#ip route 0.0.0.0 0.0.0.0 s0/0	Vytvoří implicitní cestu.
Router(config)#router ospf 1	Spustí OSPF proces 1.
Router(config-router)#default-information originate	Nastaví, aby byla implicitní cesta propagována na všechny směrovače OSPF.
Router(config-router)#default-information originate always	Volba <i>always</i> (= vždy) propaguje implicitní „čtyr-nulovou“ cestu i když na tomto směrovači žádná není nastavená.
	POZNÁMKA: Příkazy <i>default-information originate</i> nebo <i>default-information originate always</i> jsou konfigurovány obvykle pouze na „vstupním“ nebo „bránovém“ směrovači, tzn. na směrovači, který propojuje Vaši síť s vnějším světem - <i>Autonomous System Boundary Router (ASBR)</i> .

Ověření konfigurace OSPF

Router#show ip protocol	Zobrazí parametry všech směrovacích protokolů
-------------------------	---

	běžících na tomto směrovači.
Router#show ip route	Zobrazí kompletní směrovací tabulku IP.
Router#show ip ospf	Zobrazí základní informace o směrovacích procesech OSPF.
Router#show ip ospf interface	Zobrazí informace o OSPF, vztahující se ke všem rozhraním.
Router#show ip ospf interface fastethernet 0/0	Zobrazí informace o OSPF, vztahující se k rozhraní fastethernet 0/0.
Router#show ip ospf border-routers	Zobrazí informace o hraničních a okrajových směrovačích.
Router#show ip ospf neighbor	Vypíše všechny OSPF sousedy a jejich stavy.
Router#show ip ospf neighbor detail	Zobrazí detailní výpis sousedů.
Router#show ip ospf database	Zobrazí obsah OSPF databáze.
Router#show ip ospf database nssa-external	Zobrazí stavy externích linek do oblastí <i>Not-So-Stubby Area (NSSA)</i> ²⁹ .

Odstraňování závad OSPF

Router#clear ip route *	Vymaže obsah směrovací tabulky a vynutí si její znovu naplnění.
Router#clear ip route a.b.c.d	Smaže cestu do konkrétní sítě a.b.c.d
Router#clear ip ospf counters	Vynuluje čítače OSPF.
Router#clear ip ospf process	Vynuluje celý proces OSPF, vynutí si znovuvytvoření sousedství, databáze a směrovací tabulky.
Router#debug ip ospf events	Zobrazí všechny události OSPF.
Router#debug ip ospf adjacency	Zobrazí jednotlivé stavy OSPF a volby DR/BDR mezi směrovači ve vztahu sousedství.
Router#debug ip ospf packets	Zobrazí pakety OSPF.

²⁹ Netranzitní sítě, do kterých se propagují směrovací informace.

Přehled základních příkazů pro OSPF

Příkaz (Command)	Popis (Description)
Router(config)# router ospf 123	Zapne OSPF s číslem procesu (<i>process number</i>) 123. Identifikátor (ID) procesu je číslo s jakoukoliv hodnotou mezi 1 a 65 535. Číslo procesu není stejné jako oblast OSPF (OSPF area).
Router(config-router)# network 172.16.10.0 0.0.0.255 area 0	<u>OSPF inzeruje (advertise) rozhraní, nikoli síťe.</u> <u>Používá pseudomasku (wildcard mask) k určení,</u> <u>která rozhraní se mají inzerovat.</u> Zobrazený příkaz je třeba číst takto: kterékoliv rozhraní s adresou 172.16.10.x má být vloženo do oblasti 0. (<i>OSPF area 0</i>)
Router(config-if)# ip ospf priority 50	Mění prioritu (<i>priority</i>) OSPF rozhraní na 50.
Router(config-if)# bandwidth 128	Mění šířku pásmo, přenosovou kapacitu (<i>bandwidth</i>) rozhraní na 128 kbps.
Router(config-if)# ip ospf cost 1564	Mění cenu (<i>cost</i>) na hodnotu 1564.
Router(config-if)# ip ospf hello-interval 20	Mění časovač intervalu rozesílání paketů Hello (<i>Hello interval timer</i>) na 20 sekund.
Router(config-if)# ip ospf dead-interval 80	Mění nastavení, jak dlouho se bude čekat na paket Hello před prohlášením, že linka je shozena (<i>Dead interval timer</i>), na 80 sekund.
Router(config)# ip route 0.0.0.0 0.0.0.0 s0/0/0	Vytváří statickou implicitní cestu směřující ven z rozhraní Serial 0/0/0. Tato cesta bude mít administrativní vzdálenost 0.
Router(config-router)# default-information originate	Nastaví, že je implicitní cesta propagována na všechny směrovače OSPF.
Router# show ip protocols	Zobrazí parametry pro všechny, na směrovači běžící, směrovací protokoly.
Router# show ip route	Zobrazí kompletní směrovací tabulku.
Router# show ip ospf	Zobrazí základní informace pro všechny procesy OSPF běžící na směrovači.
Router# show ip ospf interface	Zobrazí informace o OSPF jak jsou vztažené ke všem rozhraním.
Router# show ip ospf neighbor	Zobrazí všechny OSPF sousedy a jejich stavu.

Router# show ip ospf neighbor detail	Zobrazí detailní výpis sousedů.
--------------------------------------	---------------------------------

Cvičení

Základní konfigurace OSPF podle Lab 11.6.1 scénář A, scénář B.

Kontrolní opakovací otázky a odpovědi (kvíz):

- 1) Tři pravdivá tvrzení ohledně směrovacích protokolů typu stav linky:
 - a) jsou všeobecně známé jako protokoly SPF (*Shortest Path First*, nejkratší cesta jako první),
 - b) udržují komplexní databázi síťové topologie,
 - c) jsou založené na algoritmu Dijkstra.
- 2) Spárování termínů a jejich popisů:
 - a) kontaktní pakety hello = vytvářejí a udržují vztah sousedství směrovačů (*adjacency*),
 - b) výměna LSA = je spuštěna, když nastane změna topologie,
 - c) stav linky = popis rozhraní a jeho vztahu s jiným směrovačem,
 - d) algoritmus SPF = vypočítává nejlepší (nejkratší) cestu do cílové sítě.
- 3) Z jakých dvou důvodů by měl administrátor při konfiguraci OSPF používat rozhraní typu zpětná smyčka (*loopback*)?
 - a) Zpětné smyčky jsou logická virtuální rozhraní, která nelze vypnout (*shodit, do not go down*)
 - b) Adresa zpětné smyčky bude použitá jako ID směrovače a „přebije“ IP adresu lokálního rozhraní na směrovači.
- 4) U kterých dvou typů sítí nemůže být zvolen pověřený směrovač OSPF?
 - a) Point-to-point,
 - b) point-to-multipoint.
- 5) Administrátor vložil příkaz „router ospf 100“, jaký je význam čísla 100?
 - a) číslo (identifikátor) procesu OSPF (process ID).
- 6) Smysl příkazu „bandwidth 56“ vloženého na sériovém rozhraní směrovače OSPF?
 - a) Změní hodnotu ceny (*cost*) linky.
- 7) Který faktor bere v úvahu Cisco implementace OSPF při výpočtu ceny linky?
 - a) Bandwidth.

- 8) Propagace implicitní cesty v OSPF:
 - a) default-information originate
- 9) Pokud mají participující směrovače stejnou prioritu, co bude vzato v potaz při volbě DR/BDR v OSPF?
 - a) Router ID.
- 10) Podtrhněte vadný typ paketu pro OSPF: hello, LRU, LSR, LSAck, DBD.

Přílohy

Opakování - příklady na adresaci IPv4

IPv4 má 32 bitovou adresu. Toto binární číslo se zapisuje v dekadickém tečkovém (= kanonickém) zápisu po celých oktetech (bajtech) oddělených tečkou.

Například 10101100.00010000.00000001.00000010 zapíšeme jako 172.16.1.2.

Uvědomte si, že tento kanonický zápis je jakoby v číselné soustavě o základu 256.

(Platí například: 0.0.2.0 – 0.0.0.1 = 0.0.1.255).

Celkový počet všech možných IP adres je $2^{32} = 4\ 294\ 967\ 296$. Maximální rozsah IP adres (všechny možné adresy v IPv4) je 0.0.0.0 až 255.255.255.255. Jiný způsob zápisu tohoto **adresního bloku** je 0.0.0.0/0. To znamená **adresa sítě a /prefix (délka prefixu)**. Prefix (lomítkový tvar) je jiný způsob zápisu masky podsítě. (Takovéto názvosloví je používáno v novém adresním schématu VLSM, ve starším adresním schématu CIDR se naopak prefixem myslí síťová část adresy a o masce v lomítkovém tvaru se mluví jako o délce prefixu.) Prefix vyjadřuje kolik je v masce zleva binárně jedniček. (Například: / 19 = 255.255.224.0.). IP adresa se skládá ze dvou částí: **síťové části a hostitelské části**. Počet bitů v síťové části adresy je daný počtem jedničkových bitů zleva v masce síti (podsítě).

Masku může zapsat také v kanonickém tvaru. Například /19 = 255.255.224.0.

Krajní **meze adresního bloku** se nazývají:

- **adresa sítě** (*network address*) – v hostitelské části adresy jsou binárně samé nuly (např. 172.16.16.16/28)
- **adresa všeměrového vysílání** (*broadcast address*) – v hostitelské části adresy jsou binárně samé jedničky (např. 172.16.16.31/28)

Tyto dvě adresy nelze použít pro adresaci fyzického zařízení (portu).

Počet adres v jedné síti je určen počtem bitů v **hostitelské části** adresy = $2^{(32-\text{prefix})}$

Počet všech možných sítí se stejnou maskou je určen počtem bitů v **síťové části** adresy = $2^{(\text{prefix})}$

Počet stejně velkých podsítí k jedné výchozí síti je určen **počtem vypůjčených bitů** = $2^{(\text{počet vypůjčených bitů})}$

Počet vypůjčených (borrowed) bitů (rozdíl mezi počtem bitů v nové masce podsítě a počtem bitů v masce výchozí sítě) = prefix nové podsítě – prefix výchozí sítě. Ve vypůjčených bitech je přímo obsaženo pořadové číslo podsítě vzhledem k výchozí síti.

Jaké procento obsadí všechny adresy ve třídě A, B, C, D a E vzhledem ke všem možným adresám v protokolu IPv4?

Třída	Počet všech adres ve všech sítích jedné třídy = počet všech TŘÍDNÍCH SÍTÍ (v jedné konkrétní třídě) * ADRES v jedné nich	% vzhledem ke všem možným IPv4 adresám
Celý IPv4	2^{32}	100,00%
A	$2^7 \cdot 2^{24} = 2^{31}$ (1. bit v prvním bitu je fixně 0, zbývá 7 bitů v síťové části IP adresy a je 24 bitů v hostitelské části)	50,00%
B	$2^{14} \cdot 2^{16} = 2^{30}$	25,00%
C	$2^{21} \cdot 2^8 = 2^{29}$	12,50%
D	$2^4 \cdot 2^{24} = 2^{28}$	6,25%
E	$2^4 \cdot 2^{24} = 2^{28}$	6,25%

Z toho vidíme, že třídní adresa neekonomicky plýtvá s dostupnými adresami. Tomu se zabraňuje:

1. Zavedením privátních adres v privátních sítích. (Na hraničním směrovači mezi neveřejnou a veřejnou sítí potom musí být NAT).
2. Beztřídní adresací – tvorbou menších adresních bloků – podsítí (*subnets, subnetting*).

Pro zadanou adresu hostitele (10.65.10.10) a různé masky (/9, /10, /11, /12, /13, /14, 15) vypočtěte adresu sítě, ve které příslušná adresa leží, a adresu všeobecného vysílání téže sítě:

Vypočteme pomocí binárního tvaru IP adresy. Maska leží vždy ve druhém bajtu zleva a ten v tomto případě je $(65)_{10} = (0100\ 0001)_2$

10.65.10.10/9: 10.0.0.0 – 10.127.255.255
 10.65.10.10/10: 10.64.0.0 – 10.127.255.255
 10.65.10.10/11: 10.64.0.0 – 10.95.255.255
 10.65.10.10/12: 10.64.0.0 – 10.79.255.255
 10.65.10.10/13: 10.64.0.0 – 10.71.255.255
 10.65.10.10/14: 10.64.0.0 – 10.67.255.255
 10.65.10.10/15: 10.64.0.0 – 10.65.255.255

Máte IP adresu hostitelského počítače a prefix (masku v lomítkovém (*slash*) tvaru) 172.16.61.210/20. Určete:

- masku v kanonickém tvaru: 255.255.240.0
- velikost bloku adres v kanonickém tvaru: počet bitů v hostitelské části adresy je $32 - 20 = 12$ bitů $\Rightarrow 2^{12} = 2^4 \cdot 2^8 = 0.0.16.0$
- adresa sítě (odmaskováním): $172.16.61.210 \text{ AND } 255.255.240.0 = 172.16.48.0/20$
- adresa všeobecného vysílání je adresa následující sítě zmenšená o 0.0.0.1: $172.16.48.0 + 0.0.16.0 - 0.0.0.1 = 172.16.63.255/20$
- Jiný způsob výpočtu adresy sítě (pomocí velikosti bloku): hranice masky leží ve 3. bajtu a adresa sítě musí být celočíselný násobek velikosti bloku. Nejbližší nižší násobek 16 k 61 je **48** a tedy $172.16.48.0/20$.

Máte zadánu adresu sítě 10.60.0.0 a adresu všeobecného vysílání 10.63.255.255 jedné sítě (jednoho adresního bloku). Určete masku a velikost adresního bloku.

- Rozdíl krajních adres sítě $10.63.255.255 - 10.60.0.0 = 0.3.255.255$. To je číslo, které má binárně samé jednotky v hostitelské části a nazývá se pseudomaska (zástupná maska). Masku

je tedy dvojkový doplněk této inverzní masky (pseudomasky). $255.255.255.255 - 0.3.255.255 = 255.252.0.0 \Rightarrow /14$. Velikost bloku je rovna pseudomasce plus 0.0.0.1. $0.3.255.255 + 0.0.0.1 = 0.4.0.0$. (Jinak řečeno hostitelská část má $32-14=18$ bitů a velikost bloku $2^{18} = 0.4.0.0$.)

Máte zadánu síť 172.16.48.0/20. Kolikátá podsíť to je při zadané adrese a masce vzhledem k plné třídě.

- První bajt je 172, jde tedy o třídu B s implicitní maskou /16. K implicitní masce máme vyplýčeny 4 byty (17. až 20.).
- Třetí bajt je binárně 00110000. Ve vypůjčené části je binárně 0011 což jsou dekadicky 3. Jde tedy o třetí podsítě.
- Jiný postup řešení: velikost bloku při masce /20 je $2^{4*2^8} = 0.0.16.0$ a $172.16.48.0 - 172.16.0.0 = 0.0.48.0 = 3*(0.0.16.0)$. Jde tedy o třetí podsítě.

Máte zadánu síť 172.16.48.0/20. Rozdělte ji alespoň na 3 nové stejně velké podsítě. Určete druhou podsíť z nich. (Číslovat začínáme vždy od 0.) Vejde se do ní 1000 klientů?

- Nová maska: Máme vytvořit alespoň 3 nové sítě. Počet vytvořených podsítí je vždy mocnina základu 2. Takže, nejbližší vyšší mocnina dvou ke třem jsou $4 = 2^2 \Rightarrow$ k původní masce si musíme vypůjčit 2 byty a nová maska podsítě je tedy /20+2=22
- Velikost bloku: $32-22=10$ bitů může adresovat $2^{10} = 2^2*2^8 = 0.0.4.0$ (vejde se do ní 1022 klientů + dvě rezervované adresy).

Adresy sítí a všeobecného vysílání u čtyř vytvořených podsítí budou:

Č.	Rozsah adres	Binární tvar: síťová část (vyp.bity) + hostitelská část
0.	172.16.48.0 – 172.16.51.255	10101100.00010000.001100 <u>00.00000000</u> 10101100.00010000.001100 <u>11.11111111</u>
1	172.16.52.0 – 172.16.55.255	10101100.00010000.001101 <u>00.00000000</u> 10101100.00010000.001101 <u>11.11111111</u>
2	172.16.56.0 – 172.16.59.255	10101100.00010000.001110 <u>00.00000000</u> 10101100.00010000.001110 <u>11.11111111</u>
3	172.16.60.0 – 172.16.63.255	10101100.00010000.001111 <u>00.00000000</u> 10101100.00010000.001111 <u>11.11111111</u>

- Druhá podsíť je $172.16.48.0 + 2*(0.0.4.0) = 172.16.56.0$. Adresy klientů této sítě leží v rozsahu: 172.16.56.1 až 172.16.59.254.
- Ve vypůjčených dvou bitech (21. a 22. bit zleva) je hodnota $(10)_2 = (2)_{10}$, což je přímo pořadové číslo vytvořené podsítě. (Obsah 3. bajtu zleva je $(56)_{10} = (0011\ 1000)_2$.)

Máte zadánu adresu 172.16.25.100/21 v kolikáté podsítě vzhledem k výchozí třídní síti tato adresa leží? (Předpokládáme adresní schéma CIDR – je podsítována třídní adresa a všechny podsítě jsou stejné.)

- výchozí třídní síť je ve třídě B a má implicitní masku /16,
- obsah 3. bajtu je $(25)_{10} = (\underline{0001}\ \underline{1001})_2$, ve vypůjčených bitech je $(00011)_2 = (3)_{10}$, tedy leží ve třetí podsítě,
- $172.16.25.100 - 172.16.0.0 = 0.0.25.100$, velikost bloku je 0.0.8.0, adresa leží ve 3. podsítě

172.16.24.0/21 – 172.16.31.255/21.

Máte zadánu výchozí síť 172.16.16.0/20 (jde už o podsíť plné třídy). Začínající administrátor učeň ji podsíťoval (na stejně velké podsítě) maskou /25. Na portu směrovače zkouší nastavit adresu 172.16.31.127/25. V kolikáté podsíti tato adresa leží? A proč se mu nedáří nastavit tuto adresu?

- V binárním tvaru zobrazíme v posledních 2B vypůjčené bity: 0001 1111. 0111 1111. $(11110)_2 = (30)_{10}$ Jde tedy o 30. podsíť.
- Velikost bloku je pro prefix /25 rovna 0.0.0.128. Adresa sítě je 172.16.31.127 AND 255.255.255.128 = 172.16.31.0. Následující síť je $172.16.31.0 + 0.0.0.128 = 172.16.31.128$. Z toho vidíme, že adresa 172.16.31.127 je všesměrová adresa.

Máte zadány dvě adresy 172.16.2.3/22 a 172.16.3.2/24. Můžete je použít na síťových rozhraních jednoho směrovače, to znamená nepřekrývají se sítě, ve kterých ty dvě adresy leží?

- Rozsahy adres sítí, ve kterých zadané adresy leží, jsou 172.16.0.0-172.16.3.255/22 a 172.16.3.0-172.16.3.255/24, sítě se překrývají a nelze je proto použít na jednom směrovači najednou.

Máte zadány dvě adresy, které chce vložit jako adresy síťových rozhraní na jednom směrovači: 172.16.1.100/23 a 172.16.3.10/22. Překrývají se sítě, ve kterých leží?

172.16.0.0 – 172.16.1.255 velikost bloku je 0.0.2.0

172.16.0.0 – 172.16.3.255 velikost bloku je 0.0.4.0

Sítě se překrývají a nelze je proto použít na jednom směrovači. Tuto chybu odhalí operační systém směrovače. Ale POZOR adresy nelze použít ani v jedné skupině sítí (směrovací doméně). Tato chyba je ale zákeřnější, protože ji na rozdíl od předchozí operační systém směrovače neodhalí a zapojení „z neznámých příčin“ nefunguje.

Je adresa 172.32.1.1 neveřejná (privátní)?

- Není. Nejbližší adresní blok neveřejných adres je 172.16.0.0-172.31.255.255 to jest 172.16.0.0/12. Zadaná adresa leží mimo tento rozsah. Všimněte si, že rozsah privátních adres je tvořen 16 třídními bloky ve třídě B. Představuje tedy jednu nadsíť (*supernet*) pro 16 třídních bloků 172.16.0.0/16 až 172.31.0.0/16.

Cisco NetAcad: CCNA Exploration - Routing Protocols and Concepts – studijní materiály

Vztah délky prefixu a velikosti bloku

Ověřte si pochopení látky:

1. Které z následujících jsou adresy sítě? (Vyberte dvě.)
 - a) 64.104.3.7/28
 - b) 192.168.12.64/26
 - c) 192.135.12.191/26
 - d) 198.18.12.16/28
 - e) 209.165.200.254/27
 - f) 220.12.12.33/27
2. Administrátor sítě vytváří síť pro malou firmu, která má 22 hostitelských počítačů. ISP přiřadil pouze jednu IP adresu směrovatelnou do Internetu. Který adresní blok může administrátor použít pro adresaci této sítě?
 - a) 10.11.12.16/28
 - b) 172.31.255.128/27
 - c) 192.168.1.0/28
 - d) 209.165.202.128/27
3. Která maska podsítě může být použita na hostitelské počítači v síti 128.107.176.0/22?
 - a) 255.0.0.0
 - b) 255.248.0.0
 - c) 255.255.252.0
 - d) 255.255.255.0
 - e) 255.255.255.252
4. Pro vytvoření dvoubodového WAN spojení (*point-to-point*) vám by vám přidělen adresní blok 10.255.255.224/28. Kolik takových sítí WAN může být v tomto bloku adres?
 - a) 1
 - b) 4
 - c) 7
 - d) 14
5. Co definuje jednu logickou IP síť?
6. Pojmenujte a popište účel tří typů adres IPv4:
7. Administrátor sítě potřebuje vytvořit novou síť, která má 14 počítačů a dvě síťová rozhraní na směrovači. Která maska podsítě poskytne odpovídající počet adres s minimálním plýtváním adresami.
 - a) 255.255.255.128

- b) 255.255.255.192
 - c) 255.255.255.224
 - d) 255.255.255.240
 - e) 255.255.255.248
 - f) 255.255.255.252
8. Co rozlišuje každý ze tří typů adres IPv4?
 9. Napište seznam tří forem komunikace IPv4.
 10. Napište důvod proč jsou definovány specifické rozsahy IPv4 adres pro veřejné a pro privátní použití.
 11. Hostitelský počítač z jižní pobočky firmy nemůže přistupovat k serveru s adresou 192.168.254.222/24. Během prozkoumávání hostitelského počítače jste zjistili, že má IPv4 adresu 169.254.11.15/16. Co je očividný problém?
 - a) hostitelský počítač používá adresu lokální linky (local-link)
 - b) server používá vadnou masku podsítě
 - c) hostitelský počítač má přiřazenou adresu všesměrového vysílání
 - d) server si myslí, že hostitelský počítač je v jedné logické síti s tímto serverem.
 12. Vypište tři důvody pro plánování a dokumentaci adres v síti.
 13. Uveďte příklady zařízení, kde by měl administrátor přiřazovat IPv4 adresy staticky.
 14. Co je primární motivací pro vývoj a zavádění protokolu IPv6.
 15. Jaký je účel masky podsítě v adresaci IPv4?
 16. Vypište faktory, které by se měli vzít v úvahu při plánování adresního schéma IPv4.
 17. Které jsou tři testy pomocí služebního programu (utility) ping pro ověření funkčnosti hostitelského počítače v síti?
 18. Které jsou to rezervované a speciální IPv4 adresy a jak se používají?
 19. Proč je protokol ICMPv4 důležitý ve vztahu k činnosti IPv4? Jaké jsou typy zpráv ICMP?

Zabezpečení sítě pomocí přístupových seznamů IP

(Pro ty, kdo chtejí vědět více. Zde uvádím pouze základní nastavení a podrobněji bude probráno ve čtvrtém semestru CCNA Exploration.)

V této kapitole se naučíme:

- Základní informace o přístupových seznamech (*Access Control List, ACL*)
- Čísla přístupových seznamů ACL (*Access Control List, ACL*)
- Použití zástupných (pseudo)masek
- Klíčová slova pro ACL
- Vytvoření standardního přístupového seznamu
- Aplikace standardního přístupového seznamu na rozhraní
- Ověření funkčnosti ACL
- Odstranění ACL
- Vytvoření rozšířeného ACL
- Aplikace rozšířeného ACL na rozhraní
- Klíčové slovo „**established**“ (nepovinné)
- Vytvoření pojmenovaného ACL
- Pořadová čísla řádků v pojmenovaném ACL
- Odstranění konkrétních řádků z pojmenovaného ACL
- Tipy pro číslování řádků
- Komentáře k řádkům ACL
- Omezení přístupu k virtuálnímu terminálu

Základní informace o přístupových seznamech

Přístupové seznamy (*Access Control List, ACL*) se používají k zabezpečení sítí a řízení provozu do a ze sítě. Přístupové seznamy (ACL) filují provoz na základě pravidel, které můžete nastavit v příkazech (jednotlivých řádcích) svého ACL. Tato pravidla určují, zda pakety jsou povoleny nebo zakázány, jaké služby mají možnost používat, a kdo s kým může komunikovat. Příkladem toho je například, zda má hostitel povolen přístup k Internetu nebo má přístup k určitému serveru v síti.

Přístup ke službám je filtrován na základě čísel portů. Porty 0 až 1023 se nazývají dobře známé porty. Patří mezi ně běžné služby, jako Telnet s portem 23 a HTTP, který používá port 80. Firmy vyvíjející SW mohou požádat organizaci IANA o přidělení čísla portu k identifikaci konkrétní aplikace v rozmezí čísla portu 1024 až 49151. Například: Shockwave používá číslo portu 1626. Porty 49 152 až 65 535 jsou přiřazována dynamicky koncovým zařízením a jsou dočasné, tj. trvají pouze po dobu trvání spojení.

Když je nakonfigurován, změní ACL router na firewall a testuje veškerý provoz proti každé řádce seznamu před tím, než mohou být předány do jejich místa určení. Tento proces řídí síťový provoz a pomáhá chránit vaši síť, ale rozhodně přidává latenci. Pakety jsou kontrolovány proti řádkům - příkazům ACL v pořadí, ve kterém jsou nakonfigurovány, od shora dolů, jednotlivý příkaz (řádek) najednou. Při prvním výskytu shody se zadanou podmínkou, podle toho, zda je provoz povolen (*permit*) či zakázán (*deny*), je příslušná akce provedena. Jestliže je každý příkaz akce povolení (*permit*), je na konci seznamu příkazů implicitní „zákaz všeho“ ("*deny any*"), který ale není zobrazován a není ho ani třeba nakonfigurovat. Jakýkoliv paket, který neodpovídá žádnému příkazu s povolením provozu, je potom automaticky odmítnut. Proto, pokud jsou všechny příkazy akce odmítnutí

(deny), musí být vložen jako poslední příkaz „povolení všeho“ ("permit any"), jinak je veškerý provoz zakázán! To je velmi častý omyl, který dělají správci sítě - nováčci.

Standardní ACL jsou jednoduché příkazy, které provoz povolují nebo zakazují na základě zdrojové IP adresy. Měly by být nastaveny na routeru tak blízko k cíli, jak jen to je možné.

Rozšířenými ACL lze filtrovat provoz pomocí více proměnných, jako jsou protokol, zdrojová a cílová IP adresa a číslo portu, na základě čeho je příslušná služba nebo aplikace filtrována. Protože tyto rozšířené ACL jsou přesné, jsou nakonfigurovány na routeru co nejbližše ke zdroji který je filtrován. Toto zabraňuje odepření provozu z důvodu spotřeby přenosové kapacity.

Standardní a rozšířené seznamy ACL mohou být nakonfigurovány buď jako **pojmenované** nebo jako **číslované**. ACL obecně mají dano číslo identifikující jejich typ - 1 až 99 pro standardní IP a 100 až 199 pro rozšířené IP ACL. Pojmenované ACL nemají žádná omezení, ale co je důležitější, mohou být snadno změněny bez nutnosti začínat konfiguraci znova od samého začátku. Když chcete přidat příkaz do prostředka takovéhoto seznamu, lze použít Pořadová čísla řádek, aniž by bylo třeba začít celou konfiguraci znova od začátku. Jak již bylo uvedeno, pakety jsou vyhodnocovány proti řádkům přístupovému seznamu v pořadí, ve kterém byly řádky vytvořeny. To znamená, že pokud uděláte chybu a dáte jako první příkaz, který by měl být jako poslední, nelze ho jenom jednoduše odstranit, ale musíte začít konfigurovat od začátku. To je důvod, proč je doporučeno napsat si svůj ACL v textovém editoru a nechat ho někým zkontolovat ještě před tím, než ho vložíte do vaší konfigurace. Používáte-li pojmenovaného ACL, nejste omezeni počtem příkazů, které můžete vytvářet, a také vám to umožní vyladit konfigurace ACL bez nutnosti celého jeho odstranění a začínání znova.

Po vytvoření přístupového seznamu, který slouží svému účelu, je dalším a posledním krokem jeho aplikace na rozhraní. Pro to, aby ACL pracoval, musíte ho aplikovat na rozhraní a to buď v příchodním (in) nebo odchozím (out) směru. Bez toho je ACL k ničemu a je to totéž jako nemít vůbec žádné zabezpečení.

Pro jeden směrovaný protokol lze mít aplikován **jeden ACL na jednom rozhraní a jednom směru**.

Čísla ACL

1–99 nebo 1300–1999	Standardní IP (Standard IP)
100–199 nebo 2000–2699	Rozšířený IP (Extended IP)
600–699	AppleTalk
800–899	IPX
900–999	Rozšířený IPX (Extended IPX)
1000–1099	IPX Service Advertising Protocol

Zástupné masky

Zástupná maska, pseudomaska (*wildcard mask*) určuje, které části IP adresy se při rozhodování musí shodovat, aby se na ně aplikovalo pravidlo **permit (povolit)** nebo **deny (zakázat)** v jednom příkazu (jedné řádce) přístupového seznamu (ACL):

- **0 (nula)** v zástupné masce znamená, že odpovídající bit v adrese je kontrolován a že se musí přesně shodovat.
- **1 (jednička)** v zástupné masce znamená, že odpovídající bit v adrese je ignorován a může být 1 nebo 0.

Příklad 1: 172.16.0.0 0.0.255.255

```

172.16.0.0 = 10101100.00010000.00000000.00000000
0.0.255.255 = 00000000.00000000.11111111.11111111
-----
výsledek = 10101100.00010000.xxxxxxxx.xxxxxxxx
172.16.x.x (Cokoliv mezi 172.16.0.0 a 172.16.255.255
bude odpovídat uvedenému příkazu v příkladu.)

```

TIP: Oktet složený ze samých nul v masce znamená, že se musí oktet v adrese přesně shodovat. Oktet složený ze samých jedniček v masce znamená, že příslušný oktet v adrese může být celý ignorován.

Příklad 2: 172.16.8.0 0.0.7.255

```

172.168.8.0 = 10101100.00010000.00001000.00000000
0.0.0.7.255 = 00000000.00000000.00000111.11111111
-----
výsledek = 10101100.00010000.00001xxx.xxxxxxxx
00001xxx = 00001000 až 00001111 = 8-15
xxxxxxx = 00000000 až 11111111 = 0-255

```

Cokoliv mezi 172.16.8.0 až 172.16.15.255 bude vyhovovat uvedenému příkazu.

Klíčová slova pro ACL

any	Používá se místo výrazu 0.0.0.0 255.255.255.255, při porovnání vyhovuje jakékoli IP adresy
host	Používá se v zástupné masce místo výrazu 0.0.0.0, při porovnání tedy vyhoví pouze jedna konkrétní adresa.

Vytvoření standardního ACL

Router(config)#access-list 10 permit 172.16.0.0 0.0.255.255	Čteme jako: „Všechny pakety se zdrojovou IP adresou 172.16.x.x budou mít povolen další průchod sítí.“
access-list	Příkaz pro vytvoření ACL
10	Libovolné číslo mezi 1 až 99, nebo 1300 až 1999, definuje příslušný ACL jako standardní IP ACL
permit	Pakety, které vyhovují tomuto příkazu, mají povolen pokračovat v průchodu.
172.16.0.0	Zdrojová IP adresa, která bude porovnávána
0.0.255.255	Zástupná maska

Router(config)#access-list 10 deny host 172.17.0.1	Čteme jako: „Všechny pakety se zdrojovou IP adresou 172.17.0.1 budou vyřazeny a zahozeny.“
access-list	Příkaz pro vytvoření ACL
10	Libovolné číslo mezi 1 až 99, nebo 1300 až 1999, definuje příslušný ACL jako standardní IP ACL
deny	Pakety, které vyhovují tomuto příkazu, budou vyřazeny a zahozeny.
Host	Klíčové slovo
172.17.0.1	Adresa konkrétního hostitele

Router(config)#access-list 10 permit any	Čteme jako: „Všechny pakety s jakoukoliv zdrojovou IP adresou budou mít povolen další průchod sítí.“
access-list	Příkaz pro vytvoření ACL
10	Libovolné číslo mezi 1 až 99, nebo 1300 až 1999, definuje příslušný ACL jako standardní IP ACL
permit	Pakety, které vyhovují tomuto příkazu, mají povoleno pokračovat v průchodu.
Any	Klíčové slovo, které znamená jakákoliv IP adresa

- **TIP:** Na konci každého ACL je pevně zakódován implicitní příkaz **deny**. Nevidíte ho sice, ale přikazuje „zakaž vše co dosud nebylo povoleno“. Je to vždy poslední řádka každého ACL. Pokud tomuto implicitnímu zákazu chcete zabránit, vložte na poslední řádku standardního ACL příkaz **permit any** nebo v případě rozšířeného ACL vložte příkaz **permit ip any any**.

Aplikace standardního ACL na rozhraní

Router(config)#interface fastethernet 0/0	Přejde do konfiguračního režimu rozhraní
Router(config-if)#ip access-group 10 in	Vezme všechny řádky ACL, které jsou definovány jako části skupiny 10, a aplikuje na na příchozí směr. Pakety přicházející na směrovač na rozhraní fastethernet 0/0 budou zkонтrolovány.

- **TIP:** ACL mohou být aplikovány buď na příchozí směr (klíčové slovo **in**) nebo na odchozí směr (klíčové slovo **out**).
- **TIP:** Pro jeden protokol (IP), na jednom rozhraní a na jeden směr (dovnitř/ven) je možné aplikovat pouze jeden ACL.
- **TIP:** Standardní ACL aplikujte co nejblíže je to možné k cílové síti nebo zařízení.

Kontrola ACL

Router#show ip interface	Zobrazí všechny ACL aplikované na zadané rozhraní
Router#show access-lists	Zobrazí obsah všech ACL na směrovači včetně počtu spárování jednotlivých příkazů s pakety.
Router#show access-list <i>access-list-number</i>	Zobrazí obsah ACL se zadaným číslem
Router#show access-list <i>name</i>	Zobrazí obsah ACL se zadaným jménem
Router#show run	Zobrazí všechny ACL a jejich přiřazení k rozhraním

Odstranění ACL

Router(config)#no access-list 10	Odstraní všechny ACL s číslem 10
----------------------------------	----------------------------------

Vytvoření rozšířeného ACL

Router(config)#access-list 110 permit tcp 172.16.0.0 0.0.0.255 192.168.100.0 0.0.0.255 eq 80	Čteme jako: „HTTP pakety se zdrojovou IP adresou 172.16.0.x budou mít povolen další průchod do cílové adresy 192.168.100.x.”
access-list	Příkaz pro vytvoření ACL
110	Libovolné číslo mezi 100 až 199, nebo 2000 až 2699, definuje příslušný ACL jako rozšířený IP ACL.
permit	Pakety, které vyhovují tomuto příkazu, mají povoleno pokračovat v průchodu.
tcp	Protokol musí být TCP.
172.16.0.0	Zdrojová IP adresa, která bude porovnávána.
0.0.0.255	Zástupná maska pro zdrojovou IP adresu.
192.168.100.0	Cílová IP adresa, která bude porovnávána.
0.0.0.255	Zástupná maska pro cílovou IP adresu.
eq	Operátor, který znamená „rovná se“
80	Port 80, indikující provoz HTTP.

Router(config)#access-list 110 deny tcp any 192.168.100.7 0.0.0.0 eq 23	Čteme jako: „Telnet pakety s jakoukoliv zdrojovou IP adresou budou vyřazeny, jestliže jsou adresovány do konkrétního hostitele 192.168.100.7.”
access-list	Příkaz pro vytvoření ACL
110	Libovolné číslo mezi 100 až 199, nebo 2000 až

	2699, definuje příslušný ACL jako rozšířený IP ACL.
deny	Pakety, které vyhovují tomuto příkazu, budou vyřazeny a zahozeny.
tcp	Protokol musí být TCP.
any	Jakákoli zdrojová IP adresa.
192.168.100.7	Cílová IP adresa, která bude porovnávána.
0.0.0.0	Zástupná maska; adresa musí přesně souhlasit.
eq	Operátor znamenající „rovná se“.
23	Port 23, indikující provoz Telnet.

Aplikace rozšířeného ACL na rozhraní

```
Router(config)#interface fastethernet 0/0 Router(config-if)#ip access-group 110 out
```

Přepne do konfiguračního režimu rozhraní a vezme všechny řádky ACL, které jsou definovány jako část skupiny 110, a aplikuje je na odchozí směr. Pakety opouštějící rozhraní fastethernet 0/0 budou zkонтrolovány.

- **TIP:** ACL mohou být aplikovány buď na příchozí směr (klíčové slovo **in**) nebo na odchozí směr (klíčové slovo **out**).
- **TIP:** Pro jeden protokol (IP), na jednom rozhraní a na jeden směr (dovnitř/ven) je možné aplikovat pouze jeden ACL.
- **TIP:** Rozšířený ACL aplikujte co nejbliže je to možné ke zdrojové síti nebo zařízení.

Klíčové slovo „established“ (nepovinné)

```
Router(config)#access-list 110 permit tcp 172.16.0.0 0.0.0.255 192.168.100.0 0.0.0.255 eq 80 established
```

Indikuje již navázané (*established*) spojení.

- **POZNÁMKA:** Spárování (splnění podmínky) nyní nastane pouze pokud má datagram TCP nastavený bit ACK nebo RST.
- **TIP:** Klíčové slovo „**established**“ bude funkční (a má smysl) pouze pro protokol TCP a nikoliv pro UDP (který je nespojovaný).
- **TIP:** Uvažte následující situaci: chcete zabránit hackerům zneužít port 80 pro vniknutí do vaší sítě. Protože neprovozujete žádný Web server zablokovat příchozí provoz na portu 80 ovšem s výjimkou kdy vnitřní uživatelé potřebují přístup na Web. Při jejich požadavku na Web je nutné povolit návratový provoz na port 80. Řešením je použití příkazu **established**. ACL nyní povolí vstup odpovědi do vaší sítě, protože bude mít nastavený ACK bit jako výsledek prvního požadavku zevnitř vaší sítě. Požadavky zvenčí budou blokovány, protože ACK bit nebude nastaven, ale odpověďm bude povolen průchod.

Vytvoření pojmenovaného (named) ACL

Router(config)#ip access-list extended serveraccess	Vytvoří pojmenovaný rozšířený ACL s názvem <i>serveraccess</i> a přejde do konfiguračního režimu pojmenovaného ACL.
Router(config-ext-nacl)#permit tcp any host 131.108.101.99 eq smtp	Povolí průchod paketů poštovních paketů SMTP z libovolného zdroje do hostitele 131.108.101.99.
Router(config-ext-nacl)#permit udp any host 131.108.101.99 eq domain	Povolí průchod paketů DNS z libovolného zdroje do hostitele 131.108.101.99.
Router(config-ext-nacl)#deny ip any any log	Zamítne všechny ostatní pakety jdoucí kamkoliv. Jestliže bude paket zamítnut, bude zaprotokolován (log) pro pozdější prohlídku.
Router(config-ext-nacl)#exit	Návrat do globálního konfiguračního režimu.
Router(config)#interface fastethernet 0/0 Router(config-if)#ip access-group serveraccess out	Přejde do konfiguračního režimu rozhraní a aplikuje tento ACL na rozhraní fastethernet 0/0 v odchozím směru.

Použití pořadového čísla řádky v pojmenovaném ACL

Router(config)#ip access-list extended serveraccess2	Vytvoří pojmenovaný rozšířený ACL s názvem <i>serveraccess2</i> .
Router(config-ext-nacl)#10 permit tcp any host 131.108.101.99 eq smtp	Pro tuto řádku použije pořadové číslo 10.
Router(config-ext-nacl)#20 permit udp any host 131.108.101.99 eq domain	Řádek s pořadovým číslem 20 se zařadí za řádku 10.
Router(config-ext-nacl)#30 deny ip any any log	Řádek s pořadovým číslem 30 se zařadí za řádku 20.
Router(config-ext-nacl)#exit	Návrat do globálního konfiguračního režimu.
Router(config)#interface fastethernet 0/0	Přejde do konfiguračního režimu rozhraní.
Router(config-if)#ip access-group serveraccess2 out	Aplikuje ACL na odchozí směr tohoto rozhraní.
Router(config-if)#exit	Návrat do globálního konfiguračního režimu.
Router(config)#ip access-list extended serveraccess2	Přejde do konfiguračního režimu pojmenovaného ACL se jménem <i>serveraccess2</i> .
Router(config-ext-nacl)#25 permit tcp any host 131.108.101.99 eq ftp	Pořadové číslo 25 zařadí tuto řádku mezi řádky 20 a 30.
Router(config-ext-nacl)#exit	Návrat do globálního konfiguračního režimu.

- TIP:** Pořadová čísla se používají, aby umožnila snazší editaci přístupového seznamu. V předchozím příkladu byla na řádcích ACL použita pořadová čísla 10, 20 a 30. Pokud byste

chtěli přidat další řádku, přidala by se za poslední řádku číslo 30. Pokud byste chtěli jít více nahoru, museli byste smazat celý ACL a potom znova použít čísla řádek ve správném pořadí. Nyní můžete vložit novou řádku s pořadovým číslem přímo na správné místo.

- **POZNÁMKA:** Argument *sequence-number* byl přidán v Cisco IOS Release 12.2(14)S. Byl plně integrován do Cisco IOS Release 12.2(15)T.

Odstranění řádky v pojmenovaném ACL s použitím čísla řádky

Router(config)#ip access-list extended serveraccess2	Přejde do režimu konfigurace pojmenovaného ACL s názvem <i>serveraccess2</i>
Router(config-ext-nacl)#no 20	Smaže řádku 20 ze seznamu.
Router(config-ext-nacl)#exit	Návrat do globálního konfiguračního režimu.

Tipy pro číslování řádek

- Pořadová čísla začněte od 10 a přidávejte na každé další řádce číslo o 10 více.
- Pokud zapomenete připsat pořadové číslo, je řádka přidána na konec seznamu.
- Při restartu směrovače se pořadová čísla přečíslují, aby odpovídala zásadám ikrementace po 10 (tip 1). Pokud jste v ACL měli čísla 10, 20, 30, 32, 40, 50 a 60, po restartu tato čísla budou 10, 20, 30, 40, 50, 60, 70.
- Ve výstupech příkazů Router#show running-config nebo Router#show startup-config se pořadová čísla řádků nezobrazují. Vypsat čísla řádků ACL lze následujícími příkazy:
 - Router#show access-lists
 - Router#show access-lists list name
 - Router#show ip access-list
 - Router#show ip access-list list name

Komentáře k řádkům v ACL

Router(config)#access-list 10 remark only Jones has access	Příkaz remark dovolí vložení komentáře (omezeného na 100 znaků).
Router(config)#access-list 10 permit 172.16.100.119	Tuto řádku čteme jako: „Hostitel 172.16.100.119 má povolen průchod sítí“.
Router(config)#ip access-list extended telnetaccess	Vytvoří pojmenovaný ACL s názvem <i>telnetaccess</i> a přejde do režimu konfigurace pojmenovaného ACL.
Router(config-ext-nacl)#remark do not let Smith have telnet	Příkaz remark dovolí vložení komentáře (omezeného na 100 znaků).
Router(config-ext-nacl)#deny tcp host 172.16.100.153 any eq telnet	Tuto řádku čteme jako: „Tento konkrétní hostitel 172.16.100.153 má zakázán přístup Telnetem do libovolného místa v síti“.

- **TIP:** Příkaz **remark** můžete použít do libovolného standardního IP, číslovaného rozšířeného nebo pojmenovaného IP ACL.
- **TIP:** Příkaz **remark** můžete použít buď před nebo po příkazové sekvenci **permit** nebo **deny**. Proto buďte konzistentní v umisťování, abyste předešli zmatku, ke které řádce se ten který

komentář vztahuje.

Omezení přístupu k virtuálnímu terminálu

Router(config)#access-list 2 permit host 172.16.10.2	Hostiteli 172.16.10.2 povolí přístup k tomuto směrovači Telnetem v závislosti na tom, kde je tento ACL aplikován.
Router(config)#access-list 2 permit 172.16.20.0 0.0.0.255	Povolí komukoliv z rozsahu adres 172.16.20.x přístup k tomuto směrovači Telnetem v závislosti na tom, kde je tento ACL aplikován.
	Implicitní příkazová řádka deny any zakáže komukoliv dalšímu přístup Telnetem.
Router(config)#line vty 0 4	Přejde do režimu konfigurace linky vty.
Router(config-line)#access-class 2 in	Aplikuje tento ACL na všech 5 virtuálních rozhraní v příchozím směru.

- **TIP:** Pro omezení přístupu virtuálním rozhraním vty Telnet používejte příkaz **access-class** místo příkazu **access-group**, který slouží k aplikaci ACL na fyzické rozhraní.

Rychlé zopakování druhého semestru (Cram Sheet)

Posloupnost zavádění OS (BOOT Sequence) pro směrovač/přepínač

1. POST – Zařízení vyhledá HW a vykoná kontrolu HW
2. Nalezení OS
3. Zavedení (*Load*) OS
4. Nalezení konfigurace (startup-config)
5. Zavedení této konfigurace do RAM (running-config)

Nastavení konfiguračního registru

- 0x2102 (implicitní, *default*): Zkontroluje v NVRAM příkazy "boot system", pokud nejsou zavede první platný IOS z paměti Flash
- 0x2100: Zavede z ROM režim ROM Monitor (ROMMON).
- 0x2101: Zavede režim ROM RxBoot. RxBoot se může připojit k serveru TFTP a stáhnout IOS do paměti Flash.
- 0x2142: Během zavádění ignoruje počáteční konfiguraci z NVRAM – používá se pro obnovu ztraceného hesla (*password recovery*).

Paměti směrovače/přepínače

- **ROM**: obsahuje základní mikrokód pro start a údržbu zařízení Power on Self Test (POST), bootstrap (zavaděč), ROM Monitor (ROMMON), RXBOOT
- **Flash memory**: ukládá obrazy IOS
- **NVRAM**: ukládá *startup-config* (konfigurace zaváděná po startu systému)
- **RAM**: operační paměť obsahuje běžící IOS a běžící konfiguraci *running-config* (aktivní konfigurace po startu)

Zabezpečení směrovače

Pro konfiguraci hesla na 5-ti linkách Telnet, bude konfigurace podobná následující:

```
Router(config)# line vty 0 4
Router(config-line)# password cisco
Router(config-line)# login
```

Konfigurace SSH

Pro konfiguraci SSH na směrovači či přepínači potřebujete následující prvky:

- *hostname*
- *domain name*
- *RSA key*
- *username* a *password* pro lokální autentizaci

Vzorová konfigurace SSH

```
Switch>enable
Switch#configure terminal
Switch(config)#hostname Branch_2960
Branch_2960(config)#ip domain-name Example.net
Branch_2960(config)#crypto key generate rsa
Branch_2960(config)#username admin password ciscocisco
Branch_2960(config)#line vty 0 4
Branch_2960(config-line)#login local
Branch_2960(config-line)transport input ssh
Branch_2960(config-line)#exit
```

Směrování

Implicitní administrativní vzdálenosti (AD) směrovacích protokolů

Protocol	AD
Connected Interface	0
Static Route	1
EIGRP Internal	90
IGRP	100
OSPF	110
RIP	120
EIGRP External	170

Statická trasa (Static Route)

```
Router(config)#ip route 192.168.1.0 255.255.255.0 10.1.1.1
```

Syntaxe implicitní cesty:

```
Router(config)#ip route 0.0.0.0 0.0.0.0 192.168.1.1
```

Směrovací protokoly typu vektor vzdálenosti (Distance Vector Routing Protocols)

Inzerují celou směrovací tabulkou přilehlým (přímo připojeným) sousedům a posílají aktualizace bez ohledu na to, zda nastala změna topologie sítě (periodicky každých X sekund). RIPv1, RIPv2, IGRP, EIGRP.

Směrovací protokoly typu stav linky (Link State Routing Protocols)

- Posílají aktualizace obsahující stavy jejich vlastních linek ke všem ostatním směrovačům v síti. Příklady těchto protokolů: OSPF, ISIS.
- Automaticky spouští výměnu oznamovačů vyvolanou změnou topologie sítě.
- Vytváří a udržuje topologickou databázi z kontaktních paketů (*hello*) a oznamovačů LSA

(*Link State Advertisements*) přijatých z ostatních směrovačů.

- Vypočítávají cesty do každého cíle z topologické databáze a vkládají nejlepší z nich do své směrovací tabulky.

Classful (FLSM³⁰) Versus Classless (VLSM)

- Třídní - *Classful* (RIPv1, IGRP): neinzerují masky podsítí
- Beztrídní - *Classless* (RIPv2, IS-IS, OSPF, EIGRP): inzerují masky podsítí

Sumarizace cest (Route Summarization)

Sumarizace/agregace/nadsíťování cest (*Route summarization/aggregation/supernetting*) reprezentuje několik sítí/podsítí jako jednu větší síťovou adresu, zkrácením masky podsítě, která zahrnuje pouze „společné“ byty ze všech těchto sítí.

RIP

Syntaxe: všechny přilehlé (přímo připojené) třídní sítě:

```
Router(config)#router rip
Router(config-router)#network 192.168.111.0
Router(config-router)#network 192.168.165.0
```

EIGRP

- Proprietární protokol firmy Cisco.
- EIGRP: rychlá konvergence, podpora VLSM. Podpora více směrovaných protokolů: IP, IPX, AppleTalk. Pro každý tento protokol EIGRP udržuje tabulku směrovací, topologickou a sousedů.
- Metrika EIGRP je stejná jako u IGRP, ale je 32-bitová proti 24-bitové metrice u IGRP.
- *Successor route* je nejlepší cesta, která je vložena do směrovací tabulky. *Feasible successor* je záložní cesta v topologické tabulce.
- Maximální počet přeskoků EIGRP = 224
- Konfigurace EIGRP pro síť 192.168.16.0/24 ve směrovací doméně 100:
- Router(config)#router eigrp 100
- Router(config- router)#network 192.168.16.0 0.0.0.255

OSPF

- Protokol rozšířitelný pro velké sítě (neomezený počet přeskoků), neproprietární (*vendor-neutral*), typu stav linky (*link-state*), podpora VLSM.

Oblasti OSPF:

- Číslo oblasti může být libovolné číslo v rozsahu 0 to 65535.
- Oblast 0 je páteřní oblast (*backbone area*).

Kritéria automatické přiřazení identifikátoru směrovače *RouterID* u OSPF:

- Nejvyšší IP adresa na rozhraní loopback (logické, virtuální rozhraní).
- Jestliže není loopback, potom nejvyšší IP adresa na fyzickém rozhraní.

Pověřený a záložní pověřený směrovač (DR/BDR) se volí pouze v následujících topologiích:

30 Fixed-Length Subnet Mask – všechny masky podsítě jsou se stejnou délkou

- Síť se všesměrovým vysíláním a více branami (*Broadcast multi-access*) (například Ethernet)
- Síť nepodporující všesměrové vysílání s více branami (*Non-broadcast multi-access*) (například Frame Relay)

Konfigurace OSPF pro síť 192.168.16.0/24 v oblasti 0:

```
Router(config)#router ospf 7
Router(config-router)#network 192.168.16.0 0.0.0.255 area 0
```

Metrika OSPF = cena (*Cost*). Výpočet metriky - ceny (*Cost*) = $10^8/\text{bandwidth v b/s}$.

Přístupové seznamy (Access Lists)

- Na konci je implicitní zákaz všeho (*deny any*): každý přístupový seznam musí mít nejméně jednu povolující rádku (*permit*), jinak zakáže všechn provoz.
- **Standardní přístupové seznamy pro IP** (*Standard IP access lists*) filují celý protokol IP na základě IP adresy zdrojové sítě. Rozsah čísel 1-99. Umisťují se co nejblíže k cíli.
- **Rozšířené přístupové seznamy pro IP** (*Extended IP access lists*) filují na základě IP adresy zdrojové sítě, IP adresy cílové sítě, specifických protokolů (TCP, UDP, ICMP ...) a čísla portu. Umisťují se co nejblíže ke zdroji.
- Jeden přístupový seznam na jeden směr, jeden protokol a jedno rozhraní.
- Zástupná maska, pseudomaska (*Wildcard mask*):
 - nuly – příslušná pozice bitu se v adrese porovnává,
 - jedničky – příslušná pozice bitu se v adrese ignoruje.

Syntaxe rozšířeného přístupového seznamu:

```
access-list list# [permit | deny] [protocol] [source IP] [Wcmask] [dest. IP]
[Wcmask] [operator] [operand]
```

Použitá literatura

- Kolektiv: Online kurikulum CCNA Exploration – Routing Protocols and Concepts verze 4.0 (aktuální verze pro registrované uživatele je dostupná na portálu cisco.netacad.net)
- Kolektiv: Course Booklet CCNA Exploration – Routing Protocols and Concepts verze 4.0, Cisco Press 2009
- Prezentace PowerPoint k jednotlivým kapitolám kurikula (pro registrované instruktory jsou dostupné na portálu cisco.netacad.net)
- GRAZANI, Rick a JOHNSON, Allan: CCNA Exploration Companion Guide – Routing Protocols and Concepts, Cisco Press 2008
- JOHNSON, Allan: CCNA Exploration Labs and Study Guide – Routing Protocols and Concepts, Cisco Press 2008
- SCOTT, Empson: CCNA Portable Command Guide, Cisco Press 2007 (v roce 2009 vyšel český překlad „CCNA Kompletní přehled příkazů“ v nakladatelství Computer Press)
- Kolektiv: jednotlivá RFC ke zmiňovaným protokolům: <http://www.ietf.org/rfc.html> .
- CIOARA, Jeremy a kol.: CCNA Exam Prep (Second Edition), Pearson Education 2008
- VALENTINE, Michael a kol.: CCNA Exam Cram (Third Edition), Que Publishing 2008
- CIOARA, Jeremy: CCNA Practice Questions (Exam 640-802) (Third Edition), Que Publishing 2008
- ODOM, Wendell: CCNA Video Mentor (Second Edition), Cisco Press 2008
- McQUERRY, Steve: Authorized Self-Study Guide Preparation Library (Seventh Edition), Cisco Press 2008
- LAMMLE, Todd: CCNA: Cisco Certified Network Associate, Study Guide (Sixth Edition), Wiley Publishing 2007 (v roce 2010 vyšel český překlad „CCNA Výukový průvodce přípravou na zkoušku 640-802“ v nakladatelství Computer Press)

Doporučená motivační četba – bezpečnost datových sítí

- MITNICK, Kevin: The Art of Deception: Controlling the Human Element of Security, John Wiley & Sons, Inc. 2002 (v roce 2003 vyšel český překlad „Umění klamu“ v nakladatelství Helion S.A.)

Studijní materiál
CCNA Exploration – Směrování, koncepce a protokoly
(Semestr 2)

Kolektiv autorů (řešitelé projektu):

Koncepce a text: Ing. Miroslav Páv

Vektorová grafika: Mgr. Jan Syřínek

Konzultace angličtiny: Mgr. Jana Hošková

Vydala: VOŠ a SPŠE Plzeň, Kotterovská 85, 326 00 Plzeň v roce 2011

Tisk: Typos, tiskařské závody, s.r.o., Podnikatelská 1160/14, 320 56 Plzeň

Vydání: 1. (elektronická verze: 3.06, export do formátu PDF: 24.11.2011)

189 stran

NEPRODEJNÉ

Tato publikace je spolufinancována Evropským sociálním fondem a státním rozpočtem České republiky v rámci projektu „Výuka počítačových sítí v mezinárodním programu Síťová akademie Cisco na střední průmyslové škole elektrotechnické“.

Registrační číslo projektu: CZ.1.07/1.1.12/01.0004.