DHCP a NAT

Dynamické přidělování IPv4 adres

Jak nastavit DHCPv4 na routeru

Je třeba stanovit tzv. pool, což je rozsah adres, které budou zájemcům přidělovány. Rozsah je určen adresou sítě s maskou, ale některé adresy z tohoto rozsahu budeme chtít vyjmout z přidělování (třeba proto, že patří samotnému routeru/bráně a případně některým serverům či různým síťovým zařízením). Například pokud chceme přidělovat rozsah 10.10.0.0/16, ale z něj chceme vynechat prvních 8 adres a pak ještě jednu:

```
ip dhcp excluded-address 10.10.0.1 10.10.0.8        ; rozsah
ip dhcp excluded-address 10.10.0.254                ; jedna adresa
ip dhcp pool LAN-1                                  ; nazev rozsahu pro přidělování
      network 10.10.0.0 255.255.0.0
      default-router 10.10.0.1
      dns-server xxxxxxx              ; tady by byla adresa DNS serveru, nepovinne
      domain-name xxxxxxx             ; nazev domeny, taky nepovinne
      lease 2                         ; přiřazení bude trvat dva dny
```

Jak vidíme, definice DHCP poolu nás převede do subkonfiguračního módu, ve kterém určíme parametry: přidělovaný rozsah adres, adresu brány, a pak další (už nepovinné) parametry jako adresa DNS serveru, název domény, lease time (doba přidělení adresy z poolu) atd.

Samozřejmě existují `show` příkazy:

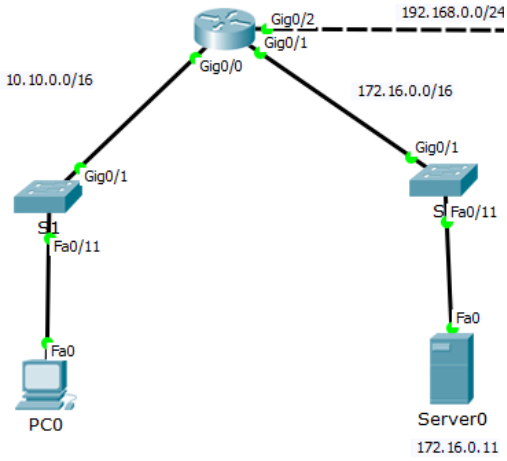
```
show run | section dhcp
show ip dhcp pool
show ip dhcp binding            ; stav přidělení adres
show ip dhcp server statistics
show ip dhcp conflict
```

atd., použijte otazník.

Na jednom routeru může být definováno více DHCP poolů, protože každé rozhraní routeru „vidí“ do jiné sítě a více takových sítí může používat soukromé adresy. Pro každou síť potřebujeme jiný rozsah.

Příklad:

Použijte předpřipravený soubor. Na routeru R1 nakonfigurujte pool pro rozsah adres 10.10.0.0/16 podle výše uvedeného návodu (stačí uvedený rozsah adres a rozsah vyjmutých adres).



Pak se podívejte, jestli už počítač PC0 už má adresu a bránu, a kterou adresu vlastně získal. Vyzkoušejte, jestli půjde z PC0 pingnout server, případně použijte webový prohlížeč na PC0.

Proč myslíte, že není nutné definovaný pool nasazovat na rozhraní? Pokud by bylo definováno více poolů, podle čeho pozná router, ze kterého má žadateli adresu přidělit?

Dále na routeru zkontrolujte stav přidělení adres (tj. dynamickou tabulku přidělených adres), případně vyzkoušejte další `show` příkazy.

Helper

Pokud máme ve firmě více routerů s různými soukromými sítěmi a pouze jeden z nich bude plnit roli DHCP serveru, pak ostatní routery musejí plnit roli tzv. relay agenta. To je přeposílač, který přepošle broadcast

daného typu do jiné sítě (to by jinak nešlo, broadcasty končí na hranici sítě). Pokud chceme, aby router plnil roli relay agenta, spustíme na něm tzv. helper. Helpery mohou být pro různé protokoly/služby, záleží, co vše na zařízení s určitou adresou běží.

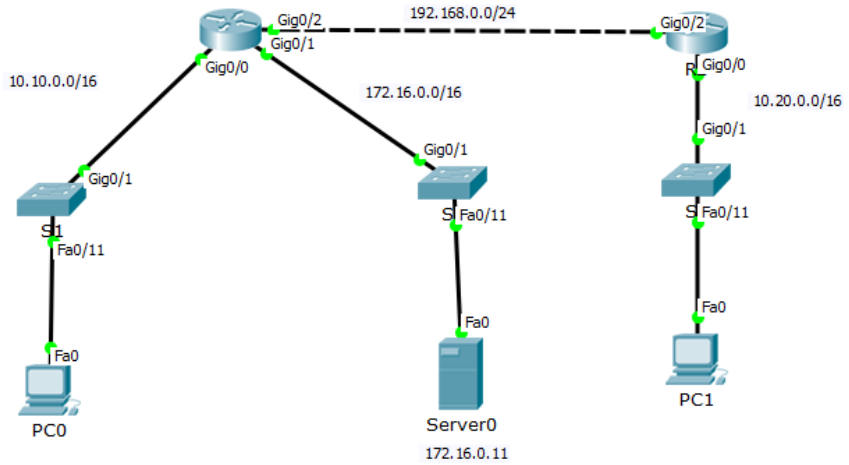
Postup: na budoucím relay agentovi, konkrétně rozhraní, za kterým se nacházejí žadatelé o adresu, nastavíme helper odkazující na adresu serveru poskytujícího danou službu (dáme tam tu adresu, pod kterou náš „agent“ vidí poskytovatele služby).

```
int g0/xxx
      ip helper-address 192.168.0.1
```

Nastavení helperu se dá ověřit třeba v running-configu nebo ve výpisu parametrů rozhraní:

```
sh ip int g0/xxxx
```

Příklad:
Na routeru R1x vytvořte druhý pool pro adresy 10.20.0.0/16, opět prvních 8 adres vyjměte. Na routeru R2 pak spusťte helper umožňující zařízením za rozhraním g0/0 získat od routeru R1 adresu.



Vyzkoušejte, jestli jde z PC1 pingnout jak server, tak i PC0.

Bezstavový SLAAC pro IPv6

Ve světě IPv6 máme poněkud více možností, jak získat IP adresu. Nejjednodušší pro koncová zařízení je SLAAC (Stateless Autoconfiguration), která je na Cisco routerech defaultní – jen je třeba zprovoznit IPv6:

```
ipv6 unicast-routing
```

A samozřejmě přidělit IPv6 adresu na daném rozhraní routeru.

Samotný router posílá RA zprávu (Router Advertisement), ze které se klienti dozví adresu sítě, adresu brány a případně další parametry. Hostitelskou část adresy si pak dopočtou sami.

Další možnosti jsou stavový a bezstavový DHCPv6. Stavový funguje stejně jako v IPv4 (přiděluje i adresy), ktežto bezstavový je „na půl cesty“: adresu (i bránu) si klient vyřídí pomocí routeru, ale ostatní informace dostane od DHCP serveru (například adresu DNS serveru, název domény apod.). Klient je o konkrétním nastavení sítě informován v RA zprávě od routeru, kde jsou dva důležité příznaky: M (managed flag) a O (other config flag).

	SLAAC	stateful DHCPv6	stateless DHCPv6
Nastavení flagů:	M = 0, O = 0	M = 1	M = 0, O = 1

Příklad:
V rozpracovaném souboru zprovozněte na routeru R1 IPv6, přidělte na rozhraní g0/0 tyto adresy:

- 2001:db8:acad:10::1/64 jako globální unicast
- fe80::1 jako link-local adresu

Na rozhraní g0/1:

- 2001:db8:acad:172::1/64 jako globální unicast
- fe80::1 jako link-local adresu

Ověřte si, jakou adresu si PC0 dopočítalo.

Nastavení stateless DHCPv6

Na routeru je třeba kromě toho, co bylo uvedeno u SLAAC, provést dvě další nastavení:

- vytvořit pool, ve kterém však nebude přidělitelný rozsah adres, ale jen ty další parametry, o které budou klienti žádat (DNS server apod.),
- na daném rozhraní nasadit pool a nastavit other config flag.

```
ipv6 dhcp pool BEZSTAVOVY1
       dns-server xxxxxx
       domain-name xxxxx
       exit
int g0/1
       ipv6 addr xxxxx
       ipv6 dhcp server BEZSTAVOVY1
       ipv6 nd other-config-flag
```

Nastavení stateful DHCPv6

Postupujeme podobně jako v předchozím případě, jen při vytváření poolu definujeme také rozsah IP adres a na rozhraní nastavíme managed config flag:

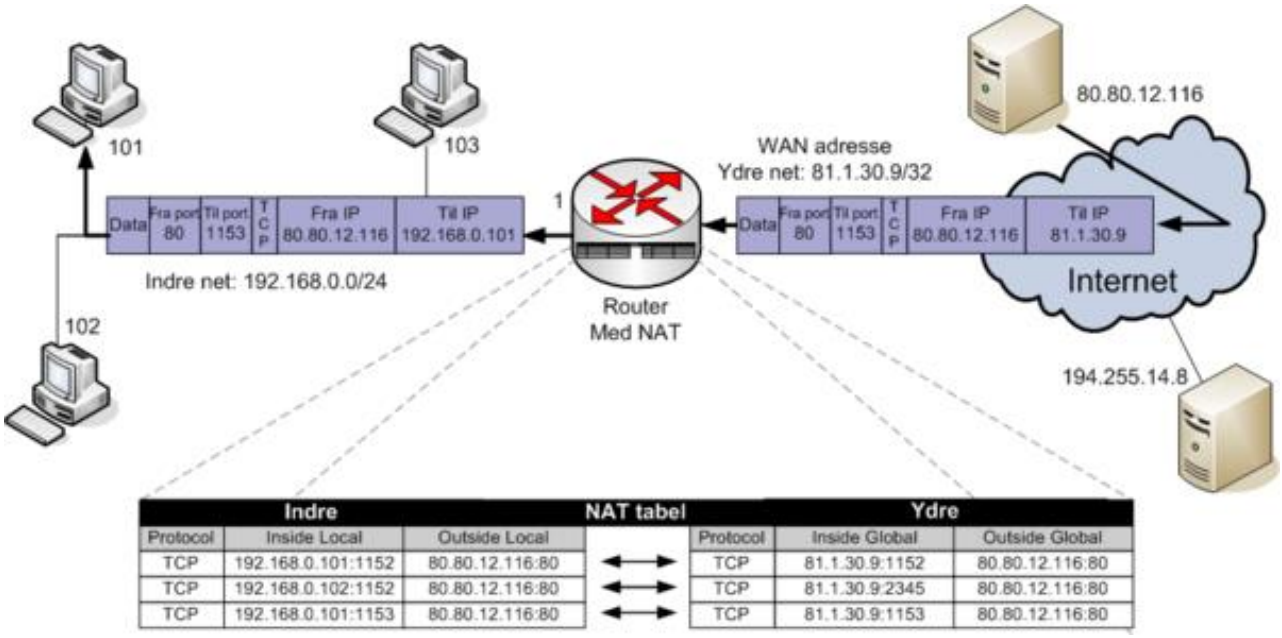
```
ipv6 dhcp pool BEZSTAVOVY2
       address prefix 2001:db8:........::/64 lifetime infinite
       dns-server xxxxxx
       domain-name xxxxx
       exit
int g0/1
       ipv6 addr xxxxx
       ipv6 dhcp server BEZSTAVOVY1
       ipv6 nd managed-config-flag
```

Všimněte si, jakým způsobem se stanovuje rozsah adres – trochu jiné je klíčové slovo, a také doba pronájmu adresy je součástí určení rozsahu.

Překlad adres

NAT (Network Address Translation) může být buď statický nebo dynamický. Zatímco u statického „ručně“ určujeme, která adresa se na kterou adresu přeloží, u dynamického se určí, který rozsah se na který rozsah přeloží, a router si už samotné přiřazování řídí sám. Momentální stav přeložení si ukládá do tabulky pro dynamický překlad adres, proto hovoříme o stavovém NAT.

Dynamický NAT se prakticky výhradně provádí jako PAT (Port Address Translation): skupina vnitřních adres se překládá na jednu „vnější“, ale rozlišují se podle čísla portu. A v případě, že dojde ke shodě v čísle portu s jiným lokálním zařízením, přeloží se kromě adresy i číslo portu. Následující obrázek ukazuje princip:



Zdroj: https://mars.merhot.dk/w/index.php/NAT_Cisco_IOS

Oblast vnitřní sítě je „inside“, oblast za routerem je „outside“. Adresy označené jako „local“ překládáme na adresy označené jako „global“, třebaže ve skutečnosti ani ty nemusejí být globálně platné, je to jen názvová konvence.

Příklad:
Vytvoříme na routeru R1 dynamický NAT pro síť 10.10.0.0/16. Nejdřív pomocí ACL určíme, které adresy vlastně mají být překládány:

```
access-list 1 permit 10.10.0.0 0.0.255.255
```

Teď se dá pokračovat dvěma různými způsoby. Buď budeme překládat na více adres, a tedy by bylo třeba definovat pool adres, nebo zvolíme jednoduchý PAT s jedinou adresou pro překlad. To je jednodušší a vystačíme si s jednou adresou na odchozím rozhraní, tedy zvolíme druhou možnost. Jako parametr uvedeme access list, který jsme si předem vytvořili, a pak rozhraní, kterého se bude překlad týkat a jehož IP adresu použijeme.

```
ip nat inside source list 1 interface g0/2 overload
```

Klíčové slovo `overload` právě znamená, že se nebude překládat na pool adres, ale na adresu uvedeného rozhraní.

Ještě zbývá provést změny na příslušných rozhraních, tedy sdělit routeru, za kterými rozhraními je „inside“ a „outside“.

```
interface g0/0
      ip nat inside
interface g0/2
      ip nat outside
```

Také pro NAT existují `show` příkazy:

```
show ip nat translations
show ip nat statistics
clear ip nat statistics
debug ip nat
```