GRE tunely

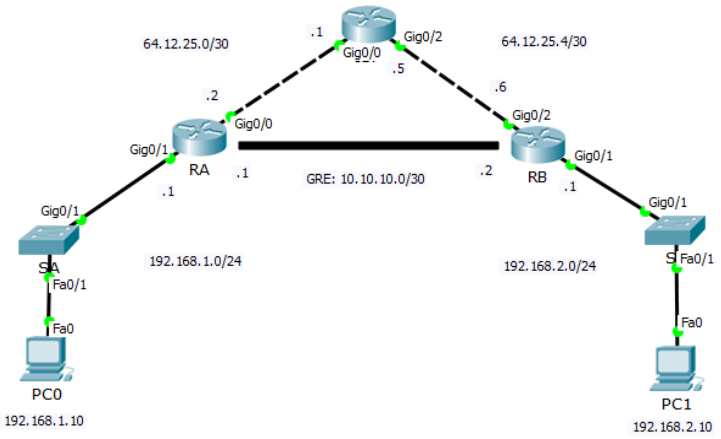
Jak vytvořit GRE tunel na Cisco routeru

GRE tunel se vytvoří jako nové virtuální rozhraní tunel0 (nebo jiné číslo, když je tunelů víc). Tunel povede mezi dvěma routery, obvykle hraničními routery našich sítí. Na vytvořeném rozhraní stanovíme

- vlastní IP adresu nového rozhraní,
- informaci o napojení na skutečné rozhraní (označení rozhraní, které je vstupem do tunelu),
- na kterou IP adresu jako cílovou má být PDU poslán (IP adresa druhého konce tunelu),
- režim provozu tunelu, v našem případě to bude GRE s IP provozem.

```
int tunnel 0
  ip address xxxxx xxxxxx
  tunnel source xxxx            // rozhraní, na které posílám
  tunnel destination xxxx       // IP adresa v reálné síti na druhém konci tunelu
  tunnel mode gre ip
```

Příklad:



Máme přednastavenou síť v souboru 08_GRE_OSPF.pkt. Na routerech i na počítačích už jsou přednastaveny IP adresy.

Na obrázku vidíme topologii: dvě pobočkové sítě (na hranicích jsou routery RA a RB), nahoře je router ISP simulující poskytovatele internetu.

Naším úkolem je nakonfigurovat GRE tunel mezi routery RA a RB tak, aby nebylo nutné dělat překlad adres na hranicích pobočkových sítí. Provoz bude reálně procházet přes router ISP, ale protože bude tunelován a vnější adresa bude patřit do vnějších sítí, NAT nebude zapotřebí.

Na routeru RA vytvoříme rozhraní `tunnel0` a nastavíme parametry následovně:

```
RA(config)#int tunnel0

RA(config-if)#
%LINK-5-CHANGED: Interface Tunnel0, changed state to up

RA(config-if)#ip addr 10.10.10.1 255.255.255.252
RA(config-if)#tunnel source g0/0
RA(config-if)#tunnel dest 64.12.25.6
RA(config-if)#tunnel mode gre ip
```

Prvním příkazem jsme tunel vytvořili, od té chvíle existuje také ve výstupu příkazu `sh ip int br` a dalších.

Podobně na druhé straně tunelu, na routeru RB:

```
RB(config)#int tunnel0

RB(config-if)#
%LINK-5-CHANGED: Interface Tunnel0, changed state to up

RB(config-if)#ip addr 10.10.10.2 255.255.255.252
RB(config-if)#tunnel source g0/2
RB(config-if)#tunnel dest 64.12.25.2
RB(config-if)#tunnel mode gre ip
```

Ale to není všechno. Máme dvě sítě, zbývá mezi nimi nakonfigurovat směrování. Použijeme směrovací protokol OSPF, ovšem pouze mezi vlastními sítěmi. Pro směrování uvnitř tunelu (reálně tedy to vnějších linkách) použijeme statické směrování. Nejdřív zprovozníme statické směrování, aby byl tunel použitelný (všimněte si, že zatím se tunel nevyskytuje ve směrovací tabulce, a v seznamu rozhraní sice je, ale jako neaktivní). Na routeru RA:

```
RA(config)#ip route 0.0.0.0 0.0.0.0 g0/0
```

Podobně na routeru RB:

```
RB(config)#ip route 0.0.0.0 0.0.0.0 g0/2
```

Zobrazte si směrovací tabulku a seznam rozhraní, a také podrobnosti o novém rozhraní:

```
sh ip route
sh ip int br
sh int tunnel0
```

Mezi routery RA a RB by už měl tunel fungovat se vším všudy, včetně zapouzdřování. Pokud pingneme z RB na RA, dostaneme tento výsledek:

```
RB#ping 10.10.10.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.1, timeout is 2 seconds:
..!!!
Success rate is 60 percent (3/5), round-trip min/avg/max = 0/0/0 ms
```

Ještě zkusíme traceroute, ať vidíme cestu:

```
RB#traceroute 10.10.10.1
Type escape sequence to abort.
Tracing the route to 10.10.10.1

1 10.10.10.1 0 msec 0 msec 0 msec
RB#
```

To znamená, že (zdánlivě) naše pakety nešly přes žádný jiný router, jen přes tunel. Reálně samozřejmě jdou pakety přes router ISP, ale s přidanými záhlavími GRE a vnější IP s jinou adresou, proto se nám v seznamu neobjevila žádná z adres nastavených na routeru ISP, protože zde se počítají pouze výsledky podle TTL z vnitřního IP záhlaví.

Zbývá zprovoznit dynamické směrování, aby mohla komunikovat i zbývající zařízení z vnitřních sítí. Dynamické směrování na routeru RA:

```
RA(config)#router ospf 1
RA(config-router)#router-id 1.1.1.1
RA(config-router)#network 192.168.1.0 0.0.0.255 area 0
RA(config-router)#network 10.10.10.0 0.0.0.3 area 0
```

Na routeru RB:

```
RB(config)#router ospf 1
RB(config-router)#router-id 2.2.2.2
RB(config-router)#network 192.168.2.0 0.0.0.255 area 0
RB(config-router)#network 10.10.10.0 0.0.0.3 area 0
```

Na routeru RB si zobrazíme i směrovací tabulku:

```
Gateway of last resort is 0.0.0.0 to network 0.0.0.0

10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 10.10.10.0/30 is directly connected, Tunnel0
L 10.10.10.2/32 is directly connected, Tunnel0
64.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
```

```
C 64.12.25.4/30 is directly connected, GigabitEthernet0/2
L 64.12.25.6/32 is directly connected, GigabitEthernet0/2
O 192.168.1.0/24 [110/1001] via 10.10.10.1, 00:00:39, Tunnel0
192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C 192.168.2.0/24 is directly connected, GigabitEthernet0/1
L 192.168.2.1/32 is directly connected, GigabitEthernet0/1
S* 0.0.0.0/0 is directly connected, GigabitEthernet0/2
```

Kromě přímo připojených sítí a jednoho statického záznamu kvůli tunelu tam máme jeden záznam začínající písmenem „O“, tedy pocházející od protokolu OSPF, jde právě o síť v druhé pobočce (za routerem RA).

Ping mezi počítači by už měl fungovat, a když si vytrasujeme cestu, zjistíme, že opravdu jde přes tunel:

```
PC>tracert 192.168.2.10

Tracing route to 192.168.2.10 over a maximum of 30 hops:

1 0 ms 0 ms 0 ms 192.168.1.1
2 1 ms 0 ms 0 ms 10.10.10.2
3 0 ms 0 ms 0 ms 192.168.2.10

Trace complete.
```

Zdroje ke zprovoznění šifrování (IPSec) v GRE tunelu

Zprovoznit IPSec v tunelu už není tak jednoduché, ale dají se najít zdroje na internetu. Například následující zdroj vysvětluje problém velmi podrobně a do důsledků:

https://gulian.uk/how-to-configure-gre-over-ipsec-in-cisco-ios-and-cisco-ios-xe-devices/

Stručnější, ale zato i včetně řešení MTU a MSS:

https://www.petenetlive.com/KB/Article/0000951