Projekt

Předmět: SPS

Howto VRF/VPN na CISCO routerech v. 2

Zpracoval: BUČKOVÁ Dagmar, BUC061
HRABÁLEK David, HRA026

Datum odevzdání: 28. 6. 2007

1. Obsah

2. Úvod

V současné době se ve výjimečných případech řeší připojení některých externích uživatelů k síti řeší prostřednictvím konfigurace policy routingu a tunelů. Např. takto můžeme řešit přístup externích pracovníků větší firmy, kteří budou mít přístup pouze do určité části podnikové sítě. Tunely jsou také vhodné k zajištění větší bezpečnosti v podnikových sítích. Uživatelé (zaměstnanci) daného úseku společnosti vidí pouze část sítě přidělené jejich úseku. Tedy nevidí síť jako celek.

Tato konfigurace je poměrně složitá a u méně triviálních případů i nepřehledná. I když u starších routerů řady 2500 zatím jiná možnost konfigurace není, v novějších verzích IOSu na silnějších routerech existují vlastnosti, které umožňují dosáhnout stejné i lepší funkcionality a větší bezpečnosti při zachování přehlednosti konfigurace. Tyto vlastnosti byly implementovány s příchodem MPLS (Multiprotocol Label Switching), ale mohou se použít i bez vybudované MPLS sítě. Tyto vlastnosti se souhrnně nazývají VRF/VPN routing/forwarding.

VPN - Virtual Private Network, virtuální privátní síť. VRF - Virtual Routing and Forwarding, virtuální směrování a přeposílání. Při tvorbě tunelu vytváříme privátní síť, která geograficky může mít část sítě v Praze a další část v Brně, přitom se stále jedna o jednu „virtuální“ síť, ačkoli části této sítě mohou různě geograficky rozesety a adresace může být různá. VRF řeší tyto tunely na routerech firmy Cisco.

3. Popis VRF

Pomocí VRF vlastností je možné na Cisco routeru vytvořit několik VRF instancí, které obsahují jednak svoji vlastní směrovací tabulku a přiřazená rozhraní. Díky odděleným samostatným tabulkám není možná komunikace ani mezi přímo připojenými rozhraními routeru, pokud spadají do různých VRF. Dokonce je možné, aby na jednom routeru byly v různých VRF na různých rozhraních stejné IP adresy. Router je dokáže rozlišit, protože k adresám přidává prefix VRF, tedy jakýsi unikátní identifikátor VRF, který je jedinečný pro každé VRF.

V podstatě se jedná o vytvoření přímé cesty (tunelu) z jedné sítě do druhé, tak aby dané sítě nevěděly o zbytku sítě, v praxi to vypadá tak, že i když data projdou přes *n* routeru či *m* cizích sítí, jeví se jako by neprošla žádným routerem či cizích sítí, a jeví se jako by se pohybovala stále ve své vlastní sítí, která spadá pod daný VRF. Taktéž ani routery neví co přes ně jde a kam to směřuje.

4. Dostupnost

Vlastnosti potřebné pro realizaci takových to VPN jsou dostupné v IOSech od verze 12.0(5)T. Nejsou však podporovány všechny platformy. Seznam podporovaných modelů se sice postupně rozšiřuje, ale je možné předpokládat, že modely slabší než C2600 tyto vlastnosti podporovat nebudou. Při konfiguraci a testování vlastností VRF/VPN při modelové situaci nám byly zapůjčeny routery 2600, 3600, 4500, firmou ČD Telematika, které mají vlastnosti pro podporu VRF. Software musí být použit IOS 12.0(5)T, či novější, přitom nezáleží na feature-setu, VRF je dostupné i v IP only versi.

5. Možné způsoby připojení externího uživatele do sítě

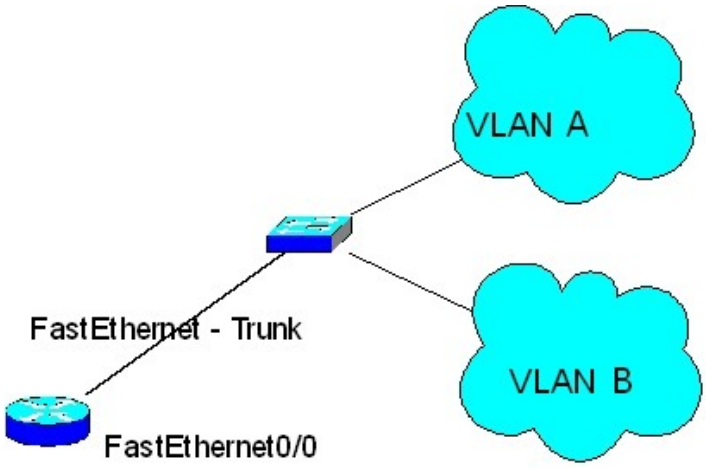
5.1 Připojení uživatele k routeru

Při připojování externího uživatele do sítě prostřednictvím routeru musí být vždy splněno jedno základní pravidlo. Uživatel musí být připojen buď na samostatné - dedikované -

rozhraní routeru, nebo prostřednictvím sítě VLAN na samostatný FastEthernet subinterface routeru. Pouze samostatný interface nebo subinterafce dané VLAN může být přiřazen do vlastní VRF.

5.2 Připojení pomocí VLAN

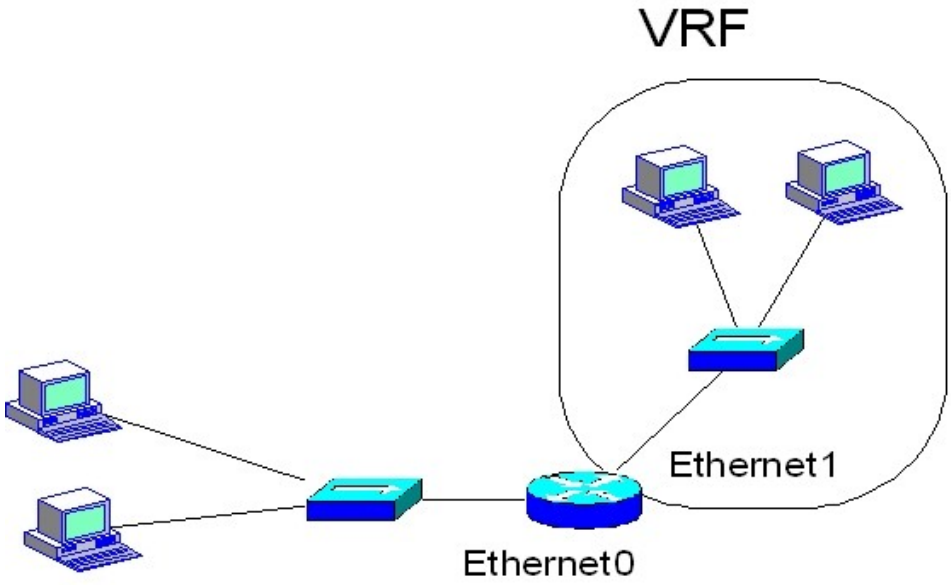
Na routeru se vytvoří subinterface pro jednotlivé sítě VLAN. Příkazem *ip vrf forwarding name*, se pak přiřadí subinterface s externím klientem ke směrovací tabulce. Tím se mu umožní komunikace jen uvnitř této VRF.



Obr. 1

5.3 Připojení na samostatné ethernet rozhraní

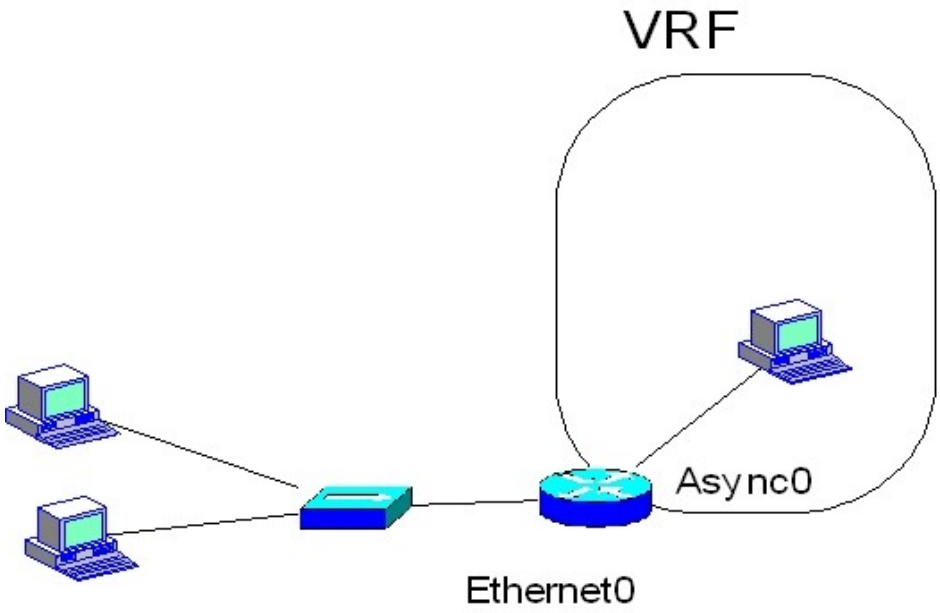
Celé rozhraní, na kterém je připojený klient se příkazem *ip vrf forwarding name* přiřadí ke směrovací tabulce. Klient může být k rozhraní připojen buď prostřednictvím vlastních LAN prvků, nebo opět pomocí technologie VLAN.



Obr. 2

5.4 Připojení na samostatný Asynchronní port

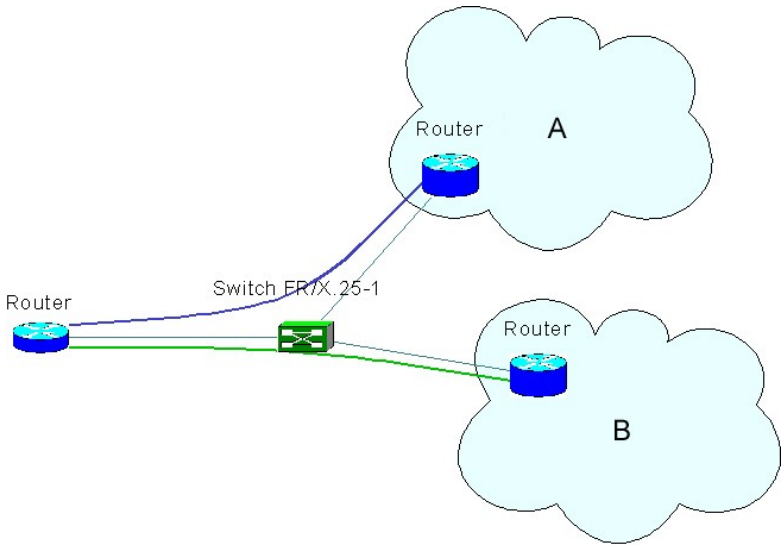
Obdobně jako v případě samostatného ethernet rozhraní je možné i asynchronní port přiřadit ke směrovací tabulce.



Obr. 3

5.5 Zajištění spojení do sítě

Aby mohl externí klient, připojený k routeru, komunikovat s Internetem, servery či ostatními pobočkami, musí být jeho pakety směrovány dále. Nejprve je potřeba získat nějaké odchozí rozhraní přiřazené ke směrovací tabulce a později nastavit statické směrování v této směrovací tabulce. Tím, že jsou rozhraní na která je klient připojen umístěna v samostatné směrovací tabulce, nemohou být pakety směrovány pomocí směrovacích informací mimo tuto tabulku.

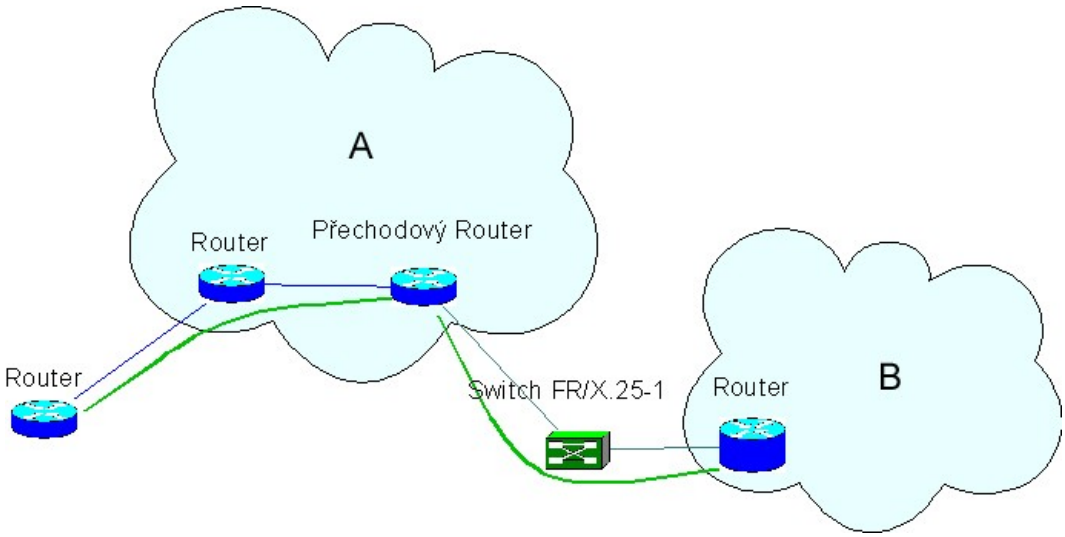


Obr. 4

6. Tunel

Pokud router, na který bude připojený externí uživatel není dále připojen pomocí Frame-Relay, je možné vytvořit odchozí interface pomocí tunelu.

Tunel je možné přiřadit k dané VRF tabulce, ale přitom jeho zdrojová a cílová adresa nemusí být v daném VRF známá. V našem případě se tunel použije k přemostění cesty po IP síti od routeru, kde je externí uživatel připojen k tzv. přechodovému routeru, který bude sloužit ke směrování paketů mezi tunel rozhraním a rozhraním vedoucím do routeru. Tento princip je podobný jako u doposud používaného řešení pomocí VPN. Rozdíl je v tom, že ani na routeru u zákazníka ani na přechodovém routeru se nepoužívá policy routing, ale oddělení je zajištěno pomocí vlastních směrovacích tabulek.



Obr. 5

7. Popis konfigurace

Důležitou vlastností konfigurace VPN pomocí VRF je ta, že pro vytvoření nové VRF na routeru není potřeba měnit stávající konfiguraci. Stačí jen doplnit konfigurační příkazy pro VRF, ale ostatní konfigurace (OSPF, statické směrování, …) zůstane zachována beze změny.

Pro používání více směrovacích tabulek na routeru musí být nejprve povolena vlastnost Cisco Express Forwarding – CEF. Tato vlastnost se povoluje globálním konfiguračním příkazem ip cef.

Vlastní směrovací tabulky se vytvářejí příkazem *ip vrf name*, čímž se dostaneme do konfigurační sekce příkazu *ip vrf*. V této sekci je nutné určit tzv. route distinguisher (RD). Z tohoto parametru se vytvoří 8 bytů veliké číslo, které se používá jako prefix, příkazem *rd prefix:cislo*, před IP adresami dané VPN. Díky tomu je možné identifikovat, do které VPN ta která adresa patří (obvykle se volí číslo autonomního systému a nějaké další číslo).

Jednotlivá rozhraní se přiřazují ke směrovacím tabulkám pomocí příkazu *ip vrf forwarding name*. Přiřazení k VRF se musí provést ještě před konfigurací IP adresy na daném rozhraní.

Při konfiguraci statického směrování se pouze uvede do jakého VRF se má zadaný záznam přiřadit, jinak se konfigurace provádí stejně, tedy pomocí běžného příkazu
ip route vrf name cíl maska brána.

8. Příklad konfigurace

Cílem bylo ověřit funkčnost a použitelnost VRF/VPN pro zvýšení bezpečnosti v podnikové síti. Případné využití VRF/VPN bylo otestováno pomocí zapojení, které je znázorněno na

obrázku níže, Obr. 6. Routery C3600 s úpravou software na C3640 a C3620 byly vybaveny IOSy, 12.2(28a), které použití VRF/VPN umožňují. Router C2600 vytvářel pouze obecnou IP síť a switch sloužil k zakončení ethernetových linek.

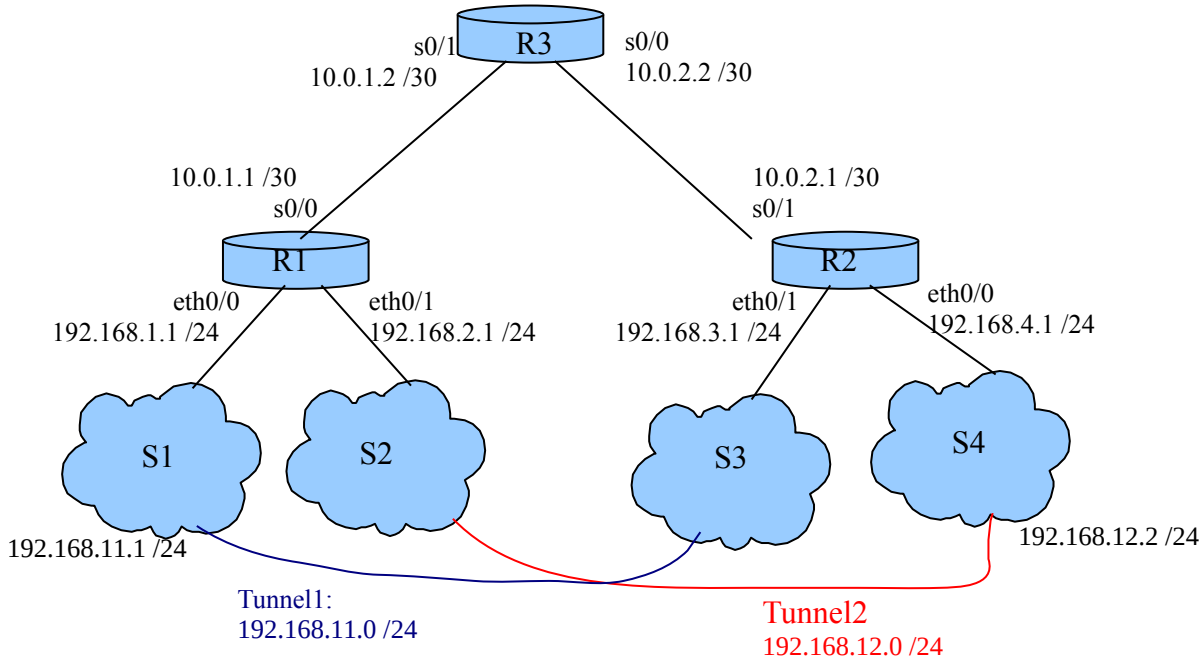
Hardware a software použitý ve VRF zařízeních:

Router1 - C3640:
IOS (tm) 3600 Software (C3620-D-M), Version 12.2(28a), RELEASE SOFTWARE (fc2)
ROM: System Bootstrap, Version 11.1(17)AA, EARLY DEPLOYMENT RELEASE SOFTWARE (fc2)
ROM: 3600 Software (C3620-D-M), Version 12.2(28a), RELEASE SOFTWARE (fc2)
System image file is "flash:c3620-d-mz.122-28a.bin"
cisco 3620 (R4700) processor (revision 0x81) with 28672K/4096K bytes of memory.
2 Ethernet/IEEE 802.3 interface(s)
2 Serial network interface(s)

Router2 - C3600:
IOS (tm) 3600 Software (C3640-I-M), Version 12.2(27), RELEASE SOFTWARE (fc3)
ROM: System Bootstrap, Version 11.1(17)AA, EARLY DEPLOYMENT RELEASE SOFTWARE (fc2)
System image file is "flash:c3640-i-mz.122-27.bin"
cisco 3640 (R4700) processor (revision 0x00) with 27648K/5120K bytes of memory.
4 Ethernet/IEEE 802.3 interface(s)
1 FastEthernet/IEEE 802.3 interface(s)
4 Serial network interface(s)

Router3 - C2600:
IOS (tm) C2600 Software (C2600-I-M), Version 12.3(17a), RELEASE SOFTWARE (fc2)
ROM: System Bootstrap, Version 11.3(2)XA4, RELEASE SOFTWARE (fc1)
ROM: C2600 Software (C2600-I-M), Version 12.3(17a), RELEASE SOFTWARE (fc2)
System image file is "flash:c2600-i-mz.123-17a.bin"
cisco 2610 (MPC860) processor (revision 0x300) with 28672K/4096K bytes of memory.
1 Ethernet/IEEE 802.3 interface(s)
2 Serial network interface(s)

8.1 Schéma zapojení



Obr. 6

Tunnel 1 - sítě: S1 a S3, IP: 192.168.11.0 /24
R1: 192.168.11.1 /24
R2: 192.168.11.2 /24
Tunnel 1 - sítě: S2 a S4, IP: 192.168.12.0 /24
R1: 192.168.12.1 /24
R2: 192.168.12.2 /24

8.2 Postup při zapojení

Nejprve si zapojíme síť, fyzicky. Poté si nastavíme jednotlivá rozhranní routerů a stanic. Nesmíme zapomenout na to, že přiřazení IP adres pro jednotlivá ethernetová rozhraní musíme zadat až poté, co se rozhraní přiřadí do příslušného VRF.

VRF se vytváří pomocí příkazu *ip vrf name*, poté se mu přiřadí unikátní id, *rd 100:12*. Aby příslušné pakety věděly kam mají procházet musíme zajistit distribuci mapování ve VRF, to zajistíme příkazem *import map id číslo dané VRF*, *export map id číslo dané VRF*, v režimu nastavení VRF.

Dále musím vytvořit interface tunelu, interface tunnel id, přiřadíme ho do příslušné VRF, *ip vrf forwarding name*. Poté nastavíme IP tunelu*, ip address ip_add netmask*, v interface tunelu. Nastavíme zdrojové a cílové adresy tunelů, *tunnel source ip_add*, *tunnel destination ip_add*. U tunelu musíme také nastavit statické cesty, *ip route vrf name cíl maska brána*.

8.3 Výpisy z jednotlivých routeru

8.3.1 R1

Router1#sh running-config

```
hostname Router1
!
ip cef
!
ip vrf S1S3
 rd 100:11
 import map 100:11
 export map 100:11
!
ip vrf S2S4
 rd 100:12
 import map 100:12
 export map 100:12
!
interface Tunnel1
 ip vrf forwarding S1S3
 ip address 192.168.11.1 255.255.255.0
 tunnel source 10.0.1.1
 tunnel destination 10.0.2.1
 tunnel mode ipip
!
interface Tunnel2
 ip vrf forwarding S2S4
 ip address 192.168.12.1 255.255.255.0
 tunnel source 10.0.1.1
 tunnel destination 10.0.2.1
 tunnel mode ipip
!
interface Ethernet0/0
```

```
 ip vrf forwarding S1S3
 ip address 192.168.1.1 255.255.255.0
 half-duplex
!
interface Serial0/0
 ip address 10.0.1.1 255.255.255.252
 encapsulation ppp
!
interface Ethernet0/1
 ip vrf forwarding S2S4
 ip address 192.168.2.1 255.255.255.0
 half-duplex
!
router rip
 network 10.0.0.0
!
ip classless
ip route vrf S1S3 192.168.3.0 255.255.255.0 Tunnel1
ip route vrf S2S4 192.168.4.0 255.255.255.0 Tunnel2
ip http server
!
end
```

Router1#sh ip route

```
    10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
R       10.0.2.0/30 [120/1] via 10.0.1.2, 00:00:05, Serial0/0
R       10.0.2.1/32 [120/1] via 10.0.1.2, 00:00:05, Serial0/0
C       10.0.1.2/32 is directly connected, Serial0/0
C       10.0.1.0/30 is directly connected, Serial0/0
```

Router1#sh ip vrf brief

```
  Name                 Default RD          Interfaces
  S1S3                 100:11              Ethernet0/0
                                           Tunnel1
  S2S4                 100:12              Ethernet0/1
                                           Tunnel2
```

Router1#sh ip vrf detail

```
VRF S1S3; default RD 100:11
  Interfaces:
    Ethernet0/0           Tunnel1
  Connected addresses are not in global routing table
  No Export VPN route-target communities
  No Import VPN route-target communities
  Import route-map: 100:11
  Export route-map: 100:11
VRF S2S4; default RD 100:12
  Interfaces:
    Ethernet0/1           Tunnel2
  Connected addresses are not in global routing table
  No Export VPN route-target communities
  No Import VPN route-target communities
  Import route-map: 100:12
  Export route-map: 100:12
```

8.3.2 R2

Router2#sh running-config

```
hostname Router2
!
ip cef
!
ip vrf S1S3
 rd 100:11
 import map 100:11
 export map 100:11
!
ip vrf S2S4
 rd 100:12
 import map 100:12
 export map 100:12
!
interface Tunnel1
 ip vrf forwarding S1S3
 ip address 192.168.11.2 255.255.255.0
 tunnel source 10.0.2.1
 tunnel destination 10.0.1.1
 tunnel mode ipip
!
interface Tunnel2
 ip vrf forwarding S2S4
 ip address 192.168.12.2 255.255.255.0
 tunnel source 10.0.2.1
 tunnel destination 10.0.1.1
 tunnel mode ipip
!
interface Ethernet0/0
 ip vrf forwarding S2S4
 ip address 192.168.4.1 255.255.255.0
 half-duplex
!
interface Serial0/0
 ip address 10.0.2.1 255.255.255.252
 encapsulation ppp
!
interface Ethernet0/1
 ip vrf forwarding S1S3
 ip address 192.168.3.1 255.255.255.0
 half-duplex
!

router rip
 network 10.0.0.0
!
ip classless
ip route vrf S1S3 192.168.1.0 255.255.255.0 Tunnel1
ip route vrf S2S4 192.168.2.0 255.255.255.0 Tunnel2
ip http server
!
end
```

Router2#sh ip route

```
     10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C       10.0.2.0/30 is directly connected, Serial0/0
R       10.0.1.1/32 [120/1] via 10.0.2.2, 00:00:02, Serial0/0
C       10.0.2.2/32 is directly connected, Serial0/0
R       10.0.1.0/30 [120/1] via 10.0.2.2, 00:00:02, Serial0/0
```

Router2#sh ip vrf brief

```
  Name                 Default RD          Interfaces
  S1S3                 100:11              Ethernet0/1
                                           Tunnel1
  S2S4                 100:12              Ethernet0/0
                                           Tunnel2
```

Router2#sh ip vrf details

```
  VRF S1S3; default RD 100:11
  Interfaces:
    Ethernet0/1               Tunnel1
  Connected addresses are not in global routing table
  No Export VPN route-target communities
  No Import VPN route-target communities
  Import route-map: 100:11
  Export route-map: 100:11
VRF S2S4; default RD 100:12
  Interfaces:
    Ethernet0/0               Tunnel2
  Connected addresses are not in global routing table
  No Export VPN route-target communities
  No Import VPN route-target communities
  Import route-map: 100:12
  Export route-map: 100:12
```

8.3.3 R3

Router3#sh running-config

```
hostname Router3
!
ip cef
!
interface Serial0/0
 ip address 10.0.2.2 255.255.255.252
 encapsulation ppp
 clock rate 128000
!
interface Serial0/1
 ip address 10.0.1.2 255.255.255.252
 encapsulation ppp
 clock rate 128000
!
router rip
 network 10.0.0.0
!
ip http server
ip classless
!
line con 0
line aux 0
line vty 0 4
 password qwe
 login
```

9. Závěr

Z Routeru1 se po nastaveni VRF nedalo již pingnout na veškerá ethernetova rozhraní Routeru2 což bylo žádoucí chování. Taktéž to bylo u Routeru2, kde se nedalo pingnout na veškerá ethernetová rozhraní Routeru1. Router3 zde působí pouze jako pasivní prvek a vidí pouze přípojné linky neboť na ethernetových rozhraních není rozjet žádný routovací protokol.

Ze sítě S1 šlo pingnout na eth rozhraní Routeru1 a dále až na rozhraní S3. Do sítě S2 na stejném routeru Router1 taktéž nešlo pingnout což je žádoucí účinek.

Ze stanice ze sítě S1 nelze pingnout ani na Router2 (10.0.2.1) což je požadovaná vlastnost.

Problémem byla distribuce routovacích map, kdy routery neví kam co poslat, řešení je zajištěno pomocí distribuce routovacích map (viz. Router1#sh ip vrf details). Dále může nastat problém s enkapsulací. Bohužel nám není známo proč tento problém vzniká. Bez nastavení enkapsulace nebylo možno se pingnout na danou síť. Problém můžeme vyřešit pomocí nastavení enkapsulace na routeru, konkrétněji na příslušném tunelu (tunnel mode ipip).