Tavasz 2017

UNIVERSITAS SCIENTIARUM SZEGEDIENSIS
UNIVERSITY OF SZEGED
Department of Software Engineering

Számítógép-hálózatok 7. gyakorlat

InterVLAN routing

Bordé Sándor

Szegedi Tudományegyetem

Tartalomjegyzék

Bevezetés 3

VLAN emlékeztető 3

Gondok a VLAN használatával 3

Megoldás 4

Inter VLAN lehetőségek 4

1. Változat: 1 router – n port 4

Hátrányok 5

2. változat: 1 router – 1 port (Router on a stick) 5

Trunk port és DOT1Q header 5

Megvalósítás 6

3. változat: L3 switch 8

Megvalósítás 8

3+1. változat: L3 switch + router 10

Gyakorló feladat 11

Felmerülő kérdések 11

Források 12

.1Q header 12

Trunk port 12

Router on a stick konfiguráció 12

Layer 3 switchek 12

Inter VLAN routing Layer 3 switchekkel 12

Beugró kérdések 13

Melléklet 14

1. Router on a stick CLI parancsok 14

2. L3 Switch CLI parancsok 15

Bevezetés

Két anyaggal korábban már szó esett a VLAN-ok használatáról és megnéztük, hogy lehet egy hálózatot szegmentálni router nélkül, virtuális hálózatok konfigurálásával. Azonban a beállítások után maradt néhány megválaszolatlan kérdés: hogy tudnak az egyes VLAN-hoz rendelt gépek kommunikálni külső hálózattal? Hogy tudnak egymással kommunikálni a különböző VLAN-ba tartozó gépek? Ebben az anyagban ezekre a kérdésekre keressük a választ.

Mielőtt rátérnénk a bevezetőben felvetett kérdésekre, röviden ismételjük át, amit eddig tudunk a VLAN-okról.

VLAN emlékeztető

Ahogy egy szervezet nő és fejlődik, úgy nő a számukra szükséges hálózat mérete és összetettsége. A leggyakrabban használt Ethernet alapú hálózatok rendszeresen használnak broadcast üzeneteket különböző szolgáltatásaikhoz. Azonban minél nagyobb a hálózat, annál valószínűbb, hogy az egyes eszközök olyan szórásos csomagot is megkapnak, ami nem nekik szól, tehát ezzel feleslegesen terhelik a hálózatot. Erre nyújt megoldást az, ha szegmentáljuk a hálózatunkat, azaz külön szórási tartományokat hozunk létre.

Az egyik lehetőség a LAN szegmentációra az, ha külön alhálózatokat hozunk létre routerek segítségével (ahogy láttuk azt a 4. gyakorlati jegyzetben). Viszont a routerek felhasználásának (a sok előnyük mellett) két nagy hátránya van: a routerek komplexebb eszközök, így általában többe is kerülnek. Illetve ezek az eszközök az OSI modell 3. rétegében működnek, így többféle döntést kell meghozniuk egy-egy csomag továbbítása előtt, ami plusz késleltetéssel jár.

Ha a routernek csak azt a feladatot szánjuk, hogy „megfogja” a szórásos csomagokat, akkor valójában kifizetődőbb választás routerek helyett VLAN-ok használatával szegmentálni a szórási tartományt. A switchek (alapesetben) az OSI modell 2. rétegében dolgoznak, így a csomag továbbítását hardveresen meg tudják oldani, ami miatt sokkal gyorsabbak tudnak lenni. Ha VLAN használatára konfigurálunk egy switchet, akkor megadhatjuk neki, hogy az egyes interfészekre csak bizonyos VLAN csomagjait továbbítsa, ezáltal a szórási címre küldött csomagok az általunk megalkotott logikai határokon belül maradnak.

Gondok a VLAN használatával

Mivel a VLAN-okat a LAN-ok leváltására vezettük be, jogos lenne az elvárás, hogy a switchek nagyjából hasonló feladatokra legyenek képesek, mint a routerek. A routereknek három fő feladatát néztük eddig:

- elválasztja egymástól a szórási tartományokat
- összeköt távoli hálózatokat
- forgalomirányítást végez

Ebből a három pontból az elsőt teljesíti a switch. Az utolsót nem, de nme is várjuk el tőle, mert a switchek továbbra is helyi hálózatok kialakításáért felelősek.

Viszont a switch nem tud átjárást biztosítani két VLAN között, ahogy a routerek LAN-ok között. Az eddigi példáinkban olyan eseteket néztünk, ahol ez nem is volt elvárás, de a valós életben ez egy természetes igény. Nem beszélve arról, hogy így a VLAN-ba tartozó gépek a külvilággal sem tudnak kommunikálni, ami a mai világban (néhány kivételtől eltekintve) elképzelhetetlen.

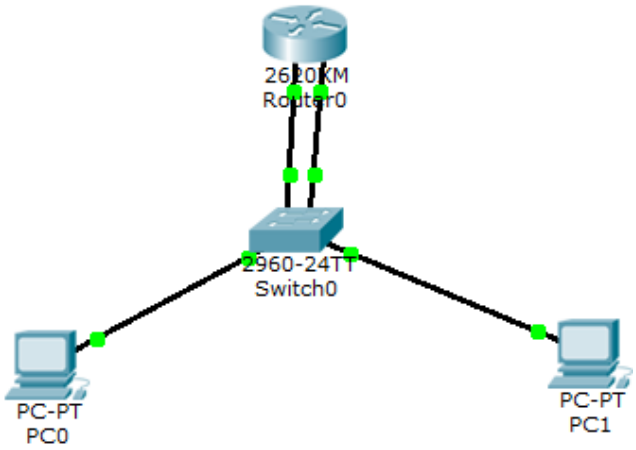
Megoldás

A fent említett hátrányt pont az okozza, ami egyben az előnye is a VLAN-nak: a VLAN-ok szerinti szegmentálás 2. rétegben történik, a switch pedig nem mérlegel, így nem is tudunk neki szabályokat, kivételeket adni. Ezt úgy tudjuk csak áthidalni, hogy feljebb lépünk az OSI modellben és rábízzuk a 3. rétegre az összekötést. Ennek az összekötésnek három, elég hasonló változata van, mindegyiket meg fogjuk nézni és megnézzük, hogy lehet összerakni Packet Tracerben.

Inter VLAN megvalósítási módok

1. Változat: 1 router – n port

Ez a megközelítési mód nagyon hasonlít a hagyományos LAN-oknál látott megoldásra: vegyünk egy routert, egy-egy interfészére csatlakoztassunk egy-egy VLAN-t, így, ha VLAN-on kívülre tart egy csomag, akkor az a routerhez kerül, és ő továbbítja egyik portjáról a másikra. Egy nagyon egyszerű példával szemléltetve, így fog kinézni a hálózatunk:



Ennek a topológiának a felépítése és konfigurálása nem igényel új ismeretet, csak a régi tudást kell megfelelően kombinálni. A beállítás lépései:

- Felvesszük a switch VLAN adatbázisába a két VLAN-t (itt 10 és 20)
- Kiosztjuk a switch gépek felé néző access portjait: balra 10, jobbra 20
- IP címet és default gatewayt állítok be a gépeknek.
- IP címet állítok be a router interfészeinek.
- A switch router felé néző portjait a megfelelő VLAN-ba teszem.

Két dologra kell nagyon figyelni a fenti lépések végrehajtásakor. Az első az, hogy a hálózatban a VLAN topológiáját követnie kell az IP címzésnek is, azaz amely gép különböző VLAN-ban van, az legyen különböző alhálózatban is. Erre azért van szükség, hogy egy másik VLAN-ban szereplő gépnek címzett csomag „felkerüljön”

a harmadik rétegbe. Ugyanis, ha azonos alhálózatban lennének (IP szerint), akkor megpróbálná elküldeni neki közvetlenül, ezt viszont már láttunk, hogy nem fog sikerülni. Azonban, ha IP cím szerint más hálózatba tartozik, akkor nem is kísérletezik az eljuttatással, egyből elküldi az alapértelmezett átjárónak.

A másik dolog, amire nagyon oda kell figyelni, hogy a switchnek a router felé néző portjai is access módba kerüljenek, és ezek feleljenek meg az adott VLAN alhálózati címzésének. Nézzük meg konkrét számokkal:

	VLAN10	VLAN20
IP címtartomány	192.168.10.0/24	192.168.20.0/24
Router interfészek	FastEthernet 1/0	FastEthernet 1/1
Switch portok	FastEthernet 0/1, 0/3	FastEthernet 0/2, 0/4

Tehát, a switch Fa0/3-as portja fog kapcsolódni a router Fa1/0 portjára. A switch Fa0/3 portja a 10-es VLAN-t továbbítja access módban, a router interfésze pedig a 192.168.10.0/24 címtartományból kap egy címet.

> ***„Ha úgyis más alhálózatban vannak a gépeink, akkor miért van szükség VLAN-ra? A csomagok így sem mennek át a másik hálózatba!"***
>
> Merülhetne fel jogosan a kérdés. De ne feledjük, a szórásos csomagok 2. rétegen továbbítódnak, míg az IP címzés 3. réteg. Tehát a szórásos csomag „megkerüli" a címfelosztásunkat és átjut a más alhálózatba is.

Hátrányok

A fenti megoldásnak az egyszerűsége a hátránya is egyben, ugyanis ettől rugalmatlanná válik. Mi történik akkor, ha felveszünk egy újabb VLAN-t? Akkor kell egy újabb interfész. És ha még kettőt? Akkor még kettőt fel kell venni. Amellett, hogy a routereket nem lehet a végtelenségig bővíteni, nem is gazdaságos ez a megoldás. Emellett aránytalan lehet a hálózat terhelése, ha a különböző VLAN-ok eltérő mértékben forgalmaznak adatot (ld. következő bekezdések).

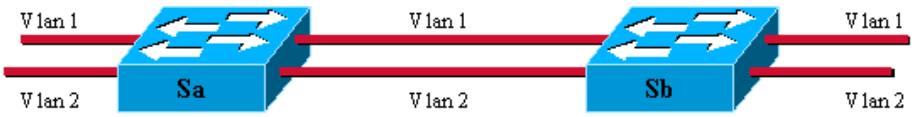
2. változat: 1 router – 1 port (Router on a stick)

A fent említett probléma foglalkoztatta a hálózati fejlesztőmérnököket is, ezért kitaláltak erre is egy megoldást. Ez pedig – a már ismerős – trunk port.

Trunk port és DOT1Q header

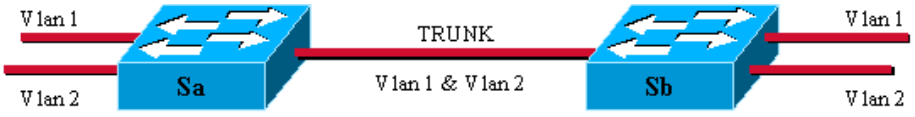
A VLAN-okról szóló részben már előkerült a trunk port fogalma, de akkor nem sokat beszéltünk róla, megelégedtünk annyival, hogy mire való. Most nézzük meg kicsit közelebbről.

Cisco terminológia szerint a trunk port egy olyan pont-pont összeköttetés, amely több VLAN-nak szánt csomagot is továbbítani tud. Tulajdonképpen segítségével portokat takaríthatunk meg, ha két, VLAN-okat használó switchet szeretnénk összekötni egymással. Alapvetően egy port egyszerre egy VLAN-t képes továbbítani (*access mód*). De mi van akkor, ha két switch között két VLAN átjárhatóságát szeretnénk biztosítani?



A fenti ábrán látható, hogy ahhoz, hogy a **Sa** jelzésű switch továbbítani tudja **Sb** jelzésűnek a **Vlan1** és **Vlan2** csomagjait is, két portra lenne szükségünk. Azt leszámítva, hogy ez pazarló megoldás, nem is lenne rugalmas: három VLAN esetén még egy, négy VLAN esetén még két port felhasználására lenne szükség. Emellett a terheléselosztás sem optimális: lehet a **Vlan1** nagyon gyakori forgalmat generál, míg **Vlan2** nagyon keveset forgalmaz (így neki felesleges egy saját összeköttetés).

A megoldást a *trunk* port biztosítja nekünk. Ekkor egy fizikai kapcsolaton keresztül több logikai kapcsolat is lehet (azaz egyszerre több *VLAN* is közlekedhet rajta).



Ez úgy lehetséges, hogy amikor a switch fogad egy csomagot az egyik VLAN-tól és a *trunk* porton keresztül kell továbbítania, akkor kiegészíti a kapott csomagot (becsomagolja, *encapsulate*) egy új *tag* mezővel (**802.1Q header**) és úgy küldi tovább. Ez a *tag* tartalmazza a VLAN tulajdonságait, többek között a VLAN azonosítót (*VID*). A fogadó eszköz ez alapján tudja megállapítani, hogy kinek szól ez a csomag.

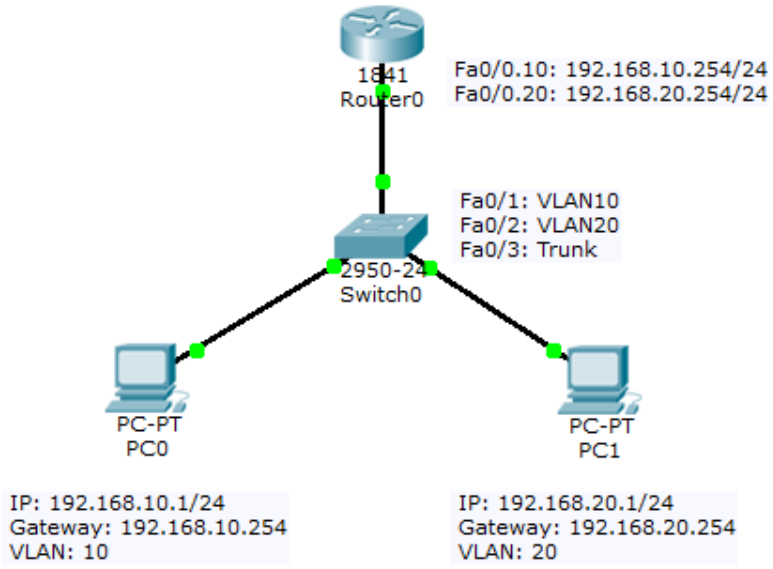
Ezzel csak az a probléma, hogy ez a fejléc egy plusz kiegészítés, nem része a szabványnak, így a hálózati eszközök nem ismerik fel. Tehát, ha ilyen csomagot kaphatnak, akkor előtte fel kell készítenünk a fogadásukra őket. (ld. következő gyakorlati példa)

Megvalósítás

Ennek a megoldásnak az alapelve nagyon hasonló az előzőhöz: vegyünk fel a hálózatba egy routert, a VLAN-okat helyezzük különböző IP címtartományokba, így, ha más VLAN-ba küldünk csomagot, akkor az továbbítódik a routernek és majd az összeköti a VLAN-okat. Az egyetlen különbség, hogy fizikailag csak egy

portot kell használnunk, amire a switch egy trunk porton keresztül fog csatlakozni.

Bár a router egy fizikai interfészt használ az összes VLAN csomagjainak fogadására és továbbítására, a VLAN-ok továbbra is külön alhálózatban vannak (ezeknek viszont kellene külön-külön interfész). Ennek az ellentétnek a feloldására alkalmas a router portjainak alinterfészekre bontása (*subinterfaces*). Hasonlóan a VLAN-hoz, logikailag osztunk fel egy fizikai egységet: egy fizikai interfészt több logikai interfészre. Módosítsuk az előző hálózatunkat úgy, hogy csak egy fizikai interfészt használjunk.



Ehhez a szükséges lépések:

- Távolítsuk el (egy kivételével) a switch-router közötti összeköttetéseket.
- A megmaradt egy vonalat a switchen állítsuk trunk módba.
- A megmaradt vonalnak a router felöli portján töröljük a konfigurációt (*no ip address* paranccsal), de hagyjuk bekapcsolva.
- Hozzunk létre egy-egy alinterfészt minden VLAN számára. Állítsuk be mindegyiken az **.1Q** header értelmezést és adjuk neki az előzőleg megállapított IP címet!

Ezekből a lépésekből egyedül az alinterfészek konfigurálása az újdonság, így ezt részletesebben is megnézzük.

Az alinterfész a router egy portjának (interfészének) a logikai felosztása, mely lehetővé teszi, hogy egy fizikai interfészen több logikai hálózatot is kezelni tudjon. Egy alinterfészt az alábbi paranccsal tudunk kiválasztani

```
Router(config)#interface fastEthernet [fizikai port].[alinterfész]
```

Pl. a Fa0/0 port 10. alinterfészét így választhatjuk ki:

```
Router(config)#interface fastEthernet 0/0.10
```

Ezután meg kell adnunk, hogy ez az interfész milyen beágyazás formátumot (esetünkben **802.1Q**) fogad.

```
Router(config-subif)#encapsulation dot1q [vlanID]
```

Ezzel adjuk meg, hogy **802.1Q** headert használunk, és a **vlanID**-ből származó csomagokat fogadjuk ezen az alinterfészen.

Végül egy IP címet is kell rendelnünk ehhez az alinterfészhez:

```
Router(config-subif)#ip address 192.168.10.1 255.255.255.0
```

Ez utóbbi parancs megegyezik a fizikai interfésznél látottakkal. A fenti utasítás sorozatot hajtsuk végre az összes VLAN-ra, amiket össze szeretnénk kötni. A felhasznált parancsokat megtaláljátok a mellékletben.

3. változat: L3 switch

Az előző bekezdésben látott „router on a stick" módszer már megfelelő számunkra: rugalmas (trunk port), logikai felosztást használ (alinterfészek) és tudja továbbítani a csomagokat egyik VLAN-ból a másikba, sőt, akár külső hálózatba is (lásd 3+1 fejezet). Azonban mégis van egy szépséghibája ennek a megközelítésnek: minden olyan csomag, ami nem saját VLAN-ba irányul, eljut a routerhez. Ha külső hálózatra megy a csomag, akkor ez elkerülhetetlen, viszont, ha azonos LAN-on, de másik VLAN-ban van, akkor ez felesleges kitérőnek tűnhet és lassul az átvitel, ugyanis a router továbbra is lassabban tud dolgozni a switchnél, mert az utóbbi funkciói hardveresen vannak megvalósítva. Ez az igény vezetett el egy új hálózati eszköz bevezetéséhez: a Layer 3 switchekhez. Ezek a switchek képesek hardveres szinten csomagokat továbbítani egyik alhálózatról a másikra, így össze tudják kötni a VLAN-okat is.

Megvalósítás

Egy Layer 3 switch beállítása sem különösen bonyolult, bár elsőre furának tűnhet, mert kicsit mintha kevernénk a switchet és a routert (de valójában valami ilyesmi is történik). A hostok és a hagyományos (layer 2) switchek beállítása ugyanúgy történik, mint eddig:

- Különböző VLAN-ba rendelt gépeknek különböző alhálózatot osztok.
- A switchek VLAN adatbázisát feltöltöm, kiosztom a gépek felé néző access portokat, valamint a Layer 3 switch felé néző portokat trunk módba teszem.

A többi tennivalónk a Layer 3 switchen lesz. Az alábbi lépésekre van szükségünk:

- Fel kell venni mindegyik VLAN-t az adatbázisba
- Be kell kapcsolni a routing funkciót
- Virtuális interfészeknek IP címet kell adnom, majd bekapcsolnom őket

A VLAN adatbázisba a L2 switcheknél látott módon tudom felvenni az egyes virtuális hálózatokat. A forgalomirányítást az *ip routing* paranccsal tudom bekapcsolni. Ezek után már csak a virtuális interfészek konfigurálása van hátra. Magukat az interfészeket máshogy jelölöm ki, de attól kezdve ugyanúgy tudom kezelni, mint egy router interfészt.

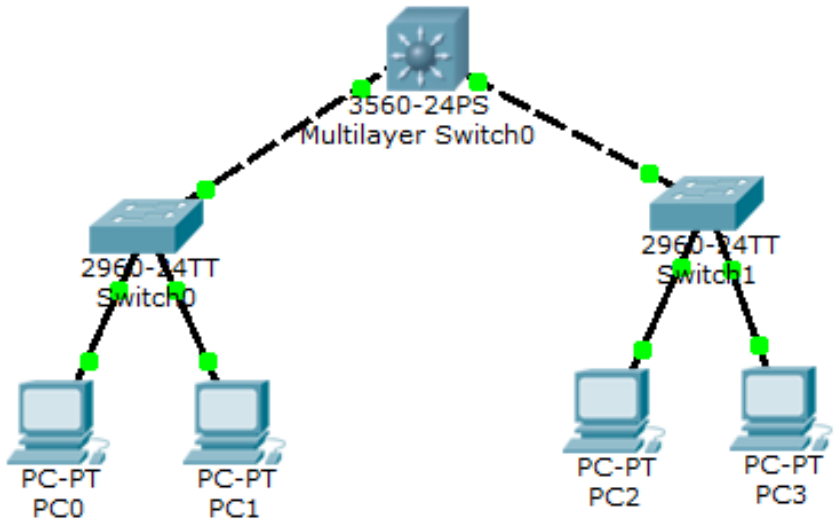
A L3 switch **vlanID** számú VLAN-jához rendelt virtuális interfészét az alábbi paranccsal tudom kijelölni:

```
Switch(config)#interface Vlan [vlanID]
```

Ezután ennek az interfésznek a már megszokott módon tudok IP címet adni és bekapcsolni:

```
Switch(config-if)#ip address [ip cím] [alhálózati maszk]
Switch(config-if)#no shutdown
```

Ezt meg kell csinálni mindegyik VLAN esetében. Használhatjuk azokat az IP címeket, amiket a router esetében (hiszen most ezek fogják helyettesíteni a router interfészeit). Ha ezeket a beállításokat végrehajtjuk, működőképes lesz a hálózatunk.



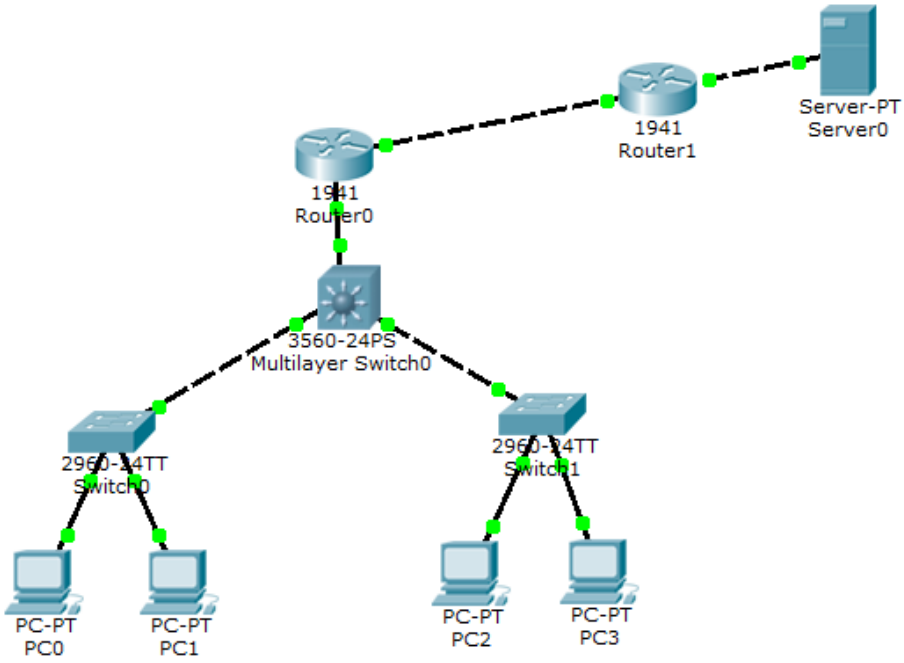
Az eddig látott három lehetőség közül ez a legelőnyösebb változat:

- A hardveres megvalósítás miatt gyorsabb, mint a másik két routeres megoldás
- A LAN-on belüli forgalmat egyedül bonyolítja, nincs szüksége a routerre, csak azt továbbítja, ami külső hálózatba irányul (azt nem bírja kezelni).
- Bár drágább, mint egy L2 switch, olcsóbb, mint a router.

A hálózat teljes beállításához szükséges parancsoka a mellékletben találhatók.

3+1. változat: L3 switch + router

Az előző bekezdésben felmerült, hogy külső hálózatba tartó csomaggal a Layer 3 switch nem tud mit tenni, továbbítani fogja a routernek. De így is nagy segítség a rotuernek, mivel csak azokat fogja továbbküldeni, amit feltétlenül muszáj, így nagy terhelést vesz le róla. Bővítsük ki az előző hálózatunkat egy routerrel, és egy jelképes „távoli hálózattal", amit most a szerver fog jelölni.



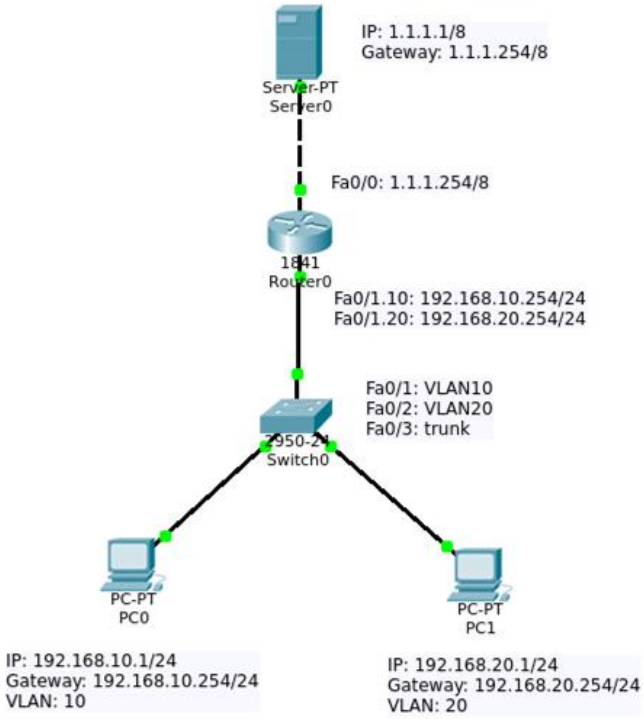
Ahhoz, hogy a VLAN-ba rendezett gépek kommunikálhassanak a külső szerverrel, meg kell oldanunk, hogy az L3 switch továbbadja a routernek a döntés jogát a továbbítással kapcsolatban. A beállítás lépései:

- Kiválasztjuk a L3 switch router felé néző portját.
- Kikapcsoljuk ezen a porton a switchinget a no switchport paranccsal
- Adunk neki valamilyen egyedi IP címet, amin keresztül eléri a külső routert.
- Bekapcsoljuk a portot.

Ilyenkor a hostok gateway címét sem kell módosítani, mert ők küldik a L3 switchnek, aki úgy viselkedik a szemükben, mint egy router.

Gyakorló feladat

Erre külön nem tértünk ki, de magától értetődik, hogy bármelyik routeres megvalósítás segítségével is elérhető a külső hálózat. Valójában, miután bekapcsoltuk a routernél a .1Q fejléc kezelését, úgy fog viselkedni, mintha két LAN-t csatlakoztattunk volna két külön fizikai portjára. Éppen ezért, a fenti hálózatnak a megépítése maradjon gyakorlófeladat (felhasználható a második módszer megoldása kiindulásnak)



Felmerülő kérdések

A fentiekben láthattuk, hogy megoldható a VLAN-ok közötti unicast kommunikáció (a *broadcast* üzenetek továbbra is megakadnak a routeren vagy a L2 switcheken). Mi a teendő akkor, ha azt szeretnénk, hogy a két VLAN egymással semmilyen módon ne tudjon kapcsolatba kerülni egymással, de a külső hálózattal igen?

Ez is megvalósítható: hozzáférési listákat (ACL) kell definiálnunk a routeren. Ezekről bővebben az utolsó két órán lesz szó.

Források

.1Q header

http://en.wikipedia.org/wiki/IEEE_802.1Q

Trunk port

http://www.cisco.com/c/en/us/support/docs/switches/catalyst-6000-series-switches/14970-27.html

Router on a stick konfiguráció

http://www.cisco.com/c/en/us/td/docs/ios/12_2/switch/configuration/guide/fswtch_c/xcfvl80q.html
http://www.orbit-computer-solutions.com/How-to-Configure-Router-on-a-Stick-InterVLAN-Routing.php

Layer 3 switchek

https://www.lifewire.com/layer-3-switch-817583
http://www.cisco.com/c/en/us/about/press/internet-protocol-journal/back-issues/table-contents-19/switch-evolution.html

Inter VLAN routing Layer 3 switchekkel

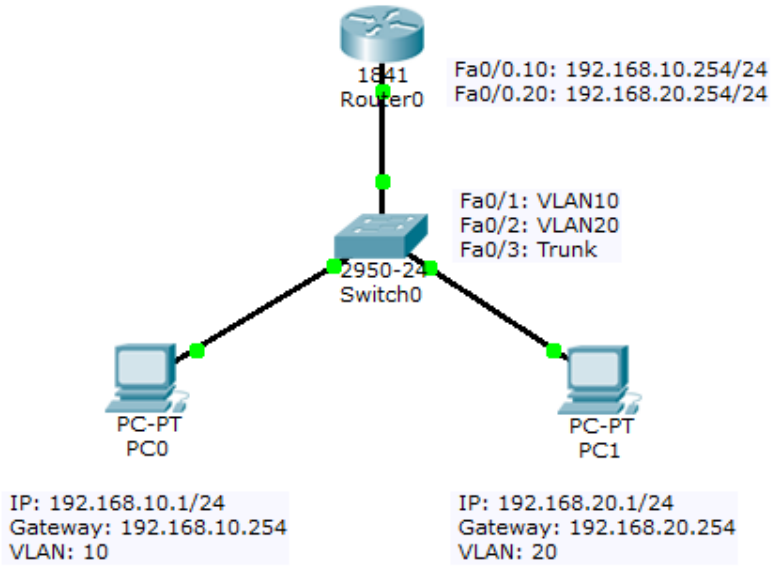
http://www.cisco.com/c/en/us/support/docs/lan-switching/inter-vlan-routing/41860-howto-L3-intervlanrouting.html

Beugró kérdések

1. Miért továbbítja gyorsabban a csomagot a switch, mint a router?
2. Mi történik akkor, ha egy L2 switch egy VLAN felől érkező csomagot trunk portra továbbít?
3. Mit nem tud a L2 switch, amit a router igen? (Több helyes válasz is van.)
4. Miért előnyös a trunk port? (Több helyes is van.)
5. Mi a router alinterfésze?
6. Hogy választjuk ki egy router alinterfészét?
7. Miért fejlesztették ki a L3 switcheket? (Több helyes válasz is lehet.)
8. Mit nem kell megtennem az alábbiak közül az L3 switch konfigurálásánál?
9. Miért jobb az inter VLAN routing az L3 switchekkel megvalósítva?
10. Mit tudunk tenni, ha a két VLAN-t teljesen el akarjuk zárni egymástól, de külső hálózatba engedni szeretnénk?

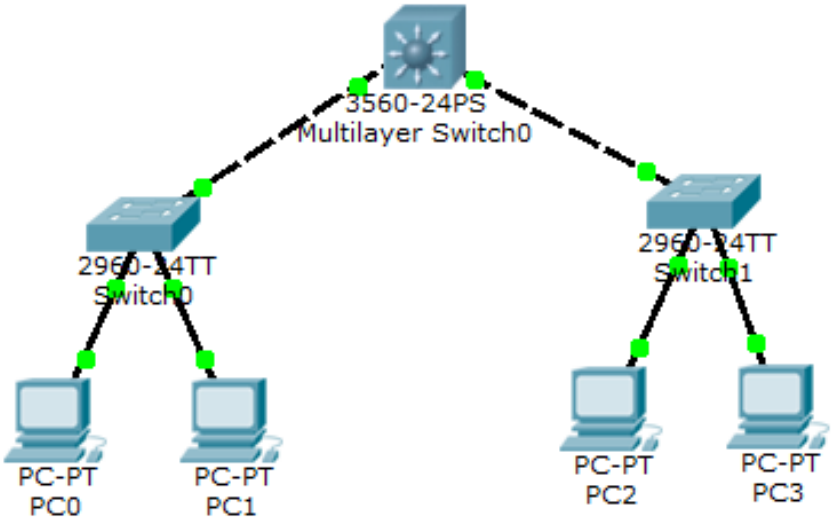
Melléklet

1. Router on a stick CLI parancsok




```
Router>enable
Router#configure terminal
Router(config)#interface fastEthernet 0/1.10
Router(config-subif)#encapsulation dot1q 10
Router(config-subif)#ip address 192.168.10.254 255.255.255.0
Router(config-subif)#exit
Router(config)#interface fastEthernet 0/1.20
Router(config-subif)#encapsulation dot1q 20
Router(config-subif)#ip address 192.168.20.254 255.255.255.0
```

2. L3 Switch CLI parancsok




```
Switch>enable
Switch#configure terminal
Switch(config)#ip routing
Switch(config)#vlan 10
Switch(config-vlan)#exit
Switch(config)#vlan 20
Switch(config-vlan)#exit
Switch(config)#interface Vlan 10
Switch(config-if)#ip address 192.168.10.254 255.255.255.0
Switch(config-if)#no shutdown
Switch(config-if)#exit
Switch(config)#interface Vlan 20
Switch(config-if)#ip address 192.168.20.254 255.255.255.0
Switch(config-if)#no shutdown
Switch(config-if)#exit
```