Ősz 2018

UNIVERSITAS SCIENTIARUM SZEGEDIENSIS
UNIVERSITY OF SZEGED
Department of Software Engineering

# Számítógép-hálózatok 11. gyakorlat
# Virtuális Helyi Hálózatok (VLAN)
# Inter VLAN routing

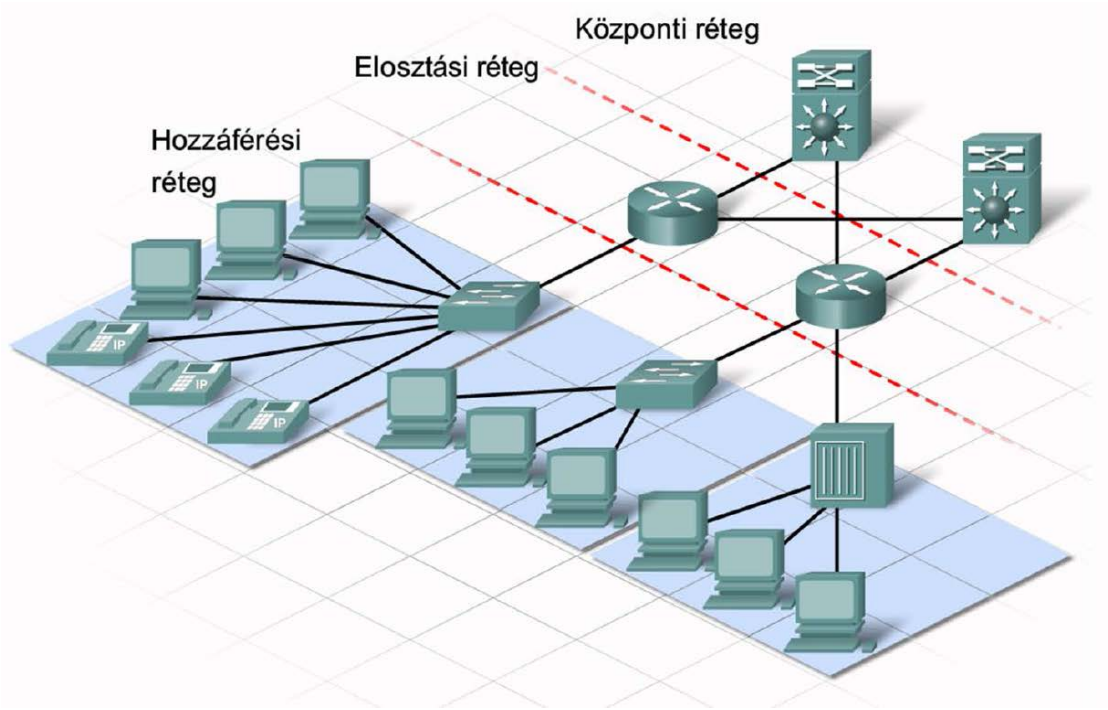
Bordé Sándor, Maczák Bálint

# Tartalomjegyzék

## Bevezetés

Az eddig órákon azzal foglalkoztunk, hogy hogyan lehet helyi hálózatokat kialakítani switchek (azaz kapcsolók) segítségével, illetve ezeket hogy tudjuk összekötni routereken (azaz forgalomirányítókon) keresztül. Most tovább lépünk a hozzáférési rétegbe, és megnézzük, hogy lehet a helyi hálózatunk szórási tartományát szegmentálni, a hálózatelérési rétegben.

## A hálózatok méret szerinti csoportosítása




A világméretű hálózatokat logikailag három rétegbe szokták sorolni: hozzáférési réteg, elosztási réteg és központi réteg. Mindegyik rétegnek van egy rá jellemző eszköze és egy vagy több feladata.

| | fő feladat | jellemző eszköz |
|---|---|---|
| **hozzáférési réteg** | eszközök összekapcsolása helyi hálózattá | switch |
| **elosztási réteg** | helyi hálózatok összekapcsolása egymással | router, L3 switch |
| **központi réteg** | nagy sebességű, nagy teljesítményű, megbízható kapcsolat távoli hálózatok között | nagy teljesítményű routerek, switchek |

Ezen kívül a hálózatokat gyakran csoportosítják földrajzi kiterjedés szerint is. Ez alapján három kategóriát különböztethetünk meg: *LAN, MAN* és *WAN*.

### LAN (Local Area Network)

Magyarul helyi hálózatnak nevezzük. Ahogyan a nevéből is látszik, főként kisebb kiterjedésű rendszerek jelölésére használják (de ez nem mindig teljesül, ld. következő bekezdés). A kifejezés önálló helyi hálózatokra, vagy közös adminisztratív irányítás alatt álló, egymással összekötött helyi hálózatok csoportjára is vonatkozhat.

A hálózatok kialakulásának kezdeti időszakában gyakran fizikailag egy helyen elhelyezkedő, kisméretű hálózatként határozták meg. Ma viszont már ideérthetjük akár az otthoni, akár a több száz gépet, több épületet és helyet magában foglaló vállalati hálózatokat is.

A LAN-ok jellemzően *Ethernet* vagy *Wireless* kapcsolatokat használnak, és nagy sebesség jellemzi őket. Az *Intranet* gyakran egy szervezethez tartozó privát LAN-t jelöl, amit úgy terveztek, hogy csak a szervezet tagjai férhessenek hozzá.

### MAN (Middle Area Network)

A LAN-nál nagyobb hálózatot jelent, de kevésbé használt kifejezés. Méretben valahol a LAN és a WAN között van, és egy nagyobb területet fed le, mint például egy város. Gyakran a kormányzatok és nagyobb szervezetek számára van jelentősége. A gyakorlaton nem fogunk a továbbiakban foglalkozni *MAN*-okkal.

### WAN (Wide Area Network)

Ha egy vállalat több, egymástól távoleső telephellyel rendelkezik, akkor szükség lehet egy távközlési szolgáltató (**TSP** – **T**elecommunication **S**ervice **P**rovider) bevonására a különböző LAN-ok összekapcsolásához. Ezeket a távoleső LAN-okat összekapcsoló hálózatokat nagytávolságú hálózatoknak (*WAN*) nevezzük.

## Kapcsolók

### Fizikai címzés

Az Ethernet hálózatokban használt fizikai cím neve a MAC-cím. A csatolóhoz annak legyártása során rendeli hozzá a gyártó, megváltoztathatatlan. 48 bit hosszú bináris számsorozat, amit hexadecimális alakban 8 bites tagolásban szoktunk megadni. Például 00:30:04:B8:40:C3, ahol az első 3 oktett a gyártót, második 3 oktett a csatolót azonosítja. A szórásos MAC-cím olyan cím, aminek minden bitje 1-es (FF:FF:FF:FF:FF:FF). Helyi hálózaton egyértelműen azonosítja a forrás és célállomást, de csakis helyi hálózaton, arról nem ad információt, hogy a készülék melyik hálózatban van, tehát nem hierarchikus címzési fajta. A hálózati csatoló csak akkor fogadja a keretet, ha annak cél MAC-címe megegyezik az ő (fogadó fél) csatolójának MAC-címével vagy a hálózat szórásos MAC-címével. Vagyis a szórásos MAC-címre küldött üzenetet a helyi hálózat összes tagja fogadja.

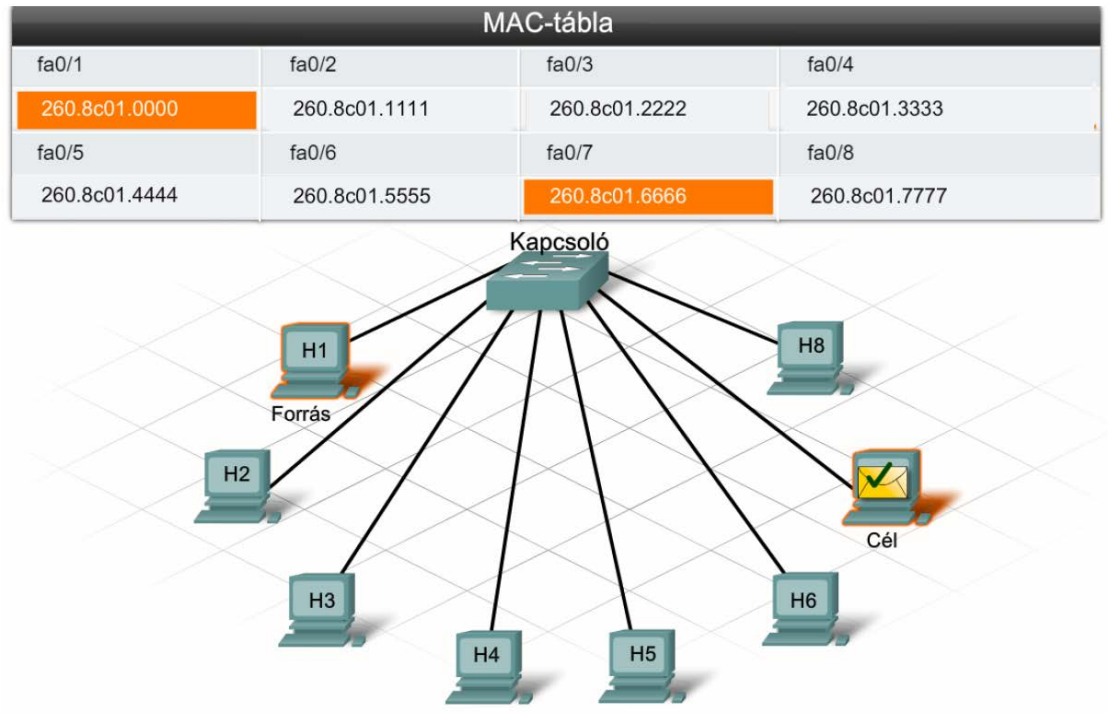
Szórási tartománynak hívjuk azt a területet, amelyben szórásos üzenet esetén az állomások üzenetet kapnak.

## Ethernet közeghozzáférés

Az Ethernet hálózatok CSMA/CD közeghozzáférést alkalmaznak. Ennél a módszernél a küldő állomás mielőtt küldene „belehallgat" a csatornába, hogy használja –e éppen egy másik állomás vagy sem, ha senki sem használja, akkor elküldi az üzenetet. A küldő által küldött üzenetet minden állomás megkapja és eldöntheti az üzenetben lévő cél MAC-cím alapján neki szól-e, ha nem, akkor eldobja az üzenetet, ha igen, akkor feldolgozza. Ütközés akkor lép fel, ha a csatornán egyszerre egy időben több fél szeretne kommunikálni, ez abból derül ki, hogy az adó is veszi a maga által küldött üzenetet, és összehasonlítja, hogy egyezik-e azzal, amit küldött, ha nem akkor hibás üzenetnek jelenti a többi félnek és megszakítja a további üzenetküldést, majd bizonyos ideig várakozik.

Ütközési tartománynak hívjuk azt a területet, amelyben ütközés esetén az állomások sérült üzenetet kapnak.

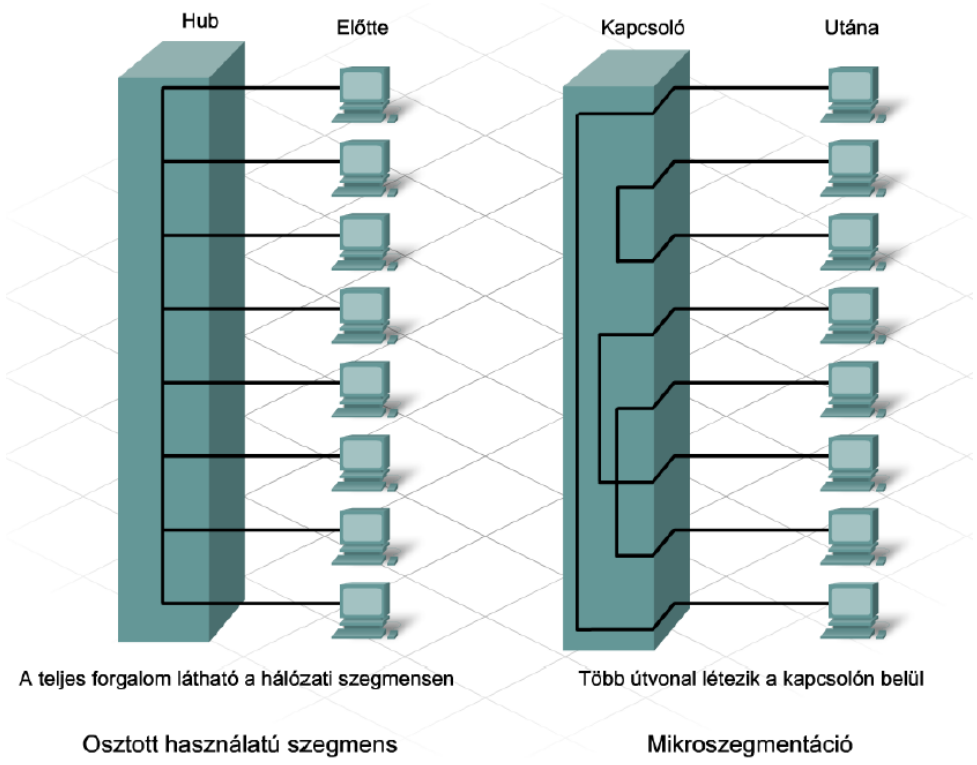
## Kapcsoló működése




*Kapcsolón keresztüli üzenetküldés, a MAC-tábla szerepe*

A kapcsoló képes arra, hogy csak egy meghatározott állomásnak továbbítson egy üzenetet. Amikor az állomás egy másik állomásnak a kapcsolón keresztül küld üzenetet, a kapcsoló fogadja és dekódolja a keretet, majd kiolvassa belőle a fizikai célcímet. A kapcsolók által használt tábla, melyet MAC-cím táblának hívnak, tartalmaz egy listát az aktív portokról és a hozzájuk csatlakoztatott állomások MAC-céméről. Amikor egyik állomás üzenetet küld a másiknak, a kapcsoló ellenőrzi, hogy a cél MAC-cím megtalálható-e a táblázatban. Ha igen, akkor felépít egy áramkörnek nevezett ideiglenes logikai kapcsolatot a forrás és célport között. Ennek a dedikált kapcsolatnak a sávszélességén nem osztozik más, csak a cél és forrás állomás. Minden egyes új párbeszédnél új áramkör jön létre (mikroszegmentáció), ezáltal egyidejűleg több párbeszéd is folyhat, anélkül, hogy ütközés következhetne

be. Ha a célállomás nem található meg a kapcsoló MAC-cím táblájában, akkor elárasztás történik, a kapcsoló a keretet mindenkinek továbbítja, az eddig nem ismert célállomás is megkapja, nyugtát küld róla a kapcsolónak. Ezután a kapcsoló érzékeli, hogy eddig ismeretlen MAC-cím üzent neki, elraktározza a táblájába, megtanulja azt. A kapcsoló felesleges módon nem terheli a hálózatot, mert csak azon portján továbbítja az üzenetet, amelyiken megtalálható a címzett.




*Mikroszegmentáció szemléltetése*

A mikroszegmentációnak köszönhetően ütközési tartomány csak a logikai áramkörön belül, azaz két port között áll fenn. Azonban az Ethernet üzenetszórásokat minden portján továbbítja. Tehát egy hatalmas szórási tartományt hoz létre. Egy hálózattervezési ökölszabály, hogy az üzenetszórásokat a hálózat lehető legszűkebb területére kell korlátozni, erre (is) jók a VLAN-ok létrehozása (lásd következő bekezdés, adatszórás).

## VLAN (Virtual LAN)

### Miért lehet szükség virtuális helyi hálózatokra?

A helyi hálózat többnyire a fizikailag egymás közelében levő gépek halmazát jelöli. A hubokat és switcheket alkalmazó Ethernet révén viszont lehetővé vált, hogy a LAN-okat ne fizikai közelség, hanem logikai kapcsolódás szerint alakítsák ki. Korábban, ha egy vállalat *k* darab LAN-t akart, akkor vásárolnia kellett *k* darab switchet. Ha pontosan szabályozták és felügyelték, hogy melyik csatlakozót melyik switchbe dugják, akkor a LANok tagjait elhelyezkedéstől függetlenül, a szervezeti felépítésnek megfelelően lehetett változtatni. Persze, ha két ember ugyanannál a részlegnél, de messzebb, más épületben dolgozik, akkor valószínűleg más LAN-

hoz fognak tartozni. Ez így nem egy tökéletes megoldás, mert körülményes tervezést kíván a hálózatépítőktől.

A rendszergazdák számára több okból is kedvezőbb, ha egy LAN nem a felhasználói csoportok fizikai elhelyezkedést tükrözik, hanem az intézmény szervezeti felépítését.

Az első szempont a **biztonság**. Sok esetben vannak olyan felhasználói csoportok egy vállalatnál (például egy kutatási-fejlesztési részleg), amelyek információit nem szeretnénk azon a körön kívül látni. Ilyenkor célszerű őket egy LAN-ba tenni, mert így egy fokkal magasabb biztonság érhető el. A vállalati vezetés viszont nem szívesen hallja, hogy egy ilyen felosztást csak akkor lehet megvalósítani, ha a részleg összes dolgozóját egymás közelébe (pl. szomszédos irodákba) költöztetik, mivel ez igen nagy átrendezésekkel járhat. Ezen kívül problémát jelentenek a szervezeti változások, ezek ugyanis egy nagyobb cég életében szinte mindennaposak. Viszont, ha a LAN-okat fizikai helyük szerint rendeznénk, akkor hatalmas többletmunkával és költséggel járna átszervezni a hálózatot.

A második szempont a **terhelés megosztása**. Azokat a LAN-okat, amiket sokkal intenzívebben használnak, érdemes lehet elkülöníteni a kevésbé igény bevett hálózatoktól. Ha például a kutatási részlegen mindenféle remek kísérletet folytatnak, amik felett néha elvesztik az uralmukat és telítődik a LAN-juk, akkor a könyvelési osztályon lehet, hogy nem fognak örülni neki, ha az ő hálózatuk erőforrásait is igénybe veszik.

A harmadik szempont az **adatszórás** kérdése. A legtöbb LAN támogatja az adatszórást, és sok felsőbb rétegi protokoll alaposan ki is használja ezt. Ez meglehetősen nagy forgalmat generál, így egy rosszul megtervezett hálózaton elég sok felesleges adat fog áramolni, ami így szükségtelenül terheli az egész rendszert. Ezen kívül, ha egy eszköz meghibásodik, akkor folyamatosan adatszórásos üzenetekkel (*broadcast storm*) áraszthatja el a hálózatot, ami így meg is bénulhat.

Látható tehát, hogy ha különválasztjuk a logikai és fizikai topológiákat, akkor a hálózatunk egyrészt sokkal áttekinthetőbbé, másrészt sokkal költséghatékonyabbá válik. Ezen indokok hívták létre a virtuális helyi hálózatokat.

## A VLAN-ok felépítése

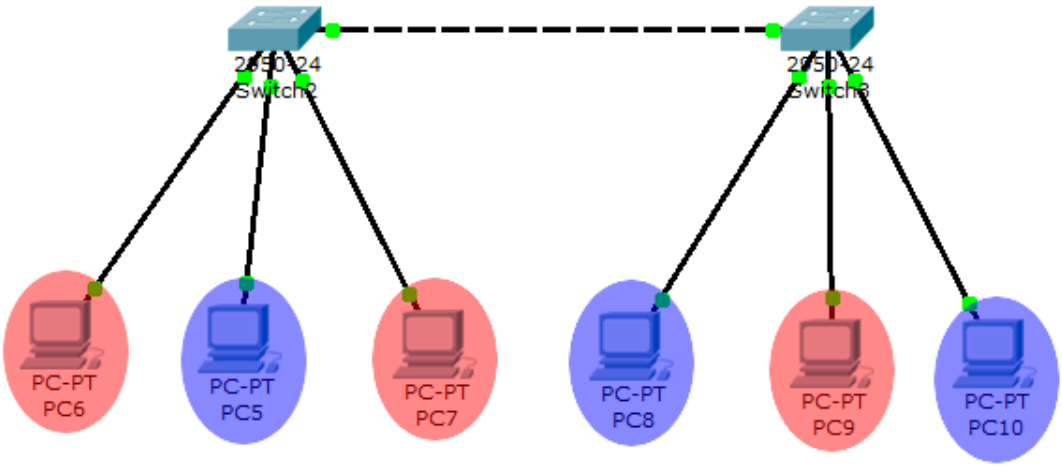
A VLAN-ok - azaz virtuális helyi hálózatok – egy szórási tartományba[1] sorolnak olyan állomásokat, amelyek fizikailag különböző szórási tartományban vannak.

Ezzel a módszerrel biztosítani tudjuk, hogy a fizikailag távol levő állomások is egy LAN-hoz tartozhassanak. Az **1. ábrán** egy példa is látható erre, ahol a különböző VLAN-okat más színekkel jelöltük. Két VLAN látható, piros és a kék színnel jelöltük.

Ahhoz, hogy a VLAN-ok helyesen működjenek, az egyes switcheken egy-egy táblázatot kell feltölteni, ami alapján megmondják, hogy az egyes VLAN-okat me-

[1] Szórási tartományon a hálózat egy olyan logikai felosztását értjük, amiben minden csomópont képes egymást elérni a hálózatelérési rétegen keresztül üzenetszórás segítségével. Például az ugyanarra a switchre kapcsolódó gépek ugyanabba a szórási tartományba tartoznak.

lyik port(ok) segítségével lehet elérni (ebből következik, hogy egy porton keresztül akár több VLAN-t is elérhetünk). Ha egy üzenet érkezik, mondjuk az egyik piros gépről, akkor a switch csak a piros gépek közül keresi a címzettet. Ebből adódik, hogy a két VLAN között nincs átjárás, és ez teljesíti a korábban elvárt követelményeket is.



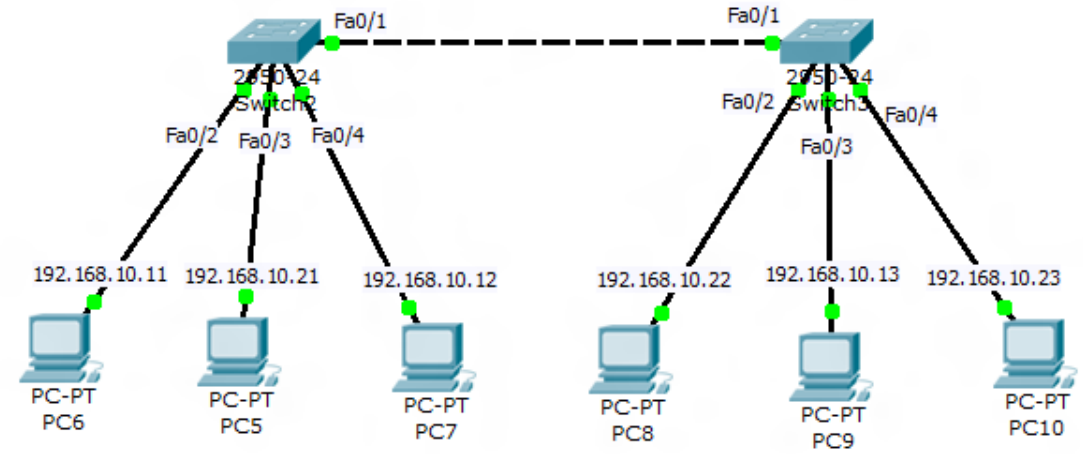

***1. ábra**, „piros" VLAN és „kék" VLAN*

Eddig nem beszéltünk arról, hogy az egyes csomagokról hogyan állapítjuk meg, melyik VLAN-hoz tartoznak. Eredetileg az Ethernet kereteknek nem volt semmilyen tartalék mezője a VLAN azonosító számára, így meg kellett változtatniuk az Ethernet csomagok fejrészét. Az új szabványt IEEE 802.1Q néven adták ki. Így a switchek az Ethernet keretből informálódnak arról, hogy az áthaladó csomag melyik VLAN-hoz tartozik, és ezek alapján merre kell tovább küldeni.

## VLAN-ok építése Packet Tracerben

A következő fejezetben az első ábrán látott hálózatot fogjuk megépíteni, és beállítjuk a VLAN-okat.

A hálózatunkat első lépésben konfiguráljuk a **2. ábrán** látható módon. (*Megjegyzés*: a gyakorlaton switcheket használunk, de ugyan az érvényes a hubokra is.)




***2. ábra**, a beállítandó hálózat*

Az egyes gépek IP címeit az ábrának megfelelő módon állítsuk be (az alhálózati maszk mindenhol 255.255.255.0 legyen, az alapértelmezett átjárót pedig hagyjuk üresen, mivel most nem lesz rá szükség). Figyeljünk arra, hogy a megfelelő csatlakozók a megfelelő portokra kerüljenek (a **FastEthernet** jelölés helyett a rövidebb **Fa** jelölést használjuk).

Ezek után a switchek fogják vezérelni, hogy melyik porton melyik VLAN-hoz tartozó eszköz érhető el. A konfigurálást kezdjük a Switch2-vel. Rákattintva nyissuk meg a konfigurációs ablakot.

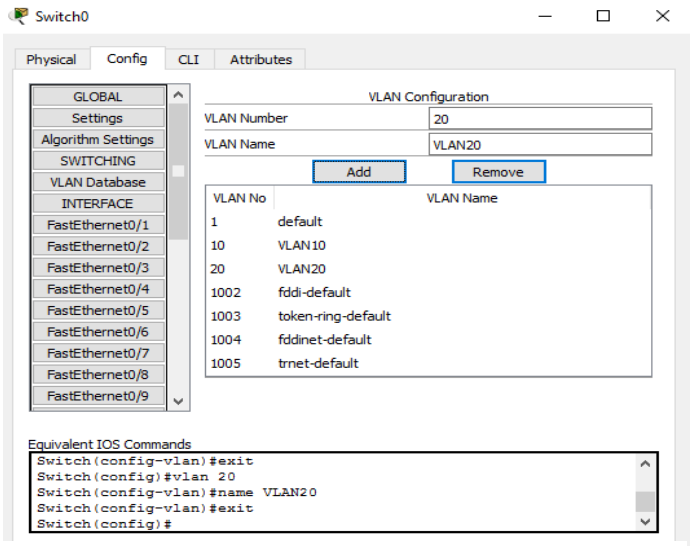
A VLAN-t kétféleképp is konfigurálhatjuk: a Config fülön, vagy a CLI parancsok segítségével. Mindkét módszert megnézzük.

## Konfigurálás GUI-n keresztül

Kattintsunk a *Config* fülre, majd bal oldalon keressük meg a „*VLAN Database*” menüt. Ezzel tudjuk megmondani a switchnek, hogy milyen VLANokat ismerjen.

Itt két virtuális hálózatot fogunk hozzáadni. Az első azonosítója (*VLAN Number*) legyen **10**, a neve (*VLAN Name*) pedig „**VLAN10**”. Ugyan így a második azonosítója **20**, és a neve **VLAN20**.

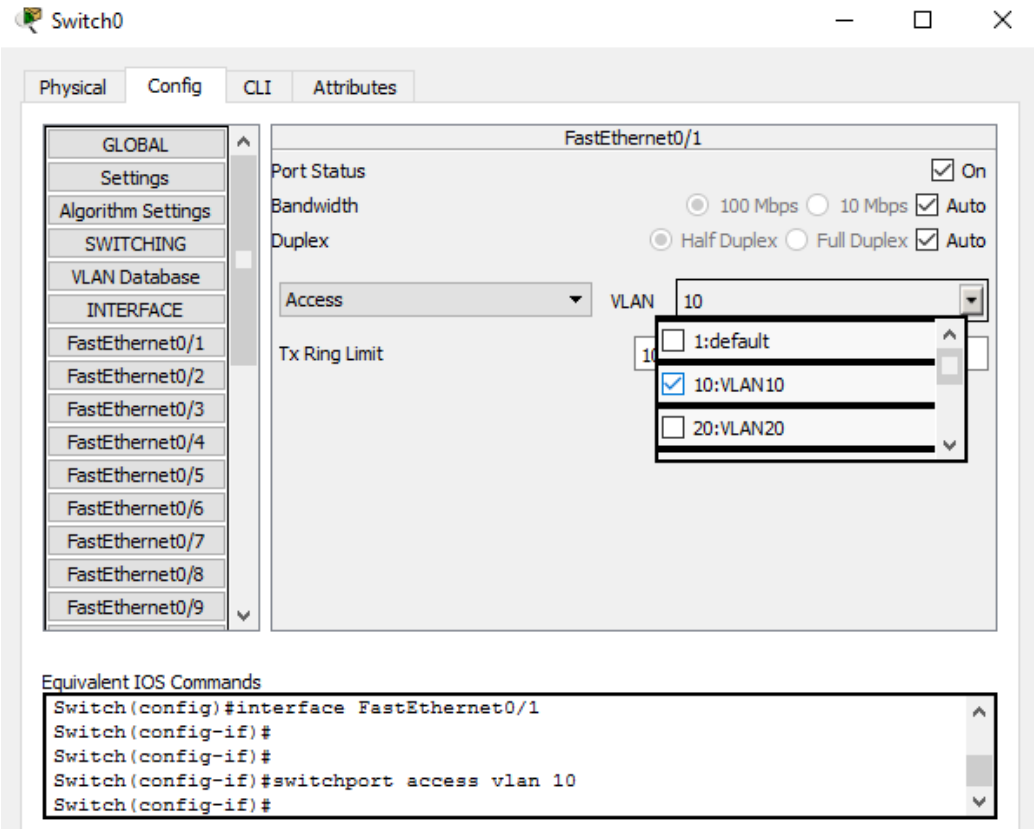
A következő lépés az, hogy megadjuk, melyik porthoz melyik VLAN tartozik.




***3. ábra**, Switch0 konfigurációja*

Nem titkolt szándékunk, hogy a 192.168.10.1x tartományba tartozó gépek legyenek a VLAN10, a 192.168.10.2x gépei pedig a VLAN20 részei.

A **4. ábrán** láthatjuk, hogyan kell konfigurálni egy tetszőleges FastEthernet portot a switchen. Szintén a bal oldali listából kell kiválasztani a konfigurálandó interfészt, majd ezután a lenti módon kell beállítani. Ehhez hasonlóan állítsuk be az Fa0/2, Fa0/4 interfészeket a switch2-n, és az Fa0/3-at a switch3-on. A többi csatlakozó, azaz (Fa0/1 switch2-n és Fa0/2, valamint Fa0/4 a switch3-on) a VLAN20-at kapja.




***3. ábra**, a FastEthernet0/2 VLAN konfigurálása*

A következő ábrán egy összefoglalást láthatunk a beállításokról. Itt megfigyelhető, hogy mindkét VLAN egy alhálózatba tartozik. Szükséges feltétel, hogy egy adott VLAN (pl. VLAN10) minden gépe egy alhálózatban legyen, mert különben nem tudnának egymással kommunikálni. Viszont minden egyes VLAN alkothat külön-külön alhálózatot, ilyenkor viszont egyébként sem tudnak kommunikálni egymással, mert külön alhálózatban vannak.

| Állomás név | IP cím | Ethernet port | Switch név | VLAN név |
|---|---|---|---|---|
| **PC6** | 192.168.10.11 | Fa0/2 | Switch2 | VLAN10 |
| **PC7** | 192.168.10.12 | Fa0/4 | Switch2 | |
| **PC9** | 192.168.10.13 | Fa0/3 | Switch3 | |
| **PC5** | 192.168.10.21 | Fa0/3 | Switch2 | VLAN20 |
| **PC8** | 192.168.10.22 | Fa0/2 | Switch3 | |
| **PC10** | 192.168.10.23 | Fa0/4 | Switch3 | |

Amennyiben a fentiekkel készen vagyunk, a VLAN-unk már majdnem teljesen működőképes. Már csak annyit kell megtenni, hogy mindkét switchnél a *FastEthernet0/1* konfigurációját megnyitjuk a bal oldali listából, és az „*Access*”-t átállítjuk „*Trunkra*”. A Trunk (magyar zsargonban „*trönk*”) egy olyan összeköttetés két switch között, amely egyszerre több VLAN forgalmát is át tudja engedni. Tehát a két switchet úgy kell bekonfigurálnunk, hogy egymás felé több VLAN forgalmát is át tudják engedni. A másik mód, az „*Access*” a trönk ellentéte, mivel itt csak egy VLAN-t adhatunk meg, így ez csak azokat a csomagokat fogja áthereszteni, amelyek arra a virtuális hálóra mennek.

### Konfigurálás CLI-n keresztül

Itt is hasonló lépéseket kell végrehajtanunk, mint az előzőekben. Először nyissuk meg az egyik switch konfigurációs ablakát, majd váltsunk a CLI fülre. A VLAN Database-t fogjuk először feltölteni. Mindezek előtt privileged módba kell lépnünk, ezt az `enable` paranccsal tehetjük meg:

```
Switch>enable
```

Ezután belépünk a konfigurációs módba:

```
Switch#configure terminal
```

Itt a következő parancsokkal tudjuk hozzáadni a VLAN10-et:

```
Switch(config)#vlan 10
Switch(config-vlan)#name VLAN10
```

Majd az exit paranccsal kilépünk a VLAN konfigurációból:

```
Switch(config-vlan)#exit
```

Végül ismételjük meg a folyamatot VLAN20-ra, illetve a másik switchen is VLAN10re és VLAN20ra!

Amint készen lettünk a fenti parancsok beírásával, konfigurálni kell a *FastEthernet* portokat is. Hasonlóan, lépjünk be az egyik switch CLI felületére, privileged módba, és azon belül is konfigurációs módba:

```
Switch>enable
Switch#configure terminal
```

Ezek után lépjünk be az első (munkaállomáshoz kapcsolódó) interfész konfigurációs módjába:

```
Switch(config)#interface FastEthernet 0/2
```

Majd a következő utasításokkal be tudjuk állítani, hogy ezen a porton a 10-es VLAN-ra, azaz a VLAN10-re érkező csomagokat továbbítsa.

```
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
```

Itt az első utasítás az „Access” és „Trunk” módok közötti választás, ahol Access-t választunk, mivel nem trönkként akarjuk kezelni az adott portot. Ismételjük meg ezeket a lépéseket minden szükséges portra.

Miután ezekkel megvagyunk, az `exit` paranccsal innen is kiléphetünk, és legvégül elvégezhetjük mindkét switchre a trönkök konfigurálását. Ezt a következő utasítássorozat adja meg:

```
Switch>enable
Switch#configure terminal
Switch(config)#interface FastEthernet 0/1
Switch(config-if)#switchport mode trunk
Switch(config-if)#exit
```

## Összefoglalás

Röviden a konfiguráció lépései:

1. Felépítjük a hálózati topológiát, azaz lerakjuk az eszközöket és összekötjük őket a megfelelő kábelekkel.
2. Töltsük fel a VLAN adatbázist azokkal a VLAN-okkal, amelyeket használni szeretnénk. Minden olyan VLAN-t vegyünk fel, ami áthaladhat a switchen, ne csak azokat, amiket Access móddal hozzárendelünk valamelyik porthoz!
3. Az egyes állomások felé mutató FastEthernet portokat is állítsuk be, azaz mindegyikre adjuk meg, hogy melyik VLAN csomagjai haladhatnak arra.
4. A két switchet összekötő portokat állítsuk be trönköknek.

## Inter VLAN routing

Az anyag első részében megnéztük a VLAN-ok használatát és hogy hogyan 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 anyagrészben ezekre a kérdésekre keressük a választ.

### 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 nem 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.

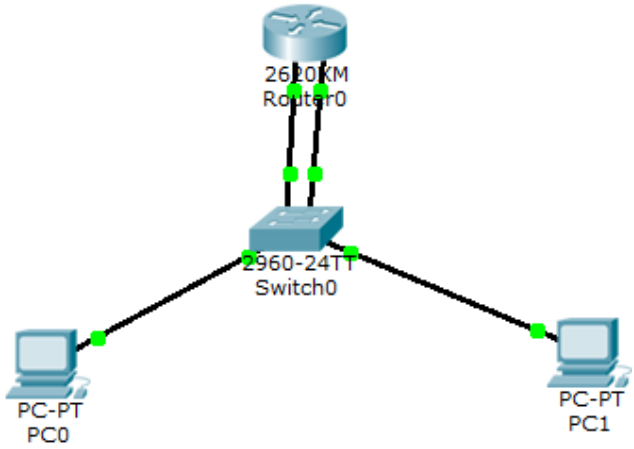
### Megoldás

A fent említett hátrányt pont az okozza, ami egyben az előnye is a VLAN-nak: a VLAN-ok szerinti szegmentálás hálózatelérési rétegben történik, a switch pedig nem mérlegel, így nem is tudunk neki szabályokat, kivételeket adni. Ezt úgy tudjuk csak áthidalni, hogy feljebb lépünk a TCP/IP modellben és rábízzuk az internet 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" az internet 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 hálózat-elérési rétegen továbbítódnak, míg az IP címzés internet 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).

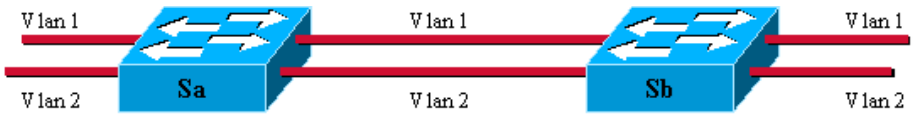
## 2. változat: 1 router – 1 port (Router on a stick)

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.

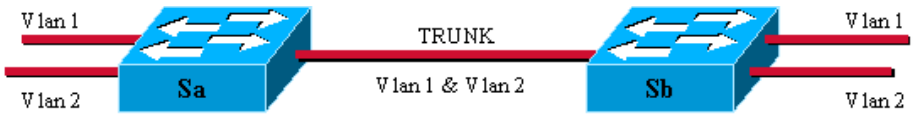
Cisco terminológia szerint a trunk port egy olyan pont-pont összeköttetés, amely több VLAN-nak szánt csomagot is továbbítani tud. Tulajdonképpen segítségével portokat takaríthatunk meg, ha két, VLAN-okat használó switchet szeretnénk összekötni egymással. Alapvetően egy port egyszerre egy VLAN-t képes 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).



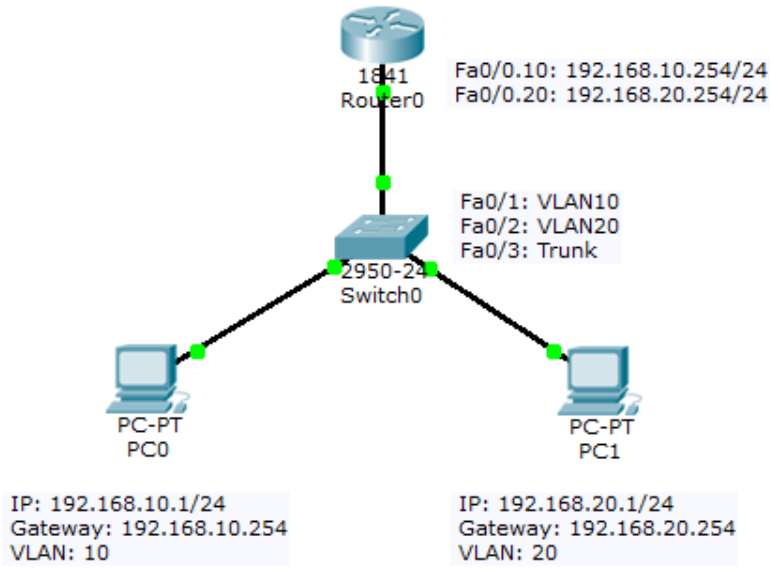

Ez úgy lehetséges, hogy amikor a switch fogad egy csomagot az egyik VLAN-tól és a *trunk* porton keresztül kell továbbítania, akkor kiegészíti a kapott csomagot (becsomagolja, *encapsulate*) egy új *tag* mezővel (**802.1Q header**) és úgy küldi tovább. Ez a *tag* tartalmazza a VLAN tulajdonságait, többek 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.

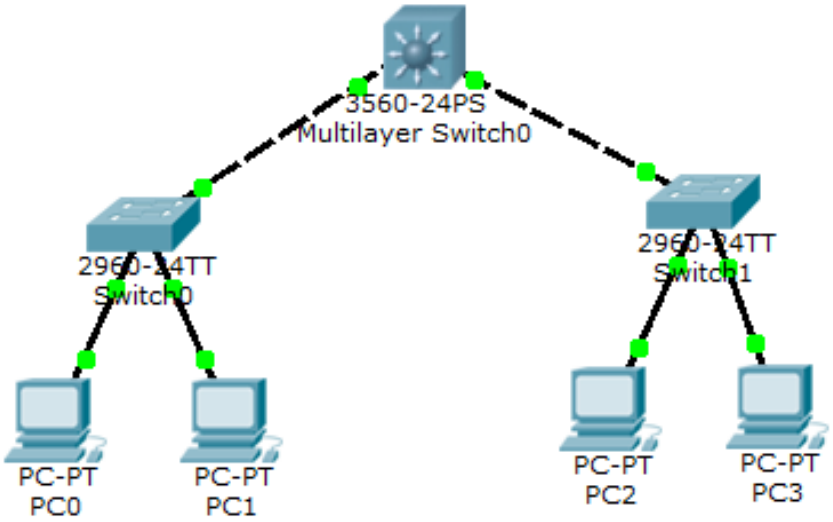
A L3 switch **vlanID** számú VLAN-jához rendelt virtuális interfészét az alábbi paranccsal tudom kijelölni:

```
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.



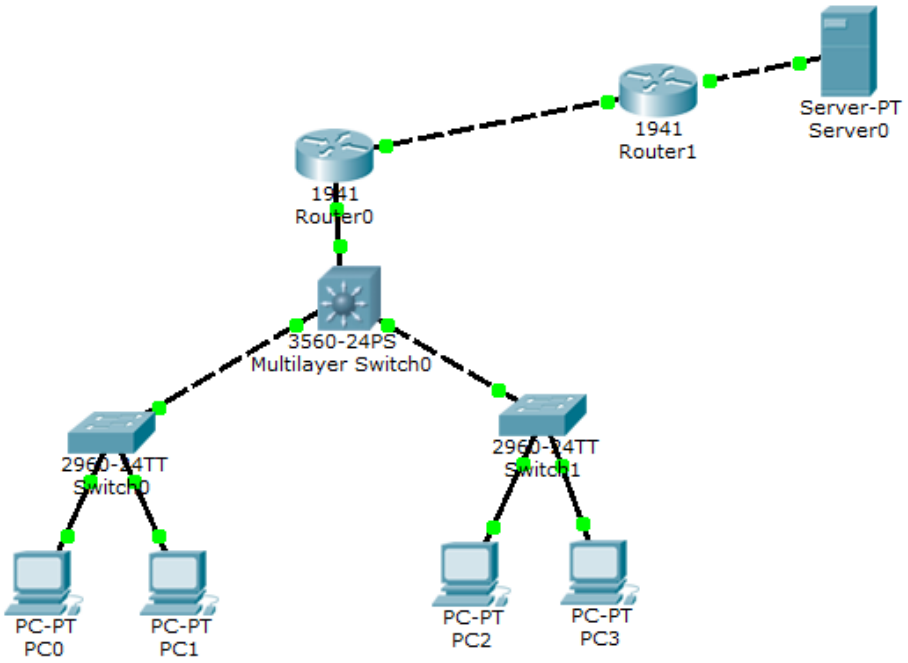

Az eddig látott három lehetőség közül ez a legelőnyösebb változat:

- 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 parancsok 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.

## 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.

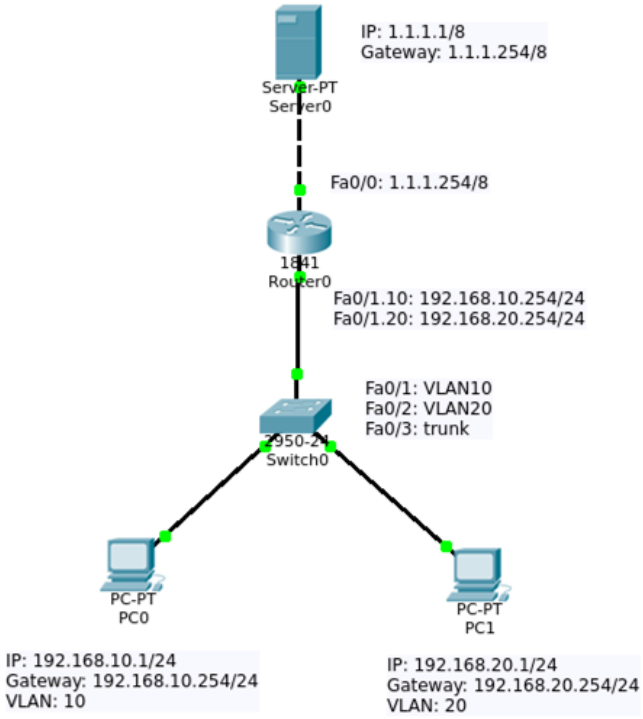
## Beugró kérdések

- Mi a LAN definíciója?
- Milyen előnyei lehetnek a VLAN-oknak?
- Milyen tulajdonságnak kell feltétlenül teljesülnie ahhoz, hogy két számítógépet egy VLAN-ba tudjunk helyezni?
- Mire szolgál a VLAN Database?
- Melyik parancs szolgál egy Ethernet port Access módban a 10-es azonosítójú VLAN-ra állítására?
- Mire szolgál a switchport access vlan 10 utasítás?
- Miért előnyös a trunk port?
- Mi a router alinterfésze?
- Hogy válasszuk ki egy router alinterfészét?
- Mit definiál az IEEE 802.1Q szabvány?

# Mellékletek

## 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)




## Videós segédletek

### VLAN

https://youtu.be/9fKoszSupsg
https://youtu.be/NcvYdKGozPE
https://youtu.be/7-jTfr5l8m8

### Inter VLAN routing

https://youtu.be/R4MfoVW8qEE
https://youtu.be/NnsmweCrXBY
https://youtu.be/xf-KljnCIi0

## Források

Cisco CCNA Discovery tananyag

### .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

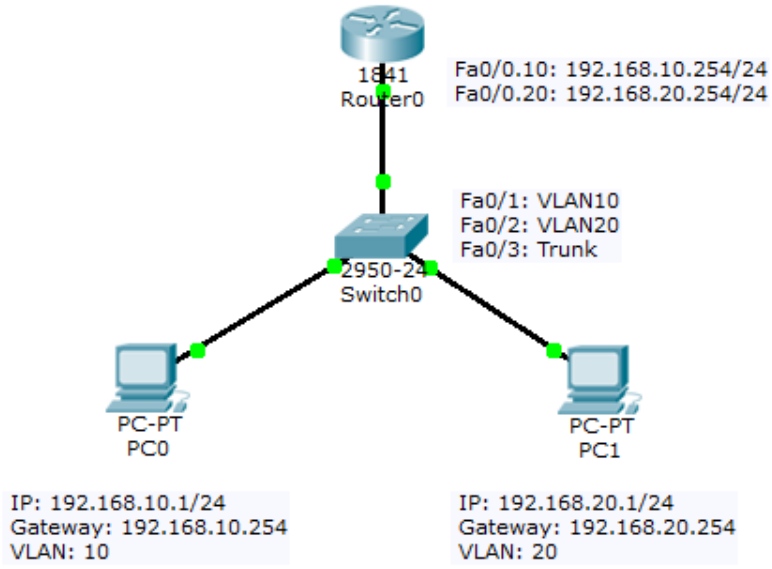
https://www.lifewire.com/layer-3-switch-817583
http://www.cisco.com/c/en/us/about/press/internet-protocol-journal/back-issues/table-contents-19/switch-evolution.html

### 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

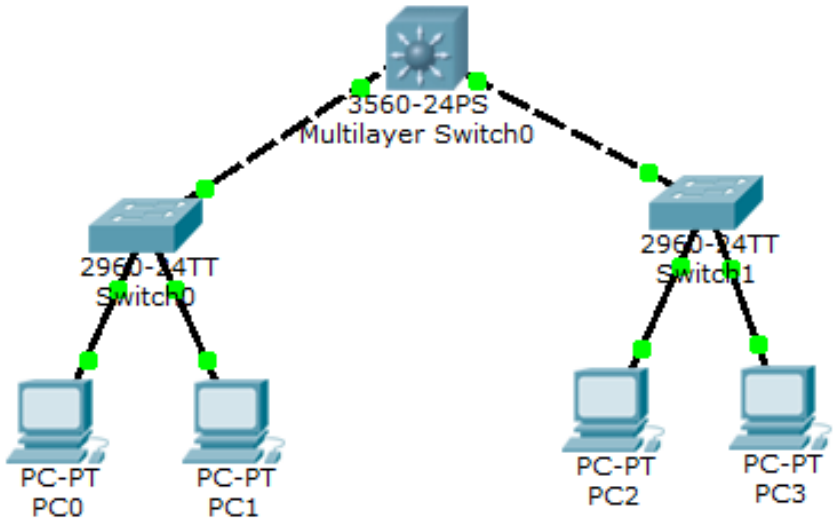
## Példafeladatok megoldásai

### 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
```