Ősz 2018

UNIVERSITAS SCIENTIARUM SZEGEDIENSIS
UNIVERSITY OF SZEGED
Department of Software Engineering

Számítógép hálózatok 9. gyakorlat

Forgalomirányítás (Statikus, RIPv2)

Bordé Sándor, Seres József

Tartalomjegyzék

Bevezetés

A múltkori anyagban megismerkedtünk a Packet Tracerrel és felépítettük az első modellünket benne. Abban a modellben a helyi hálózatok közvetlenül kapcsolódtak egymáshoz, azaz ugyanannak a routernek a különböző interfészein keresztül tartották a kapcsolatot a külvilággal. A valóságban viszont ez nagyon kevés esetben van így.

Ahhoz, hogy az egymástól távolabb lévő hálózatok is kommunikálhassanak egymással, szükségünk lesz forgalomirányítási szabályokra. Ezek a szabályok mondják meg, hogy egy hálózattól egy másik hálózatig milyen úton jutunk el. Ebben a jegyzetben áttekintjük ezeknek a szabályoknak két nagy csoportját (statikus és dinamikus szabályok), majd nézünk mindkettőre egy-egy példát.

A forgalomirányítás a hálózati, vagy más néven IP rétegnek a feladata. Az IP protokollról egy későbbi gyakorlaton beszélünk részletesebben.

Elméleti háttér

Ha az előző gyakorlaton épített hálózatunkat kiegészítenénk még egy routerrel, és a két LAN két különböző routerre csatlakozna, akkor azt figyelhetnénk meg, hogy a korábban tanult ismeretek nem elegendőek ahhoz, hogy a két hálózat kapcsolatba léphessen egymással. Ha a Packet Tracer szimulációs módjában végigkövetjük a csomag útját, akkor azt látjuk, hogy mindkét hálózatból a routerig eljut, viszont onnan nem megy tovább.

Ennek az az oka, a routerünk nem tudja, mit kell tenni ezzel a csomaggal: annyit lát csak, hogy a célhálózat címe nem egyezik meg egyik interfészén látható hálózattal sem, így eldobja a kapott csomagot. A csomag eldobását úgy tudjuk elkerülni, hogy forgalomirányítási szabályt adunk a routernek, ami meghatározza, hogy ilyen esetekben mit kell tenni.

Forgalomirányítási szabály részei

A forgalomirányítási szabályok három fő részből állnak: célhálózat címe, odavezető útvonal következő pontja és az odavezető útvonal költsége.

Célhálózat címe

Minden szabályban benne van, hogy melyik hálózat felé tartó csomagokra érvényes, ezt a célhálózat címével adhatjuk meg. Az előző, alhálózatokról szóló anyagból tudjuk, hogy egy hálózatot a hálózat címével és a hozzá tartozó alhálózati maszkkal adhatunk meg egyértelműen. A szabály felvétele után, ha érkezik a routerhez egy olyan csomag, ami nem közvetlenül csatlakozó hálózat felé tart, akkor a router a csomagban található IP címből az alhálózati maszk segítségével (bitenkénti logikai ÉS művelettel) megállapítja a célhálózat címét, majd pedig

kikeresi, hogy melyik szabály vonatkozik rá. (Ha nem találja, és nem adtunk meg neki alapértelmezett viselkedést, akkor eldobja.)

Útvonal költsége

Egy routertől egy hálózatba több út is vezethet egyidőben. Ezek közül a router úgy választ, hogy megnézi, melyik a legkisebb költségű útvonal. Ezt a költséget nem nekünk kell majd megadni, hanem forgalomirányító protokollok (algoritmusok) fogják kiszámolni, ezekről a későbbi fejezetben részletesebben beszélünk.

Útvonal következő pontja

Az internetet a feltalálásakor úgy tervezték, hogy minél kevésbé legyen sebezhető. Ennek az egyik módja az volt, hogy az egyes hálózatok között nem csak egy közvetlen kapcsolat legyen, hanem több elérhető útvonal is. Így, ha az egyik tönkremegy vagy megsemmisül, akkor a kommunikáció még folytatódhat. A hálózatokat elképzelhetjük egy gráfként: a gráf csomópontjai a routerek, a gráf élei pedig a routerek közötti közvetlen összeköttetések.

Emiatt a tagoltság miatt egy forgalomirányítási szabály az nem azt adja meg, hogy egy hálózattól egy másikig milyen úton jutok el, hanem mindig csak egy lépéssel közelebb irányít a célhoz. (Lehet, hogy amíg egyik routertől átjut a másikig, megszűnne az az útvonal, amit eredetileg használni akart.)

Statikus forgalomirányítás

Statikus forgalomirányítás esetén mi határozzuk meg, hogy melyik célhálózatba tartó csomag melyik másik csomópontra (next hop) legyen továbbküldve az aktuális routerről, és ez a szabály egészen addig életben is marad, amíg azt a rendszergazda nem változtatja meg vagy nem törli. (Ezért hívják statikusnak.) Az útvonal hosszának csak akkor van szerepe, ha egy hálózatba több útvonal is megy. Viszont a statikus módnak éppen az a lényege, hogy konkrétan meg vannak határozva az irányok, így ebben az esetben útvonal költségre nem lesz szükség.

A statikus forgalomirányításnak van néhány kézenfekvő előnye. Egyrészt, a routernek nem kell „gondolkodnia”: ha van szabály, tartsuk be, ha nincs, dobjuk el a csomagot. Ez erőforrást takarít meg a routernek. Másrészt kézben tarthatjuk vele az egész hálózatot, mivel mi határozzuk meg a csomag irányát.

Ha kicsit belegondolunk, ezek a fenti előnyök csak akkor elérhetők, ha a teljes hálózathoz van hozzáférésünk, valamint viszonylag kicsi a hálózat. Ez a két pont nagy kiterjedésű hálózatokra (pl. az Internet) nem teljesül, így ott nem praktikus a statikus forgalomirányítás a következők miatt:

- Nagy rendszereknél sok router és sok lehetséges hálózat van, emiatt nem egyszerű kitalálni a szabályokat, és sok idő a beállítása.

- Ha változik a hálózat topológiája, akkor nagy munkával jár átállítani a forgalomirányítást (ugyanis mindet „kézzel” kell módosítani).
- Esetenként akár nagyon nehezen követhető a hálózat működése.
- Nem eredményez optimális forgalomelosztást, a szabályok nem veszik figyelembe a pillanatnyi állapotot.

Ezekből egyértelműen látszik, hogy bizonyos méret és bonyolultság felett nem éri meg statikus forgalomirányítást használni, így inkább automatikus módszereket részesítenek előnyben a hálózat tervezői.

Dinamikus forgalomirányítás

A fenti gondolatmenet következményeként felmerül az igény, hogy használjunk valamilyen algoritmust, ami majd megkeresi és beállítja helyettünk a routereket. Egy ideális forgalomirányító algoritmusnak az alábbiakat kell teljesítenie:

- **Könnyen beállítható**: a hálózatot gyorsan működésbe tudjuk hozni
- **Könnyen karbantartható**: nagy hálózatban gyakoriak a hibák, így minél kevesebb munkával járjon a javítás, konfiguráció
- **Hibatűrő**: ha a hálózat egy bizonyos része hibás, a forgalom a többi részen ne álljon le
- **Skálázható**: lehessen könnyen új eszközöket beépíteni úgy, miközben a beállításokhoz ne, vagy ne nagyon kelljen hozzányúlni
- **Hatékony:** ossza szét a terhelést a hálózaton, ugyanakkor próbáljon minél rövidebb útvonalakat találni.

A forgalomirányító algoritmusoknak két fő fajtája létezik: **távolságvektor alapú** és **kapcsolatállapot alapú**. A félév során gyakorlaton csak a távolságvektor alapúval fogunk foglalkozni.

Távolságvektor alapú protokollok

A távolságvektor alapú forgalomirányításnál az algoritmus a routerektől származó információt két szempont szerint értékeli:

- Milyen távol van hálózat a forgalomirányítótól? (**Távolság**)
- Milyen irányba kell a csomagot továbbítani a hálózat felé? (**Vektor**)

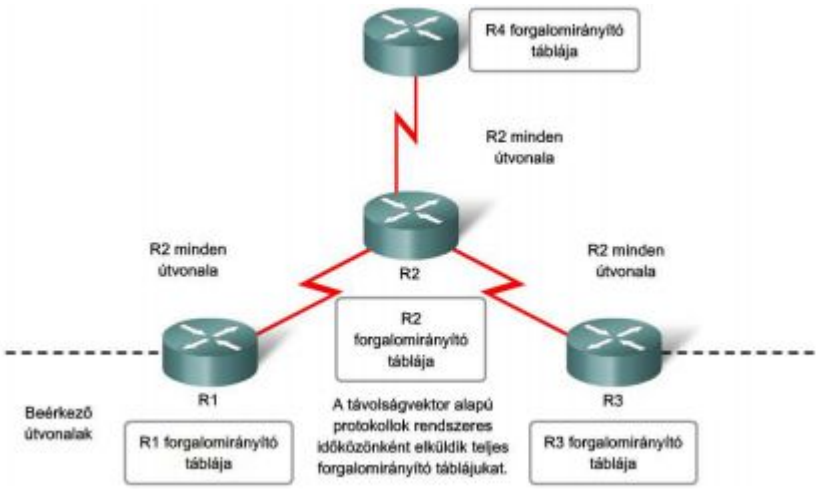
A **távolságot** itt nem fizikai távolságként értjük, hanem különböző tényezőkből származtatott (esetleg súlyozott) mértékként. Ezek a tényezők a következők lehetnek:

- Ugrások száma
- Adminisztratív költség
- Sávszélesség
- Átviteli sebesség
- Késleltetések valószínűsége
- Megbízhatóság

Az útvonal **vektor** összetevője pedig az adott útvonalban a következő csomópont IP címe.

A távolságvektort úgy is elképzelhetjük, mint egy jelzőtáblát a kereszteződésben, ami mutatja, hogy melyik irányban és milyen messze található a cél. Az út mentén pedig további táblák találhatóak, és ahogy közeledünk a cél felé, a távolság egyre csökken.

Minden távolságvektor alapú forgalomirányítást használó forgalomirányító az általa ismert irányítási információkat elküldi a szomszédjainak. Ezt az információ-



ót a szomszédok magukra vonatkoztatják (azaz megnövelik az út távolságát), ezzel jelzik, hogy innen már ennyivel nagyobb költséggel elérni az adott hálózatot. Ezután tovább küldik az ő szomszédjaiknak, és így tovább. Végül minden router a szomszédos routerek információi alapján „tanulja meg" az egyes hálózatok távolságát.

A példa kedvéért nézzük meg a fenti ábrát. Az R2 az R1-től kap információt. R2 megnöveli a kapott táblában szereplő költségértékeket, majd tovább küldi a szomszédjainak, jelen esetben R3-nak és R4-nek, így végül kialakul egy összegzett távolság. Miután kialakult minden router számára egy forgalomirányítási tábla, a hozzá érkező csomagokat ez alapján továbbítja: kikeresi a táblából a legjobb irányt és arra küldi.

RIP (Routing Information Protocol)

Ezt a protokollt az RFC 1058-ban definiálták. Jellemzői:

- Távolságvektor alapú.
- Az útvonal kiválasztásakor az ugrásszámot használja mértéknek.
- A 15 ugrásnál hosszabb útvonalakat elérhetetlennek tekinti.
- 30 másodpercenként elküldi az irányítótábláját a szomszédjainak.

Az útvonalfrissítéskor az előbbiek alapján mindig **eggyel** növekszik a távolság értéke, hiszen egy ugrással több kell a célba jutáshoz. Ezek után azonnal tájékoz-

tatja a hozzá kapcsolódó routereket is a változásokról, és így tovább gyűrűzik a frissítés.

Az RIP könnyen konfigurálható (majd meg fogjuk látni ☺), emiatt széles körben elterjedt. Ennek ellenére néhány hátránya is van, ezek pedig:

- A maximum 15 ugrásnak köszönhetően csak olyan hálózatokban alkalmazható, ahol 16 forgalomirányítónál több nincs sorban egymás után kötve.
- A frissítés jelentős forgalmat jelent nagy hálózatok esetén.
- Nagy hálózatok változása esetén lassan konvergál.

Jelenleg 2 verziója van (ezek a RIPv1 és RIPv2). Az utóbbit szokták használni, mert sokat javítottak rajta az első verzióhoz képest, például támogatja az osztályok nélküli alhálózatokat (tehát az egyedi alhálózati maszkokat).

A RIP a felsorolt hátrányok miatt kisméretű hálózatokban használatos, mert ott nincs szükség bonyolult számításokra. A nagyobb méretű hálózatokban kapcsolatállapot-alapú protokollok használatosak.

Gyakorlati háttér

Statikus forgalomirányítás Packet Tracerben

Ha több hálózatunk van, illetve csatlakozunk más, nagyobb hálózatokra, akkor több routerre lesz szükségünk a hálózat kiépítéséhez. A hálózat felépítését és a routerek alap konfigurálását már könnyen el lehet végezni a korábbi ismereteink alapján. Hozzuk létre az alábbi hálózatot!

1. ábra Két hálózat összekötése

A két router egymással szintén egy alhálózatot alkot, tehát az IP címek kiosztása az alábbi legyen:

Eszköz	Interfész	IP cím
Router 2	fastEthernet 0/0	192.168.1.10
Router 2	fastEthernet 0/1	192.168.3.10
Router 3	fastEthernet 0/0	192.168.2.10
Router 3	fastEthernet 0/1	192.168.3.20

Ekkor a hálózat létrejött. Azonban, ha *ping* csomaggal teszteljük a hálózatot, akkor a két PC között nem jut át a csomag (annak ellenére, hogy a PC-től a routerig, és a két router között is sikeres a csomag küldése).

Ez a probléma abból adódik, hogy a különböző alhálózatoknak más a címzése. Így mikor a bal oldali routerhez megérkezik egy csomag, ami a *192.168.2.1* gépet keresi, a router megnézi a portjait, hogy milyen hálózatokhoz kapcsolódnak. Azt látja, hogy a *192.168.1.0* és a *192.168.3.0* hálózatokat éri el, de nem ide kell mennie a csomagnak, így eldobja. Egyébként viszont szemmel látható, hogy ha továbbítaná a másik routernek a csomagot, akkor rögtön jó helyen lenne.

Egy statikus útvonalszabály megadásához be kell lépnünk *config* módba és ott kiadni az alábbi parancsot:

```
ip route 192.168.2.0 255.255.255.0 192.168.3.20
```

Ahol az első a cél IP cím (itt most az első három szám a hálózat címe, így az utolsó szám lényegében mindegy), a második szám az alhálózati maszk (kiszámítási módját lásd a múlt órai anyagban), a harmadik cím pedig a következő hálózati csomópont. Ugyanezt meg kell csinálni a másik routerrel is, csak más címekkel.

A RIP protokoll konfigurálása Packet Tracerben

Modulbővítés

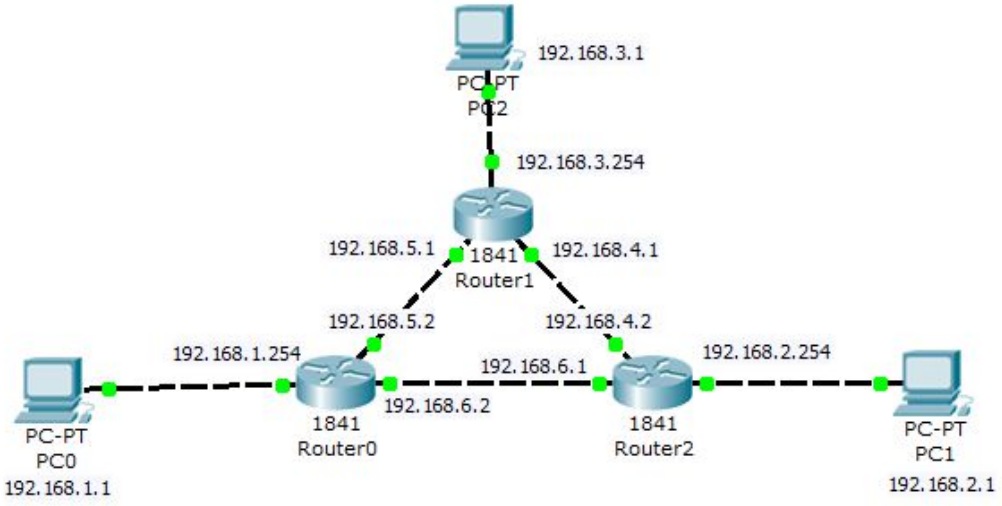
Minden router rendelkezik alapértelmezetten néhány modullal, illetve néhány üres modulhellyel (mérettől függ). A következő példában használt 1841 routereknek alapvetően csak 2 FastEthernet portja van, ami nekünk kevés lesz (mindegyik routernek háromra van szüksége). Ezt úgy tudjuk elérni, hogy az üres modulhelyekre beszerelünk egy új modult, ami két újabb portot bocsájt rendelkezésünkre. (A 1841-es router csak hagyományos Ethernet portokkal bővíthető. Ez a mi szempontunkból csak a nevében tér el, a valóságban is csak a sávszélessége kevesebb.)

A modulcserét a következő lépéssorozattal tudjuk végrehajtani:

1. Rákattintunk a kívánt routerre. Ekkor a Physical „fülön” találjuk magunkat, itt módosítható a router hardveres összetétele. (Figyelem: ha eddig konfiguráltunk bármit is, és nem mentettük el a running configot, akkor azok a következő lépéssel elvesznek.)
2. Kapcsoljuk ki az eszközt! Ezt a jobb oldalon látható kikapcsoló gombra kattintva tehetjük meg.
3. Kattintsunk a bal oldali listában a szükséges modulra (nálunk ez most legyen a WIC-1ENET nevű), majd miközben nyomva tartjuk az egérgombot, húzzuk rá az egyik üres helyre és itt engedjük el a gombot.
4. Kapcsoljuk vissza az eszközt (szintén a bekapcsoló gombra kattintva), és várjunk egy kicsit, míg betölt az IOS.

Alapértelmezett maszkokkal

A RIP protokollt, mint ahogy fentebb ígértük, igen egyszerű konfigurálni. Első lépésben hozzuk létre az alábbi hálózatot (a hálózati maszkok mindenhol az



alapértelmezett 255.255.255.0 értéket veszik fel):

Egyedül azt kell megadnunk, hogy az egyes routerek milyen közvetlen szomszédokkal rendelkeznek, ezek a mi esetünkben a következők:

- **Router0**: 192.168.1.0, 192.168.5.0, 192.168.6.0
- **Router1**: 192.168.3.0, 192.168.4.0, 192.168.5.0
- **Router2**: 192.168.2.0, 192.168.4.0, 192.168.6.0

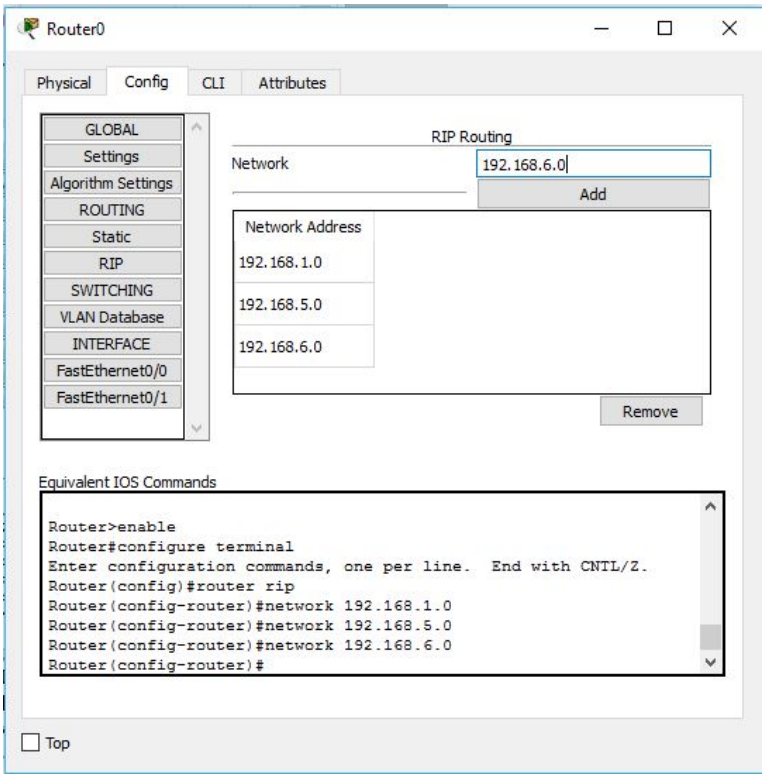
Ezt a konfigurálást parancssorból a következőképpen tudjuk elvégezni:

- Nyissuk meg mondjuk **Router0** CLI parancssorát
- Írjuk be a következő parancsokat:

```
Router>enable
Router#configure terminal
Router(config)#router rip
Router(config-router)#version 2
Router(config-router)#network 192.168.1.0
Router(config-router)#network 192.168.5.0
Router(config-router)#network 192.168.6.0
```

A fenti parancsokkal beléptünk privileged módba, majd a `router rip` parancs segítségével elindítottuk a konfigurációt. Ezután beállítottuk, hogy a kettes verziót használja, majd hozzáadtuk azokat a hálózatokat, amelyek közvetlenül kapcsolódnak a routerhez.

Mindezt megtehetjük akár a grafikus felületen is (itt viszont nem tudjuk megad-

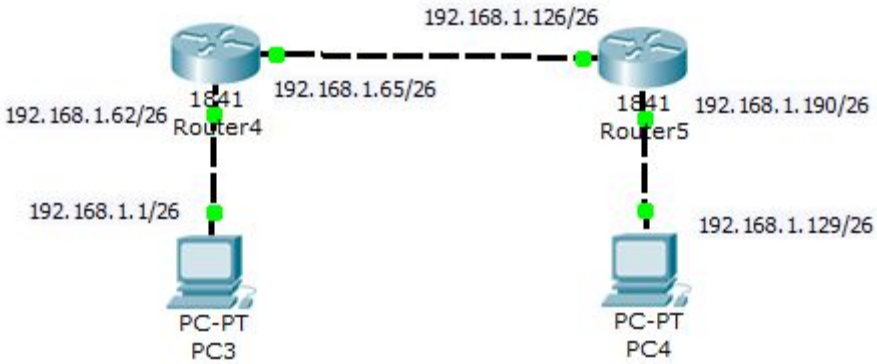


ni, hogy melyik verziót használja a router). Ez a következő ábrán látszik:

Tehát első lépésben megnyitjuk a Router konfigurációs felületét, majd a bal oldali listából kiválasztjuk a RIP menüt. Ezen belül meg már csak hozzá kell adnunk a megfelelő hálózatokat. **Figyelem!** Ez a mód nem biztosítja, hogy a kettes, azaz újabb verziót használja, ezt nekünk kézzel kell megtenni CLI-ben!

Felosztott alhálózatokkal

Vegyük az alábbi hálózatot, amely már alhálózatokat is tartalmaz. Ez esetben a RIP protokollnak elég mindössze a hálózati címet (alhálózati rész nélkül) megad-



ni, tehát így néz ki pl. a **Router4** konfigurálása:

```
Router>enable
Router#configure terminal
Router(config)#router rip
Router(config-router)#version 2
Router(config-router)#network 192.168.1.0
```

Videós segédletek

Statikus forgalomirányítás

https://youtu.be/Pn_5y7qSTus

https://youtu.be/OtweUi5BdjA

RIP protokoll beállítása

https://youtu.be/EnMsNVmK2VY

https://youtu.be/o6dn-KirxoM

Beugró kérdések:

1. Egy forgalomirányítási szabálynak milyen összetevői lehetnek?
2. Mi jellemzi a statikus forgalomirányítást?
3. A RIP forgalomirányítási protokolnak milyen jellemzői vannak?
4. Mik a távolságvektor alapú protokollok jellemzői?
5. Mik a dinamikus forgalomirányítás jellemzői?
6. Milyen parancsokat használhatunk CLI-ben RIPv1 és RIPv2 protokoll beállításához?
7. Mikor melyik dinamikus forgalomirányítási módszert célszerűbb használni?
8. Hogy kell kialakítani a routerek címét, hogy a forgalomirányítás lehetséges legyen?
9. Hogy kell CLI-ben beállítani egy statikus forgalomirányítási szabályt, ami a 192.168.4.0/26 hálózatra érkező csomagokat a 10.1.1.2 címre továbbítja?
10. Ha egy router aminek egyetlen interface-e működik és egyetlen forgalomirányítási szabályt ismer, az "ip route 10.0.0.0 255.0.0.0 192.168.10.2"-t, akkor mit csinál azzal a csomaggal amit a 192.168.1.2-re címeztek?