

Tavaszi

2017

UNIVERSITAS SCIENTIARUM SZEGEDIENSIS
UNIVERSITY OF SZEGED
Department of Software Engineering

Számítógép hálózatok 9. gyakorlat

Forgalomirányító protokollok, RIP

Somogyi Viktor, Bordé Sándor

Tartalomjegyzék

Bevezetés	3
Forgalomirányítás.....	3
Statikus forgalomirányítás	3
Dinamikus forgalomirányítás	4
Forgalomirányító protokollok.....	4
Távolságvektor alapú protokollok	4
RIP (Routing Information Protocol).....	5
EIGRP (Enhanced Interior Gateway Routing Protocol).....	6
A RIP protokoll konfigurálása Packet Tracerben.....	6
Alapértelmezett maszkokkal	6
Felosztott alhálózatokkal	8
Beugró kérdések.....	9
Források.....	10

Bevezetés

Nagyméretű hálózatoknál gyakran előfordul, hogy a topológia igen bonyolulttá, átláthatatlanná válik, ilyenkor felmerül az igény, hogy jó lenne kisebb, átláthatóbb részekre darabolni, esetleg alhálózatokra bontani. Erről korábban már esett szó az 5. gyakorlati anyagban. Ott meg is néztünk néhány érvet a több részre bontás ellen és mellett.

Ha úgy döntünk, hogy több hálózatot alakítunk ki, akkor meg kell oldanunk, hogy ezek a különálló részek kommunikálni tudjanak egymással. Az összekapcsolást a routereknek végzik a nekik megadott forgalomirányítási szabályok alapján.

Forgalomirányítás

A forgalomirányítási **szabályoknak** két fő típusa van: a statikus és a dinamikus. Az eddigi gyakorlatokon a statikussal változattal már találkoztunk, így azt most csak röviden átismétljük, majd rátérünk a valóságban gyakrabban használt dinamikus protokollokra.

Statikus forgalomirányítás

Az egyik első hálózatunk felépítése után azt figyeltük meg, hogy bár a szomszédos eszközök egyből kommunikálni tudtak egymással, ha egy másik, távolabb elhelyezkedő eszközt szerettem volna pingelni, akkor a csomag elakadt, nem jutott ki a lokális hálózatból. Erre az volt a megoldás, hogy minden routernek, megadtunk egy egyszerű szabályt, amivel tudattuk vele, hogy melyik hálózat merre van. Ilyenkor minden routernél mindegyik lehetséges hálózatra fel kellett venni egy szabályt. Ezt nevezzük statikus forgalomirányításnak.

Éles helyzetben ez viszont nem praktikus, az okokra rövid gondolkodás után magunk is rájöhettünk:

- Nagy rendszereknél sok router és sok lehetséges hálózat van, ami bonyolulttá teheti a kezdeti konfigurációt.
- Ha változik a hálózat topológiája, akkor nagy munkával jár átállítani a forgalomirányítást (ugyanis mindet ugyan így „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 nem éri meg statikus forgalomirányítást használni, így inkább automatikus módszereket részesítenek előnyben a hálózat tervezői.

A következőkben áttekintjük ezeket, illetve megnézzük, hogyan lehet a Packet Tracerben alkalmazni a megszerzett ismereteket.

Dinamikus forgalomirányítás

A fenti következtetések után nyilvánvalóvá válik, hogy valamilyen algoritmust kell alkalmaznunk, hogy egyszerűsítsük a forgalomirányítás konfigurációját. Nézzük, milyen is egy ideális forgalomirányító algoritmus:

- **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.

Forgalomirányító protokollok

A forgalomirányító algoritmusoknak két fő fajtája létezik:

- **Távolságvektor alapú** – erről lesz szó most
- **Kapcsolatállapot alapú** – erről lesz szó a következő anyagban

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 tényezők-ből származtatott (esetleg súlyozott) mértékként. Ezek a tényezők a következők lehetnek:

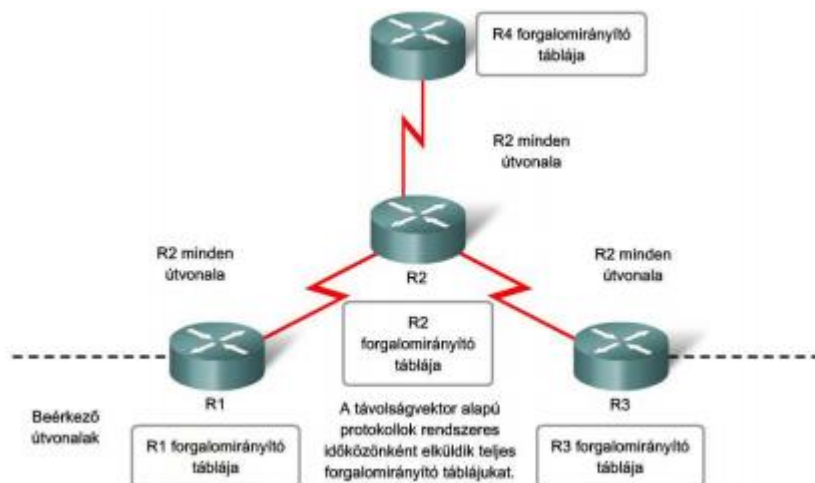
- Ugrások száma
- Adminisztratív költség
- Sáv szélesség
- Átviteli sebesség
- Késleltetések valószínűsége
- Megbízhatóság

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 találhatók, 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 információt

a szomszédok magukra vonatkoztatják (azaz megnövelik az út távolságát), ezzel jelzik, hogy innen már ennyivel nagyobb költséggel elérni az adott hálózatot. Ez-



után továbbküldik 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 továbbküldi 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 előbbieket alapján mindig **eggyel** növekszik a távolság értéke, hiszen egy ugrással több kell a célba jutáshoz. Ezek után azonnal tájékoztatja a hozzá kapcsolódó routereket is a változásokról, és így továbbgyűrűzik 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).

EIGRP (Enhanced Interior Gateway Routing Protocol)

Ez a Cisco saját fejlesztésű protokollja, amelyet többek között a RIP hibáinak kiküszöbölésére hoztak létre (pl. a maximális 15 ugrás, az ugrásszám használata mértékként). Ehelyett az EIGRP összetett mértéket használ, amely többek között a sávszélességből és a csomag adott útvonalra vonatkozó késleltetéséből adódik össze. A protokoll jellemzői:

- Egy útvonal költségének kiszámítására több mértéket használ.
- A távolságvektor alapú protokollok következő ugrás szerinti mérték tulajdonságait ötvözi további adatbázisokkal és frissítési jellemzőkkel.
- Maximum 224 ugrást engedélyez.

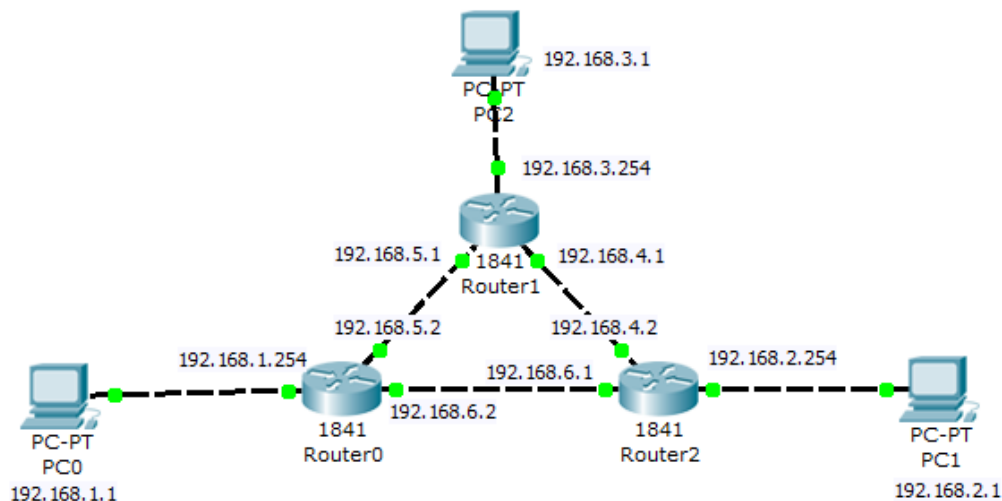
Az EIGRP nem csak a forgalomirányító táblában tárolja a működéséhez szükséges információkat, hanem használ úgynevezett szomszéd táblát és topológiát is. Az előbbiben azon router adatai (IP cím, sávszélesség, típus) találhatók, amelyek közvetlenül csatlakoznak a routerhez.

Az utóbbi a szomszédos forgalomirányítók hirdetményei alapján épül fel, és tartalmazza a szomszédok által meghirdetett összes útvonalat. E tábla segítségével gyorsan képes a topológia megváltozásakor alternatív útvonalat keresni, így gyorsabban adaptálódik, mint az RIP.

A RIP protokoll konfigurálása Packet Tracerben

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

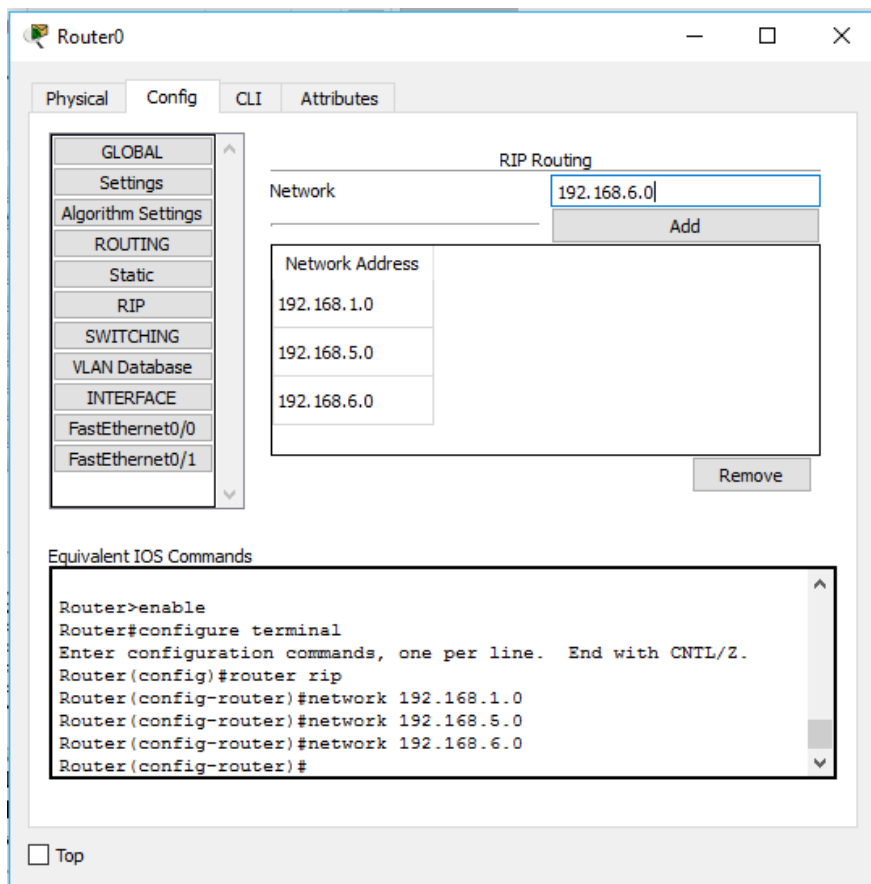
Ezt a konfigurálást parancssorból a következőképpen tudjuk elvégezni:

- 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 meg-

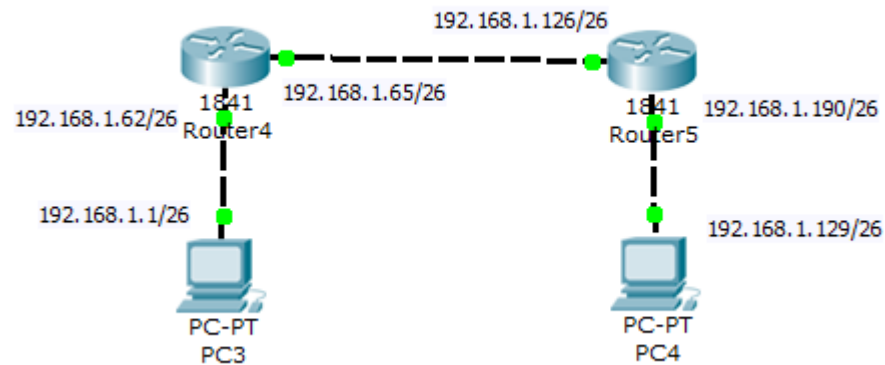


adni, hogy melyik verziót használja a router). Ez a következő ábrán látszik:

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) megadni, 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
```


Beugró kérdések

1. Melyik állítások igazak a **statikus** forgalomirányításra?
2. Melyek a forgalomirányító **algoritmusok** fő osztályai?
3. Milyen céljai lehetnek egy forgalomirányító algoritmusnak?
4. Milyen komponensek alkotják a routerek közötti távolságot?
5. Milyen **hátrányai** vannak az RIP protokollnak?
6. Milyen javításokat végeztek az EIGRP-ben az RIP-hez képest?
7. Beállítható-e a Packet Tracerben egy router grafikus konfigurációjában a használt RIP verzió?
8. Mire szolgál a `router rip` parancs?
9. Mire szolgál a `network 192.168.5.0` parancs?
10. Helyes-e a `network 192.168.10.0` parancs akkor, ha alhálózatok is vannak?

Források

1. CISCO CCNA első és második szemeszterének tananyaga
2. Andrew S. Tanenbaum: Számítógép-Hálózatok