PÁZMÁNY

Pázmány Péter Katolikus Egyetem
Információs Technológiai és Bionikai Kar

Számítógépes hálózatok
#04 – ICMP, Routing, NAT, VPN

2024. október 4.

Naszlady Márton Bese

naszlady@itk.ppke.hu

#04/1 – Internet Control Message Protocol

Internet Control Messages

Az L3 szintű csomagátvitel során számos kérdés, bonyodalom, hiba, váratlan esemény adódhat:

- Mi a router címe?
- Elérhető az adott IP című gép a hálózaton?
- Hoppá, a tűzfal visszadobta ezt az üzenetet!
- Hoppá, még nem ért célba az üzenet, de lejárt a TTL idő!
- Hamarabb elérted volna a címzettet, ha erre kerested volna…

Internet Control Messages

Ezekről a jelenségekről a node-ok
szolgálati közlemények
használatával tudnak társalogni.

Internet Control Message Protocol (ICMP)

Az internet „vezérlésére” szolgáló protokol

Bizonyos szempontból az IP fölött lévő réteg:

- IP csomagokat használ

Bizonyos szempontból az IP alatt lévő réteg:

- az IP viselkedését is befolyásolhatja

ICMP üzenet szerkezete

Az ICMP üzenet a következő egységekből áll:

type 1 byte	code 1 byte	checksum 2 byte	message

- type az üzenet elsődleges típusa
- code bizonyos üzeneteknél a típuson belüli altípust azonosítja
- checksum az egész ICMP üzenetre számolt ellenörző összeg
- message az üzenet típusától függő rész, tartalom

ICMP üzenet kategóriák

Az ICMP üzeneteknek kétféle típusa van:

ICMP hibaüzenet

Az adattovábbítás hibáját jelzi, és esetleg információt közöl arról, hogy hol történt és mi volt a hiba.

ICMP vezérlőüzenet

Kérést vagy a kérésre adott választ tartalmaz.

ICMP üzenet szabályok

Az ICMP üzenetekre nagyon szigorú szabályok vonatkoznak!

Sosem eredményezhet ICMP hibaüzenetet:

- egy másik ICMP hibaüzenet,
- IP boradcast vagy mulitcast üzenet
- alacsonyabb szintű (ethernet) boradcast vagy multicast üzenet
- egy IP csomag többedik (nem első) fragmentuma
- olyan IP csomag, aminek forráscíme nem egy létező host IP címe
- IGMP (Internet Group Management Protocol) üzenet

ICMP üzenet szabályok

Az ICMP üzenetekre nagyon szigorú szabályok vonatkoznak!

Az ICMP hibaüzenet mindig tartalmazza a kiváltó IP csomag lényeges részét:

- a teljes IP fejlécet
  ebből derül ki a küldő, címzett, típus stb.
- az IP adat első 8 byteját
  ez azért fontos, mert TCP és UDP esetén ez tartalmazza a port számát, vagyis azt, hogy melyik alkalmazást érinti a hiba

Ma is használt fontos ICMP üzenettípusok

type	code	leírás	kategória
0		Echo reply üzenet (ping válasz)	vezérlő
3	0-15	Destination Unreachable – a cél nem érhető el, a code mező részletezi az okokat	hiba
5	0-3	Redirect – átirányítás; a célállomás rövidebb úton is elérhet, ha itt keresed	vezérlő
8		Echo request üzenet (ping kérés)	vezérlő
9		Router Advertisement – a router meghirdeti magát az alhálózaton	vezérlő
10		Router Solicitation – egy eszköz kéri a routert, hogy hirdesse magát a hálózaton	vezérlő
11	0-1	Time Exceeded – lejárt a TTL mezőben lévő max. idő	hiba
12	0-2	Bad IP header – hibás az IP fejléc; a code rész jelzi, hogy mi a hiba	hiba
13		Timestamp request	vezérlő
14		Timestamp reply	vezérlő

ICMP Echo Request / Reply

Echo Request

Felkéri az adott című állomást arra, hogy az ICMP üzenetben továbbított *message* részt változtatás nélkül küldje vissza a feladónak egy ICMP Echo Reply üzenet formájában.

Echo Reply

A beérkező Echo Request üzenetre adott válaszüzenet, ami tartalmazza az Echo Request-ben kapott *message* részt.

ICMP Destination unreachable

Nagyon gyakori üzenet; nem sikerült kézbesíteni a csomagot.

A címzett eszköz, egy közbülső router vagy egy tűzfal küldheti.

A tűzfalak trükkösen is viselkedhetnek:

- lenyelhetnek csomagokat hibaüzenet nélkül
- tehetnek úgy, mintha a címzett küldené a hibaüzenetet
- tehetnek úgy, mintha hiba történt volna, pedig közben nem

ICMP Destination unreachable

A kiváltó okot a code rész tartalmazza:

type	code	leírás
3	0	**network unreachable** – egy router küldi, akkor, ha nem talál utat a címzett hálózatához
3	1	**host unreacheable** – az utolsó router küldi, akkor, ha nem találja a címzett eszközt a hálózaton
3	2	**protocol unreacheable** – akkor keletkezik, ha a csomagban megadott transzport protokolt a fogadó oldal transzport rétege nem támogatja
3	3	**port unreachable** – akkor keletkezik, ha a transzport réteg nem tudja feladni az adatot a fölötte lévő réteg számára, és erről nem is tud a saját protokolljában definiált hibaüzenetet küldeni
3	4	**fragmentation needed but don't-fragment bit set** – a csomag túl nagy, és az átvitel érdekében fragmentálni kellene, de azt nem szabad, mert be van állítva a don't fragment bit
3	5	**source route failed** – a source routing során keletkezik, ha nem lehet a csomagot továbbküldeni
3	6	**destination network unknown** – azt jelenti, hogy biztosan nem létezik a hálózat
3	7	**destination host unknown** – azt jelenti, hogy biztosan nem létezik a címzett

ICMP Destination unreachable

A kiváltó okot a code rész tartalmazza:

type	code	leírás
3	8	**source host isolated** – nem használjuk
3	9	**destination network administratively prohibited** – tűzfal által küldött kedvesebb üzenet, „meg van tiltva, hogy elérd ezt a hálózatot”
3	10	**destination host administratively prohibited** – tűzfal által küldött kedvesebb üzenet, „meg van tiltva, hogy elérd ezt az eszközt”
3	11	**network unreachable for TOS** – egy router küldi, ha nem talált a type of service beállítás szerint megfelelő útvonalat.
3	12	**host unreachable for TOS** – lásd előbb; a címzett nem támogatja a TOS beállítást
3	13	**communication administratively prohibited by filtering** – „nem beszélhetünk” kedves üzenet
3	14	**host precedence violation** – nem fontos nekünk
3	15	**precedence cutoff in effect** – nem fontos nekünk

ICMP Redirect

A célállomás rövidebb úton (is) elérhető, ha erre keresed...

A router küldi vissza a feladónak, ha tud róla, hogy egy másik router kedvezőbb utat biztosítana. Csak akkor van értelme ilyet küldeni, ha a feladó és a másik router is egy hálózaton van.

Az üzenet tartalmazza a másik (jobb) router IP címét.

Az eredeti feladó ennek fényében módosítja a saját routing tábláját (mit-hova).

Veszélyes! Gonosz üzenetekkel el lehet téríteni a csomagokat.

Nem szabad mindenkitől elfogadni (csak az alapértelmezett átjárótól (vagy talán még attól sem)).

ICMP Time exceeded

Eldobtam a csomagot, mert mire ideért, lejárt a TTL érték.

Az a router (node) küldi vissza, amelyiknek még tovább kéne küldenie a csomagot, de nem tudja tovább csökkenteni a TTL-t (0 lenne).

A visszaküldött ICMP üzenetből látszik, hogy melyik helyen járt le az idő, és hogy melyik csomag volt az érintett.

Nem minden router olyan rendes, hogy szóljon erről az eseményről. :'(

#04/1 – Összefoglalás

Protokoll	ICMP rendeltetése
	Gyakran használt ICMP üzenettípusok
Elvek	ICMP üzenet szabályok

#04/2 – ICMP alapú alkalmazások

A `ping` program

Klasszikus eszköz egy IP cím elérhetőségének vizsgálatára.

A küldő ICMP Echo Request üzenetet küld, a címzett (ha erre képessége és szándéka van) ICMP Echo Reply üzenettel válaszol.

Használatával meghatározható az RTT és MTU is.

A program paraméterezhető:

hány csomagot, milyen gyakran, milyen tartalommal, mekkora méretben stb.

A hálózati eszközök szűrhetik az ICMP Echo üzeneteket (befele és kifele is).

RTT

RTT – Round Trip Time

A fizikai médiumban a jelterjedés sebessége korlátos. Az adatátvitel során a kódolás-dekódolás szintén időbe telik.

Az RTT kifejezi, hogy mennyi idő telt el a kérés elküldése és a válasz beérkezése közt.

Mértékegysége másodperc, értéke jellemzően milliszekundum nagyságrendű.

A felsőbb rétegek számára fontos tudni a felhasználói élmény javítása érdekében.

Az RTT a teljes oda-vissza idő (nem mutatja külön az oda- és visszaút idejét).

MTU

MTU – Maximum Transmission Unit

A fizikai médium rendszerint korlátozza az átvihető frame-ek méretét.

Az MTU kifejezi, hogy mekkora az a legnagyobb méretű frame, ami még egyben átvihető a linken.

Mértékegysége byte.

A felsőbb rétegeken célszerű az MTU-nál nem nagyobb csomagokat használni.

A gyakorlatban nem tudjuk, hogy a küldő és a cél közt milyen MTU értékű közegeken megy keresztül a csomag; elvben „bármi” is lehet az MTU érték az egyes szegmensekben.

Ha túl nagy a csomag, akkor vagy daraboljuk, vagy hibát jelzünk.

Ethernet esetében 1500 byte a szokásos MTU méret.

Path MTU discovery

Trükk az útvonalon lévő legkisebb MTU megtalálására.

Lehetőleg minél nagyobb csomagokat akarunk küldeni, és nem akarjuk, hogy közben fragmentumokra szedjék szét azokat. Honnan tudjuk, hogy mi a legkisebb MTU?

Algoritmus:

1. Ismerjük a saját hálózatunkon az MTU-nkat.
2. Készítünk egy változót (`path_MTU`), ami az útvonal MTU-ja lesz. Ez kezdetben legyen annyi, mint a mi hálózatunk MTU-ja.
3. `path_MTU` méretű csomagot küldünk a célnak, amiben be van állítva a DF bit.
4. Ha „túl nagy csomag” hibaüzenetet kapunk vissza, akkor csökkentjük a `path_MTU` értékét, és újra próbálkozunk a 3. lépéstől.
5. Egyszer csak már nem kapunk vissza hibát. Ekkor megtaláltunk a path MTU értékét.

A `traceroute` program

Tudni szeretnénk, hogy egy adott címre küldött csomagokat általában merre továbbítják a routerek a nagyméretű hálózatban.

A mérés alapelve az ICMP Time Exceeded hibaüzeneten alapszik.

A `traceroute` program 1, 2, 3, ... TTL értékkel küld csomagokat a címzett felé.

A közbülső első, második, harmadik, ... eszközök szépen eldobálják a csomagot, és erről (általában) ICMP Time Exceeded üzenetet küldenek. Ebben benne van a saját címük.

A beérkezett ICMP üzenetekből kitalálható, hogy az egyes routerek merre továbbították a csomagot, és az is, hogy hány hop után értünk célba.

Router advertisement, Router solicitation

Az eszközök csak egy router segítségével tudnak IP csomagot küldeni a nem saját alhálózatukba tartozó eszköznek.

Másik hálózatba tartozó adat küldése előtt találni kell egy routert.

Lehetőségek:

- manuálisan konfigurált router címek
- boot időben letöltött konfiguráció
- router advertisement, router solicitation

Router advertisement, Router solicitation

Működési elve:

A *routerek* időről-időre *router advertisement* üzenetet küldenek:

- multicast üzenet az *„all hostst”* címre
- a router elküldi a saját interfésze IP címét
- a nodeok figyelik a beérkező advertisementeket, és ebből tanulnak

A *hostok* (a többi eszköz):

Ha a hálózathoz frissen csatlakozik egy eszköz, akkor *router solicitation* üzenetet küld:

- multicast üzenet az *„all routers”* címre
- kéri a routereket, hogy küldjenek most azonnal egy advertisement üzenetet

Ezek az ICMP üzenetek csak felderítésre szolgálnak, nem derül ki belőlük, hogy melyik a „jobb” router. (Ebben az ICMP redirect üzenetek segítenek.)

#04/2 – Összefoglalás

Eljárások

ping

path MTU discovery

traceroute

router solicitation, advertisement

#04/3 – Routing feladat

IP routing

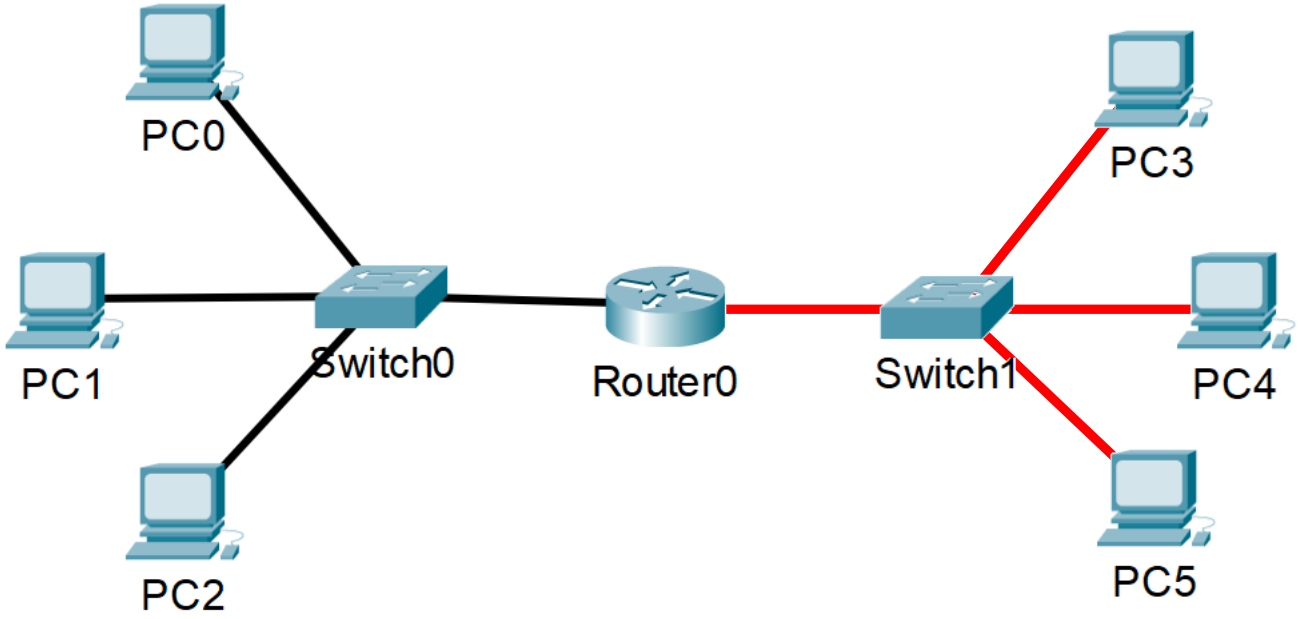
Az IP szintű forgalomirányítás feladata
egy dinamikusan változó,
irányított, súlyozott gráfban
minimális költségű utat keresni.

A feladatnak rengetegféle megközelítése és kiterjedt irodalma van.

Network layer szinten működő eszközök (ismétlés)

Router (gateway, átjáró, útválasztó)

- Legalább két IP címmel rendelkezik
- A beérkező IP csomagokat megpróbálja a címzett alhálózatába továbbítani
- Figyeli a címzést, esetleg egyes IP címtartományokból jövő vagy oda címzett üzeneteket nem visz át.



Routing tábla (ismétlés)

Az eszközöknek tudnia kell,
hogy melyik CIRD blokkba szánt csomagot hova kell küldeni.

Erre szolgál a ***routing tábla***:

- Minden, alhálózatok közt forgalmazó eszközben jelen van
  - Nem csak a routerben, hanem pl. egy PC-ben, okostelefonban is van.
  - A switch, bridge csak L2 szinten, alhálózaton belül forgalmaz, ezekben nincs routing tábla.
- Megadja, hogy melyik CIDR blokkot melyik interfészen melyik IP-re továbbítsuk
  - Egyszerűbb alhálózatban a feladat általában megoldható egy darab default route-tal.
  - Bonyolultabb hálózatban több irányba is küldhető a csomag, itt az út hossza is számíthat.

Routing tábla (ismétlés)

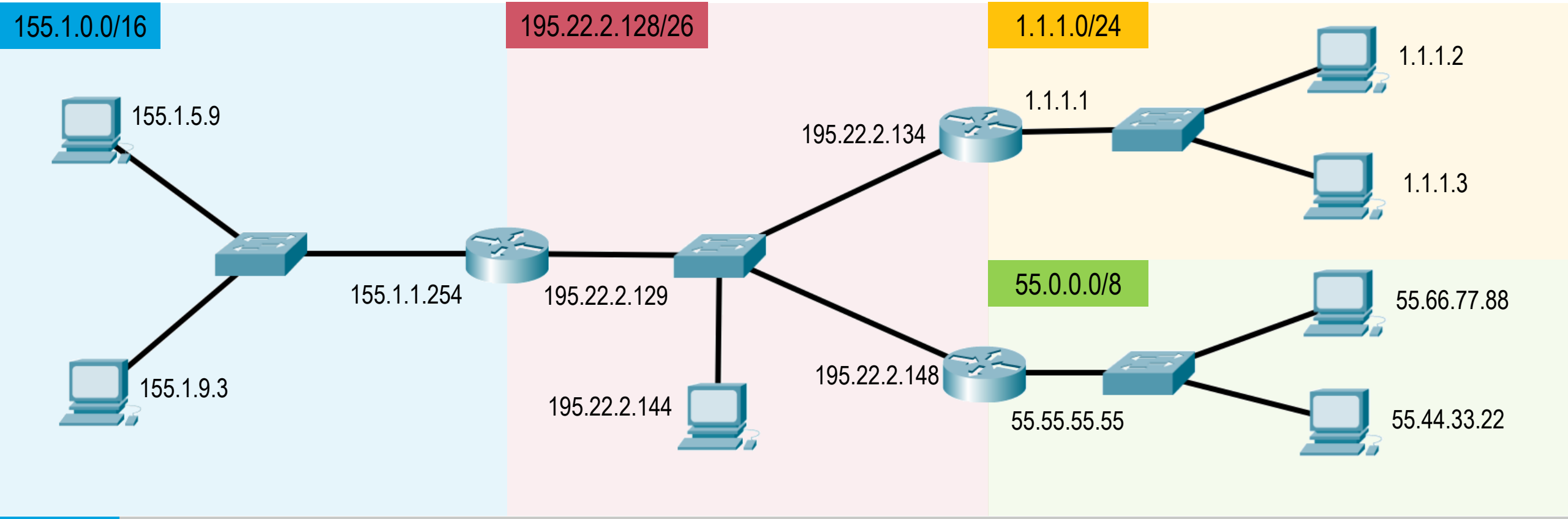
A routing tábla legalább 3 információt tartalmaz:

- **network identifier**
  Az elérni kívánt alhálózat IP címmel és netmaskkal vagy CIDR módon megadva
- **next hop**
  Annak az interface-nek az IP címe, ahova a csomagot továbbítani kell
- **metric**
  Valamilyen metrika vagy prioritás, ami az útvonal költségét jelzi

network	next hop	metric
155.1.0.0/16	195.22.2.129	1
195.22.2.128/26	-	0
1.1.1.0/24	195.22.2.134	1
55.0.0.0/8	195.22.2.144	1

Routing tábla példák (ismétlés)

Tekintsük az alábbi, négy alhálózatból (subnet) álló hálózatot.
Néhány példa az eszközökben lévő routing táblára:

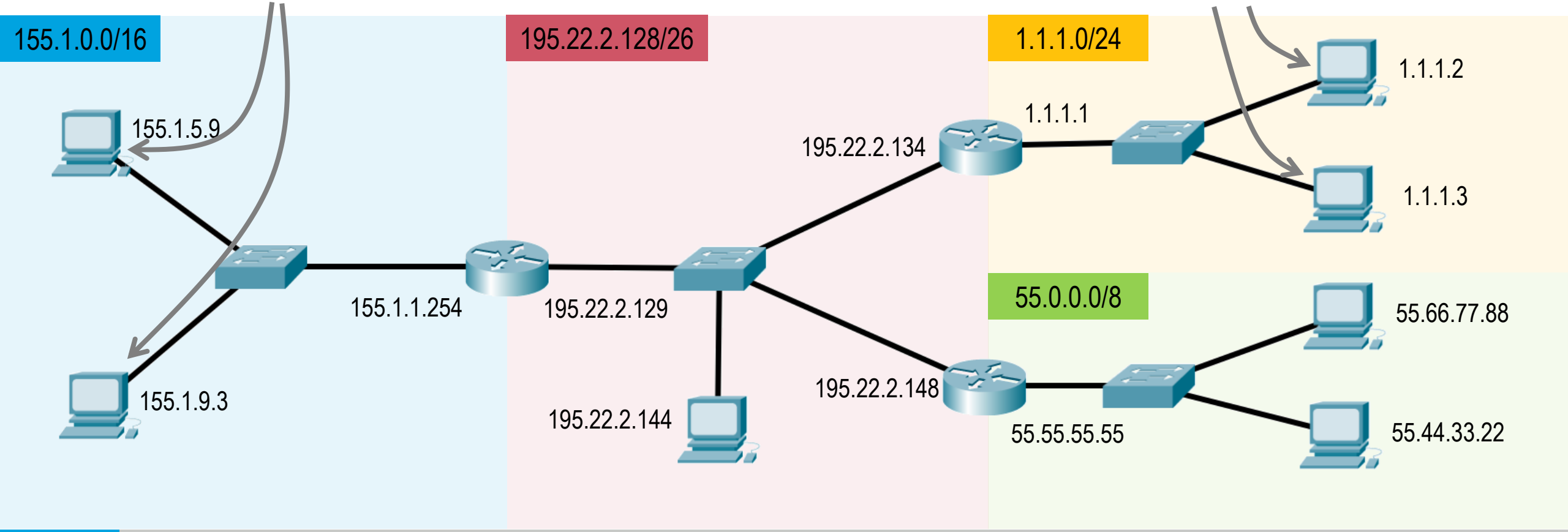


Routing tábla példák (ismétlés)

Egyes eszközökben csak egy „*default route*” található meg.
Minden, nem a saját alhálózatba címzett csomagot erre a címre továbbít.

network	next hop	metric
0.0.0.0/0	195.1.1.254	?
155.1.0.0/16	-	0

network	next hop	metric
0.0.0.0/0	1.1.1.1	?
1.1.1.0/24	-	0

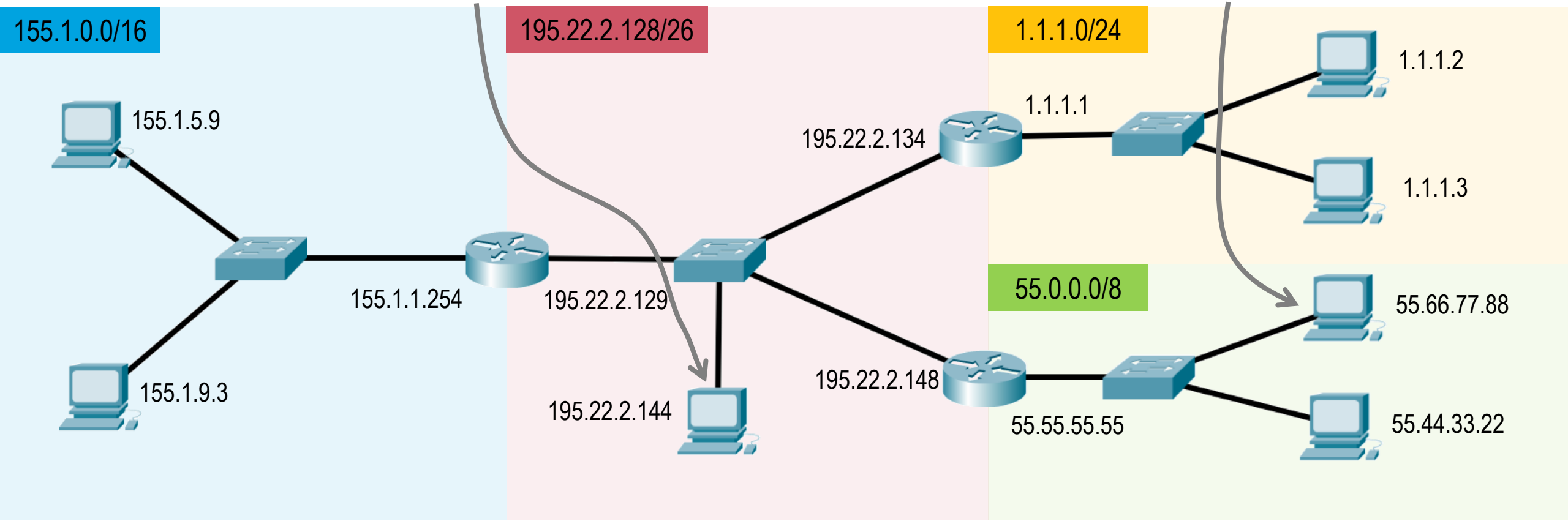


Routing tábla példák (ismétlés)

És természetesen az is jó megoldás, ha a hálózat pontos leírása van a táblában:

network	next hop	metric
195.22.2.128/26	-	0
55.0.0.0/8	195.22.2.148	1
1.1.1.0/24	195.22.2.134	1
155.1.0.0/16	195.22.2.129	1

network	next hop	metric
55.0.0.0/8	-	0
195.22.2.128/26	55.55.55.55	1
1.1.1.0/24	55.55.55.55	2
155.1.0.0/16	55.55.55.55	2



Internet routing

Nagyon nagy kiterjedésű hálózat esetén a routing feladat nem egyszerű!

Azért, mert:

- minden router nem tudhat minden más router létezéséről,
- az összeköttetések (súlyai) folyamatosan változnak,
- a hálózat egyes részeit kívülről „fekete doboznak” akarjuk láttatni,
- kifejezetten meg akarjuk határozni, hogy kik felé és milyen üzenetet akarunk átadni,
- …

Internet routing

Az internet gerincét (**backbone**) alkotó hálózatok közt is **routerek** teremtenek kapcsolatot, továbbítják a csomagokat, irányítják a forgalmat.

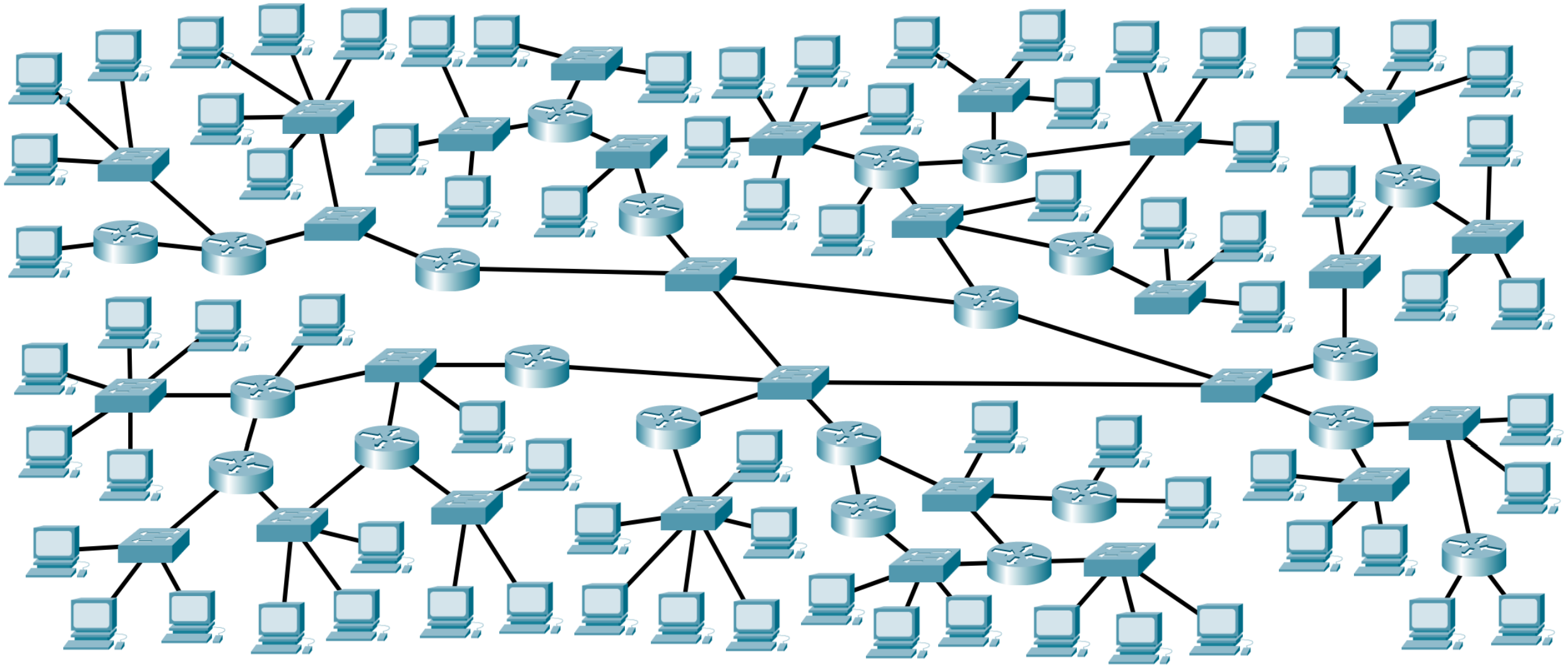
(A *backbone router* másik elnevezése *internet gateway*, gyakran ezen a néven említi a szakirodalom.)

A **backbone routerek** kettős feladatot látnak el:

- **forwarding:** a valahonnan beérkező üzenet továbbítása valamely más eszköz számára a routing tábla alapján
- **routing:** a routing tábla karbantartása (a fogadott üzenetek alapján) és hirdetése (általa küldött üzenetek révén) a hálózatban

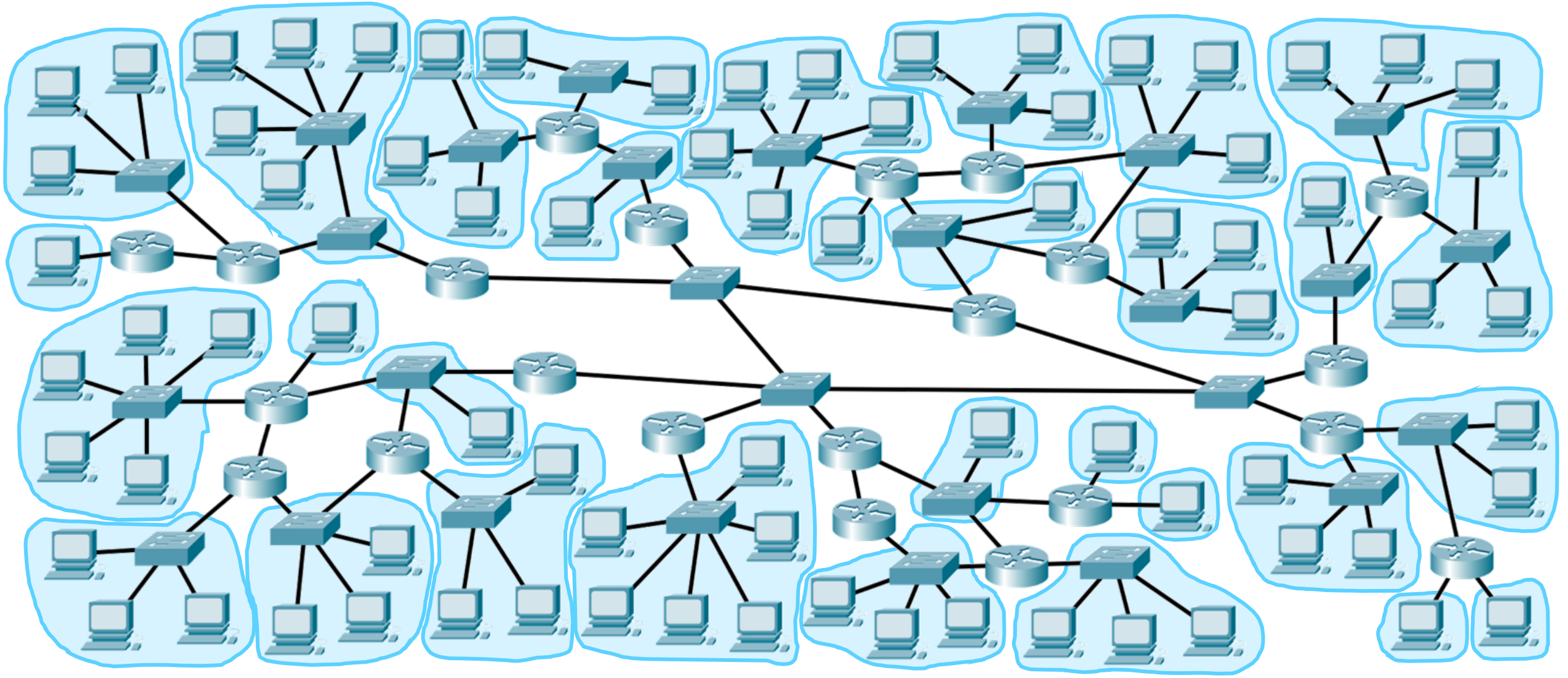
Az internet hierarchiájának szemléltetése

Szövevényes hálózat, amit routerek, switchek és végponti eszközök építenek fel.



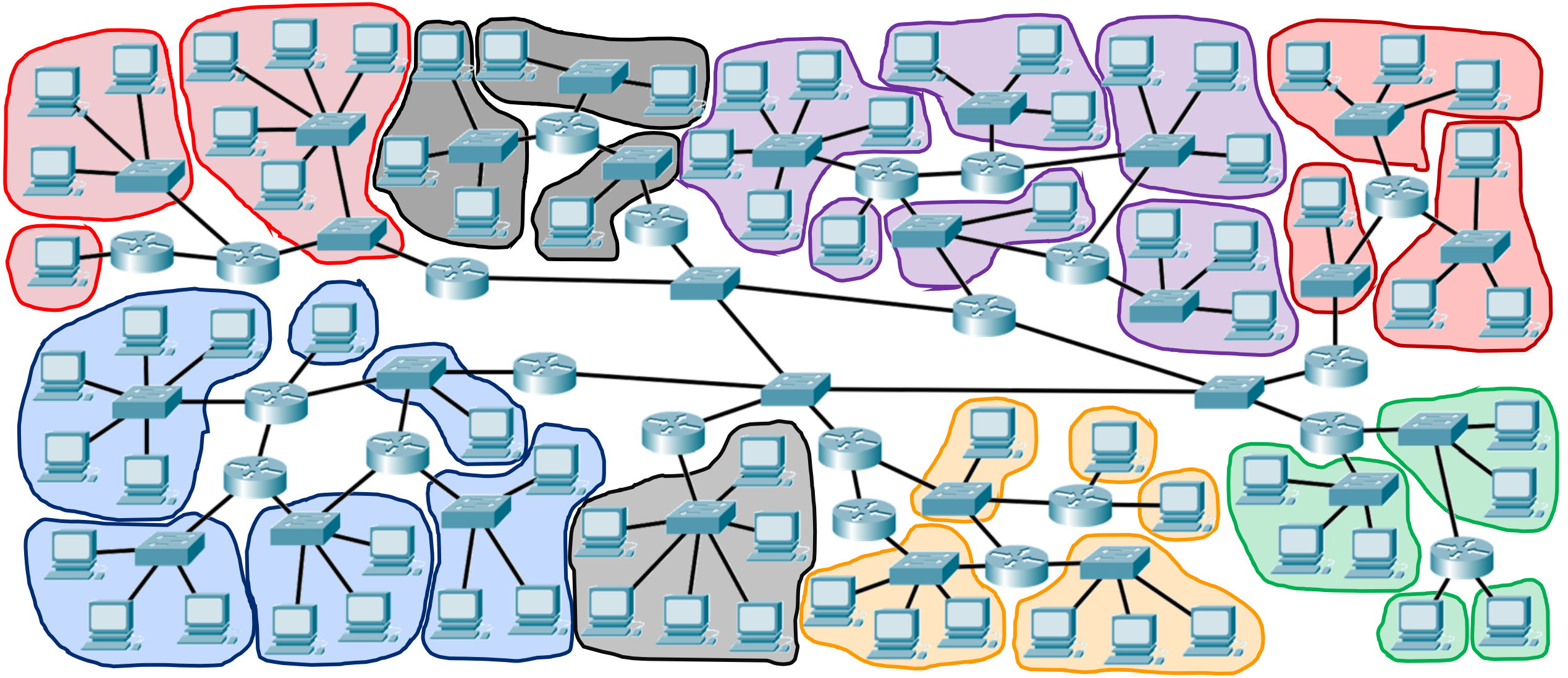
Az internet hierarchiájának szemléltetése

A gráfot a routerek mentén különálló rész-gráfokra tudjuk darabolni:



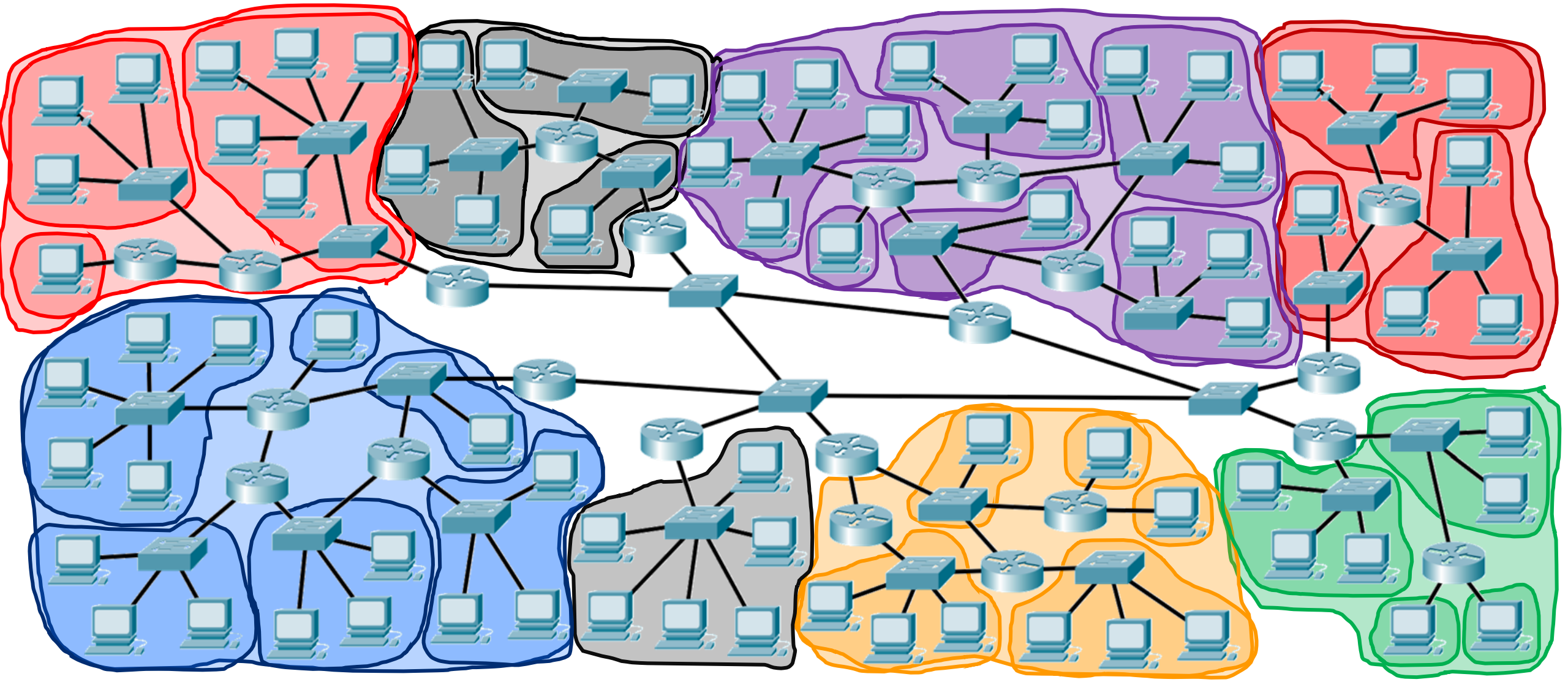
Az internet hierarchiájának szemléltetése

A részgráfokat egy-egy szervezet (telephely, cég, iskola, …) felügyeli



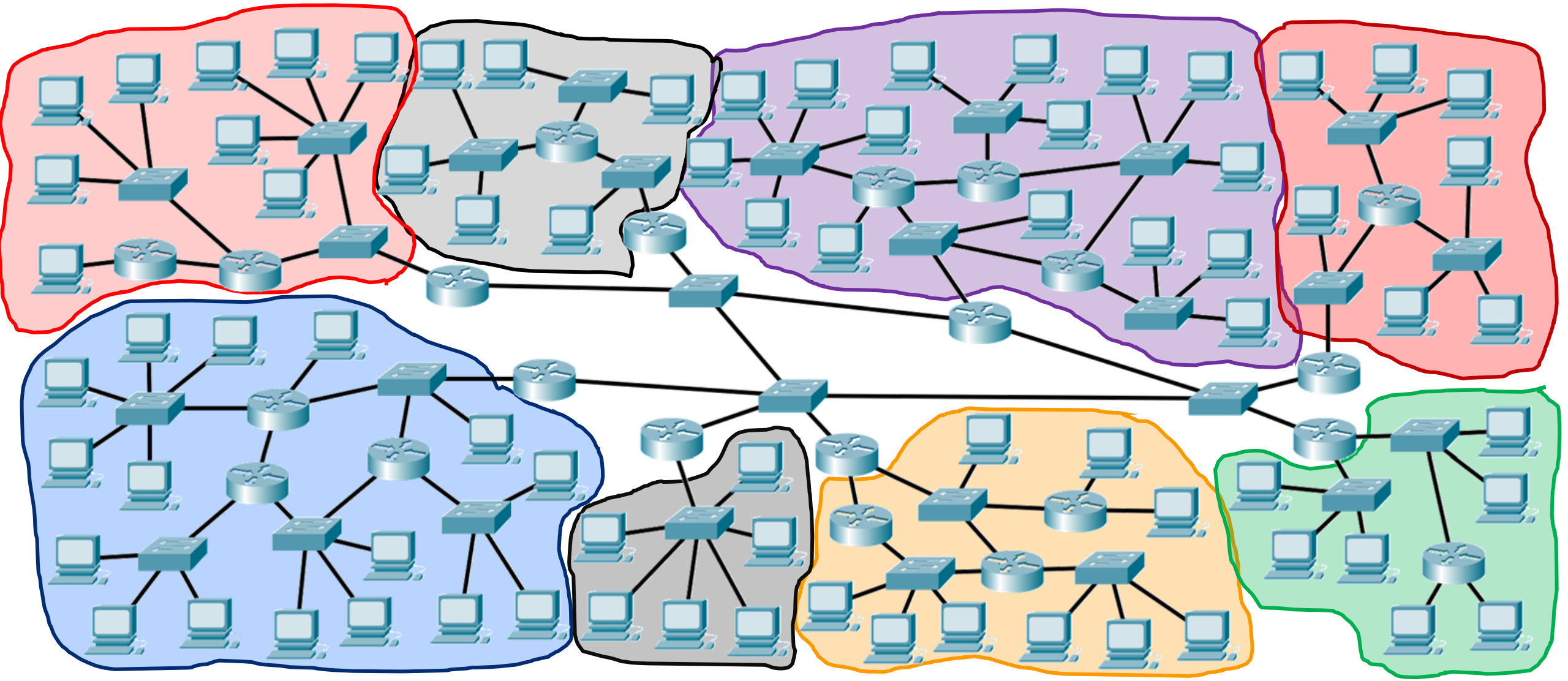
Az internet hierarchiájának szemléltetése

A szervezeteket is nagyobb, közös szervezetek kezelik egyben (pl. országok, ISP-k).



Az internet hierarchiájának szemléltetése

A forgalomirányítás feladata szétválik szervezeteken belüli és közötti routing feladatra.

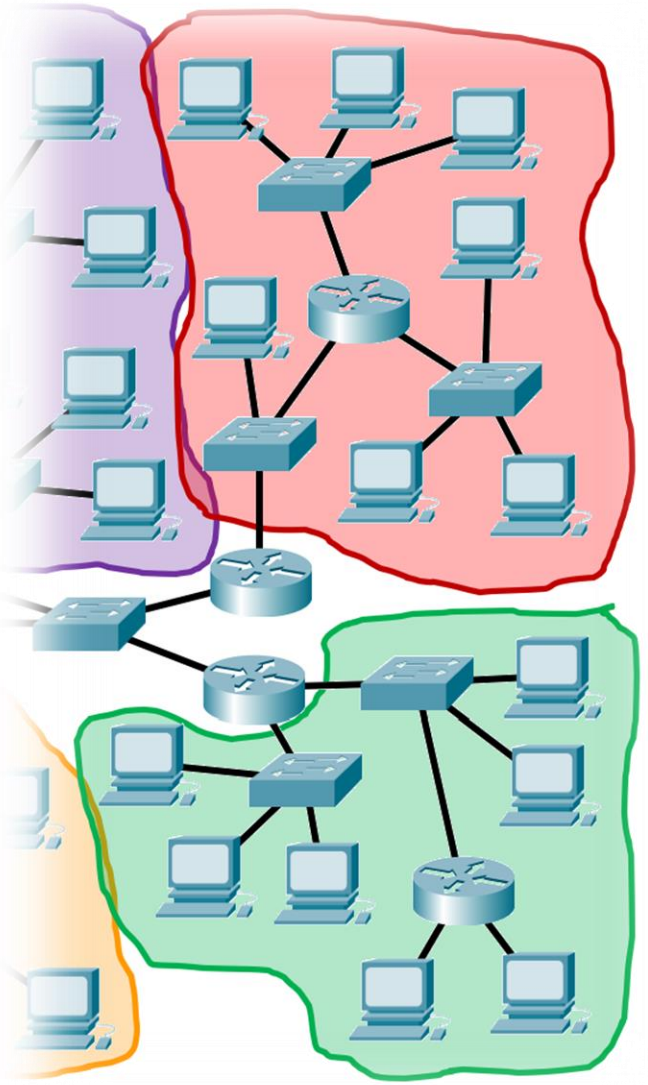


Az internet felépítésének leírásánál fontos fogalmak

- **Autonomous System (AS)** – önálló rendszer, ami egy adminisztratív hatóság (cég) felügyelete alatt áll.

- **Interior Gateway Protocol (IGP)** – az azonos AS-ben lévő routerek egymás közötti információcseréjét lehetővé tevő protokoll.

- **Exterior Gateway Protocol (EGP)** – az AS-ek határán lévő routerek ennek használatával beszélgetnek a más AS-ek határain lévő routerekkel.



#04/3 – Összefoglalás

Fogalmak

Routing feladat

Internet felépítés

AS, IGP, EGP

#04/4 – AS

Autonomous System (AS)

Az AS egy routing szempontjából önálló entitás, pl egy-egy szolgáltató által egyben felügyelt hálózat (pl. a Telekom vagy a Vodafone vagy a KIFÜ hálózata...).

A szolgáltatóhoz egy számot (AS number) rendelünk.

Példák: AS1955 (KIFÜ), AS20845 (DIGI), AS15169 (Google), AS5483 (Telekom)

Az AS szám **világállandó**, európán belül a RIPE osztja.

A szolgáltatók CIDR blokkok egy halmazát kapják meg, az AS ezeket a CIDR blokkokat tartalmazza, és ezeket (plusz a tanultakat) hirdeti.

Autonomous System (AS)

Például: a KIFÜ (AS1955) az alábbi CIDR blokkokat kapta meg:

```
193.224.0.0/15
195.199.0.0/16
195.111.0.0/16
192.146.134.0/23
192.188.244.0/22
193.6.0.0/16
146.110.0.0/16
2001:738::/32
192.190.173.0/24
192.160.172.0/24
192.188.242.0/23
```

Ebből egyben tudja hirdetni a teljes HBONE hálózatot (193.224.0.0/15)
vagy az ITK saját blokkját (193.225.109.0/24)

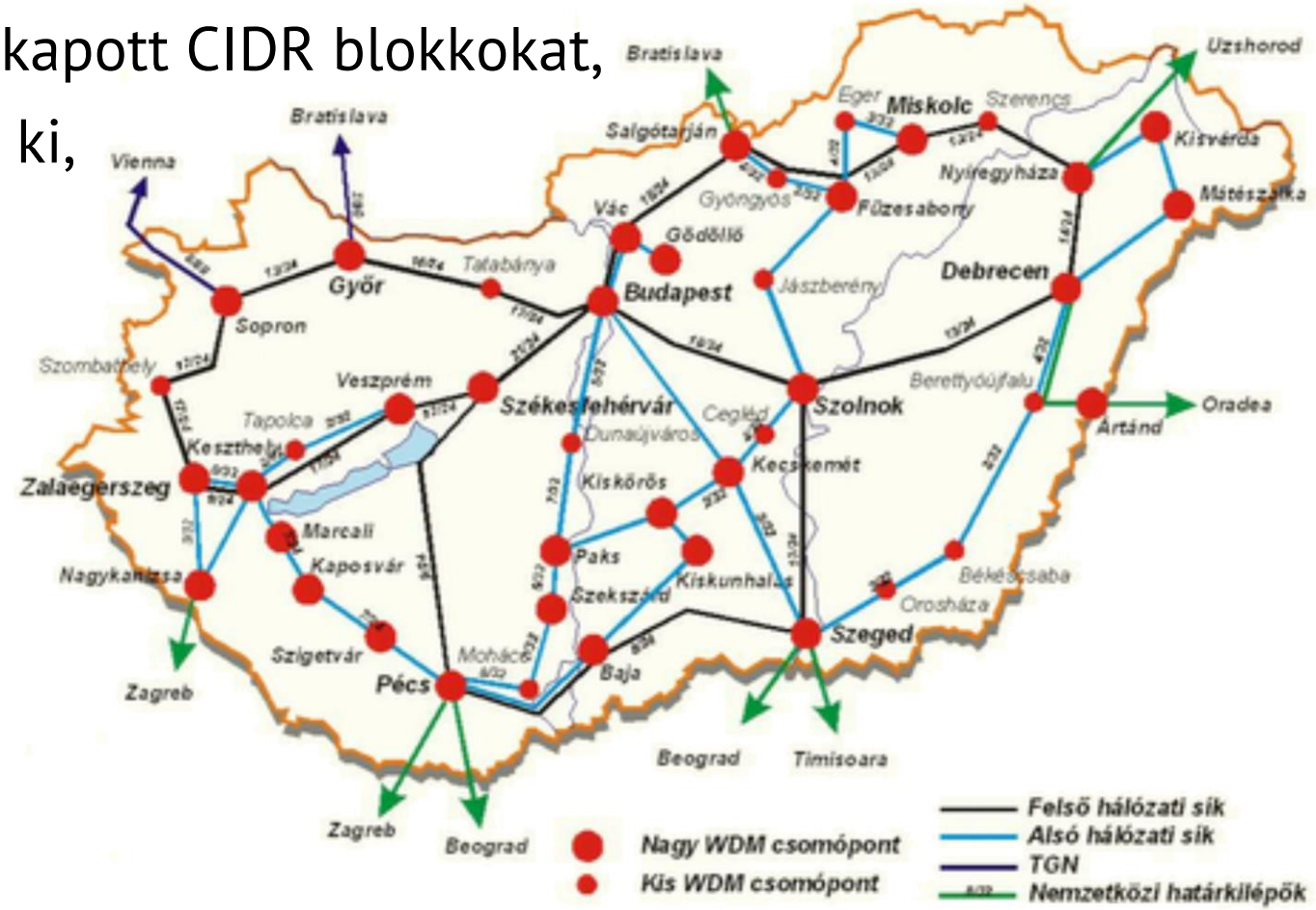
https://hackertarget.com/as-ip-lookup/

Route-olás az AS-en belül

Az **AS-en belül** az ebbe tartozó IP-k esetében a route-olást a szolgáltató végzi.

Az adott szolgáltató saját maga határozza meg, hogy…

- milyen további alhálózatokra osztja a kapott CIDR blokkokat,
- milyen belső hálózati struktúrát alakít ki,
- hány és milyen routert használ
- stb.

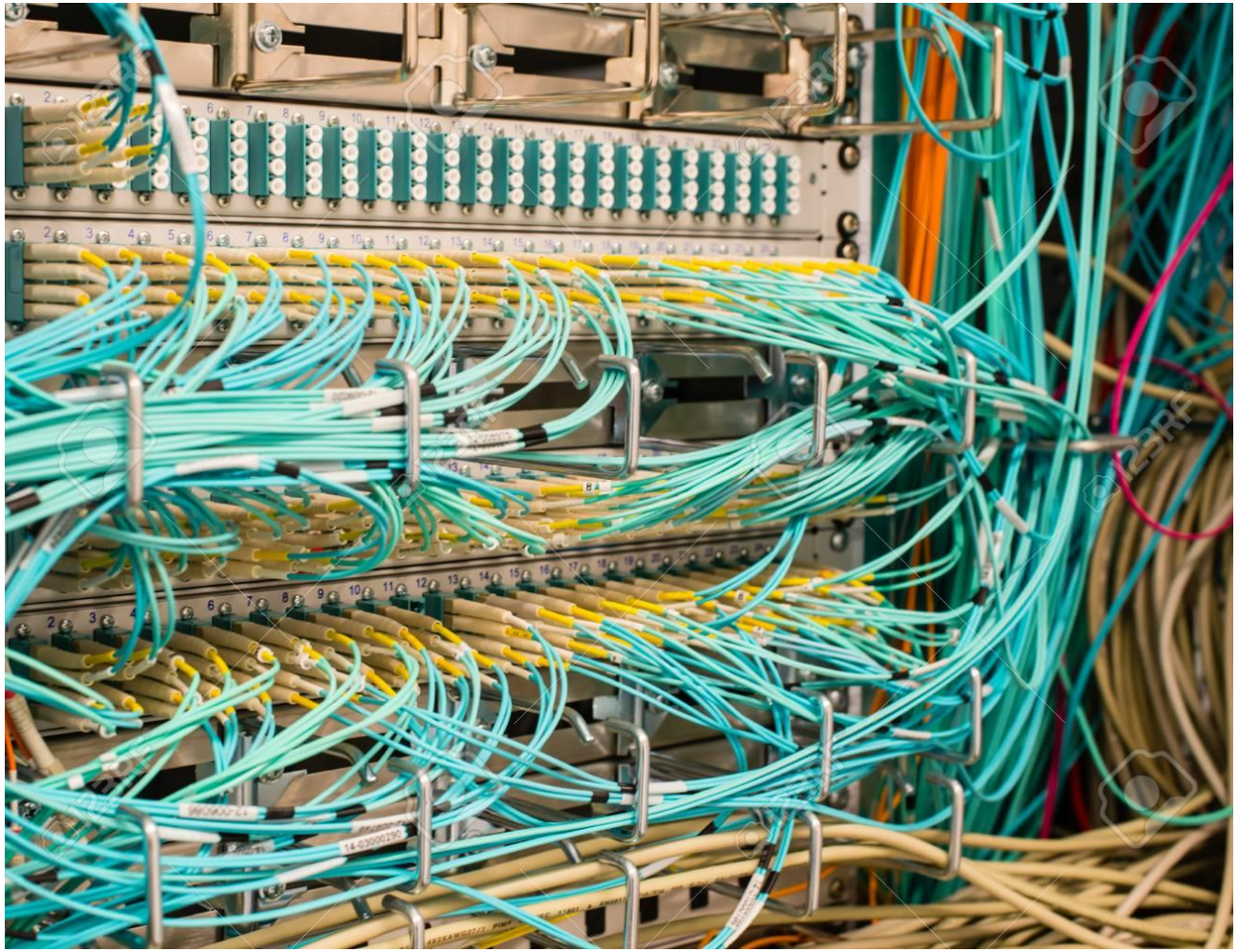


Routeolás az AS-ek között

Az **AS-ek közötti routeolás** az AS-ek határán lévő routerek és az internet backbone routerek feladata.

Az AS-ek közt egyeztetésre szorul:

- melyik AS mely CIDR blokkokkal bír,
- mely AS-eken keresztül esetleg mely másik AS-ek érhetők el,
- többféle útvonal közül melyik az optimális
- stb.



Forgalomkicserélés

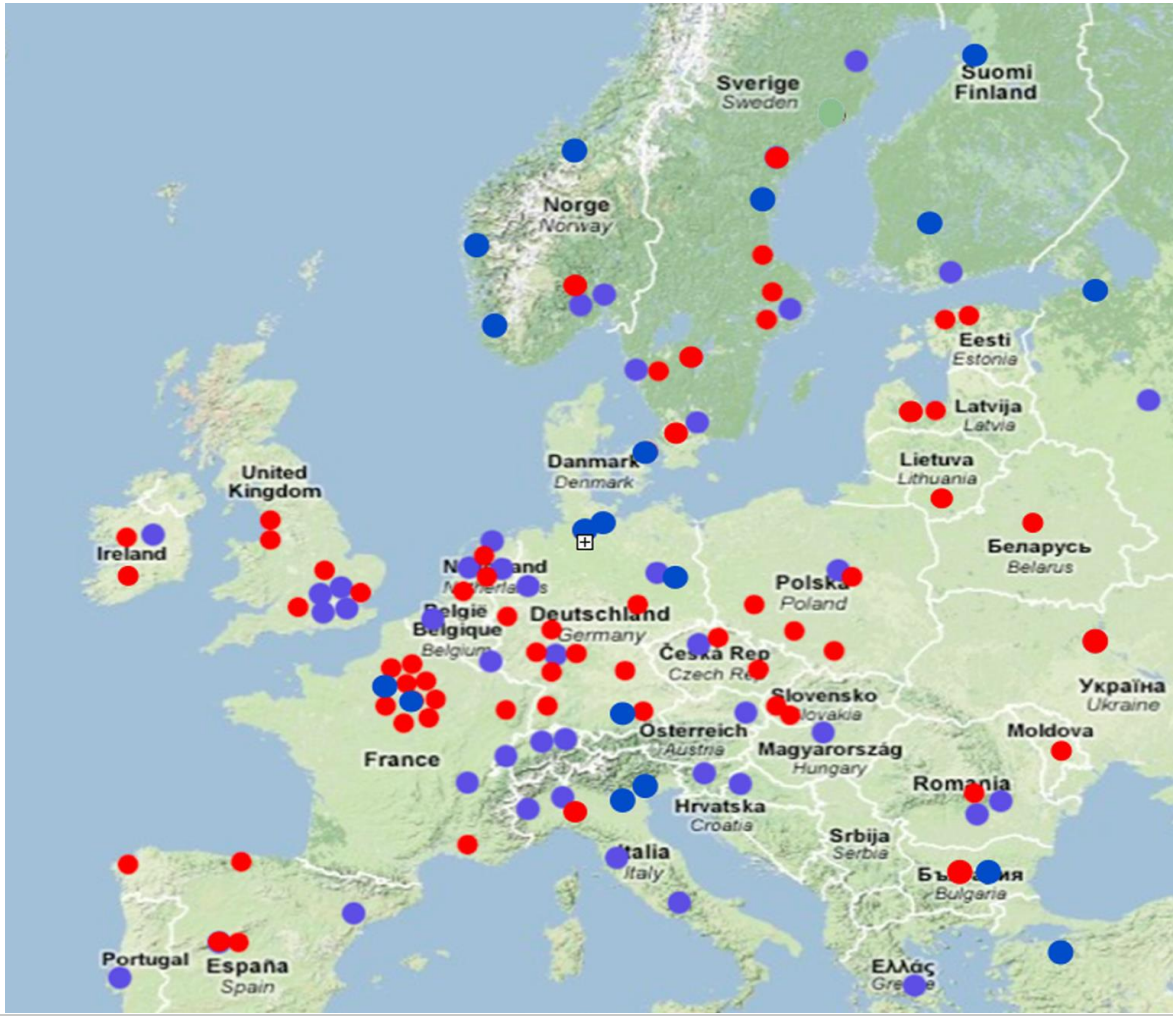
Célszerű, hogy a különböző szolgáltatók hálózatai ne csak egy-egy helyen találkozzanak egymással, vagyis a forgalom kicserélése lokális központokban is végbemehessen.

Internet Exchange Point (IXP)

Olyan hely (szervezet) ahol az ISP-k routerei forgalmat cserélhetnek ki egymás közt.

Az ISZT által működtetetett magyarországi forgalomkicserélő központ a
Budapest Internet Exchange (BIX)

https://www.bix.hu/



WHOIS

Egyszerű, emberi fogyasztásra közvetlenül alkalmas információt szolgáltat a hálózati alanyokról.

Egy ilyen alanyt **objektum**nak hívunk, ami lehet hálózat, AS vagy domain.

Egy objektum tulajdonságát **attribútum**nak hívjuk. Ilyen tulajdonságok például:

- tulajdonos neve
- adminisztratív kapcsolattartó neve, elérhetősége
- technikai kapcsolattartó neve, elérhetősége
- routing paraméterek (AS-nél)
- …

https://dnschecker.org/asn-whois-lookup.php

#04/4 – Összefoglalás

Fogalmak

AS

WHOIS

Forgalomkicserélés

#04/5 – Routing protokollok

Routing protokollok

A routing protokollok célja, hogy a routerek beszélgetni tudjanak egymással, és ki tudják simítani azokat a hibákat, amik miatt nem lenne működőképes a hálózat.

A routerek információkat küldenek magukról a többieknek, és figyelembe veszik a többiektől kapott információkat. Ez a beszélgetés kétféle stílusban történhet:

Link state protokollok
Globális információkra támaszkodik, a teljes hálózat ismeretében dönt.

Distance vector protokollok
Lokális információkra támaszkodik, a szomszédokkal beszélget.

Link state protokollok

A link state protokollok GLOBÁLISAN közelítik meg a feladatot:
„mindenki beszélget mindenkivel”
„mindenki első kézből tud mindent”

Minden csomópont elküldi a **saját kapcsolatait** az **egész hálózatba.**

Fontos, hogy mindenhova eljusson minden hirdetés.

A csomópontok egy idő után minden csomópont kapcsolatait tudni fogják, és **önállóan építik fel a teljes routing táblát.**

Link state protokollok

Előnyök:

- Gyors konvergencia – mindenki mindig a valós helyzetet mondja
- Térkép a hálózatról – az összes csomópont össze kapcsolata ismeretében a teljes hálózat térképét ismeri minden eszköz; tudnak a redundáns utakról is.

Hátrányok:

- Mindenhova el kell juttatni minden üzenetet – sávszélesség-igény
- Memóriagényes – tárolni kell a teljes hálózati struktúrát
- Processzorigényes – számolni kell a teljes hálózati struktúrában

Distance vector protokollok

A distance vector protokollok LOKÁLISAN közelítik meg a feladatot:
„mindenki csak a szomszédjával beszélget”
„a szomszédom mondta, hogy az ő szomszédja így meg úgy”

Minden csomópont az összes szomszédjának elküldi a **saját routing tábláját.**

A routerek összevetik a jelenlegi táblájukat a szomszédoktól kapott hirdetésekkel: amit egy szomszéd router k távolságra lát, azt én $k + 1$ távolságra látom.

Ha nem volt eddig jobb utam, akkor **e felé a szomszéd felé route-olok,** ha viszont tudok jobbat, akkor figyelmen kívül hagyom a hirdetést.

Ha stabil a hálózat állapota, akkor egy idő után a routing táblák is **konvergálnak.**

Distance vector protokollok

A hirdetésben küldött táblában venne van:

- a célpont CIDR blokk
- és távolság (distance), hogy milyen messze látom (pl. hop count);
- valamint kiderül belőle a szomszéd router címe

A kapott hirdetéseket úgy veszem figyelembe, hogy:

- a fogadott a hop count értékekhez hozzáadok 1-et,
- megnézem, melyik CIDR blokkra vonatkozik,
- megnézem, mi a kapott router cím ehhez a blokkhoz,
- megnézem, mi az általam ismert router cím ehhez a blokkhoz.

    Ha a két cím egyezik: eltárolom az új hop értéket.

    Ha a két cím különbözik: a kisebb hop érték felé fogok route-olni.

Distance vector protokollok

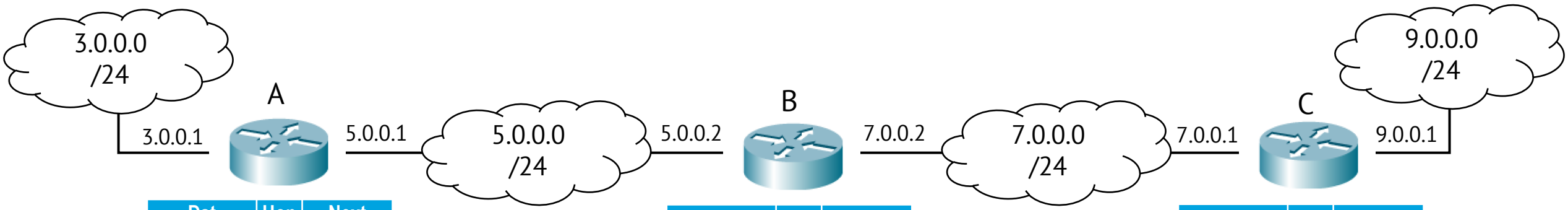
Előnyök:

- Könnyen implementálható – kis hálózatban emberi ésszel is követhető
- Csak „helyben” küld adatot – kisebb terhelés a hálózatnak

Hátrányok:

- A megszakadt kapcsolatot nehezen veszi észre – counting to infinity
- A konvergenica csak nehezebben alakul ki – iterációs folyamat, a link state esetében mindenki kapásból tudott mindent.

Counting to infinity



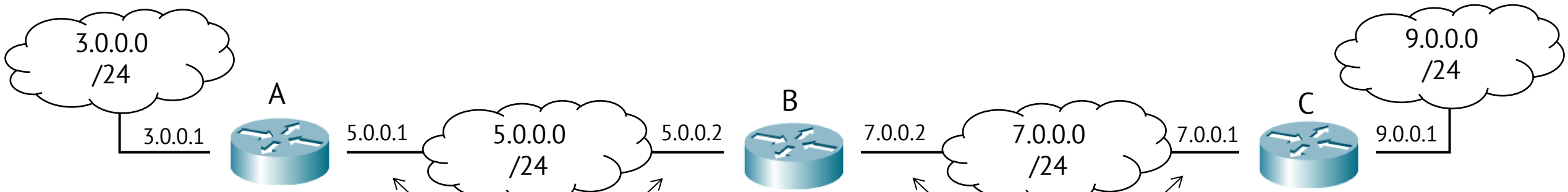
Dst	Hop	Next
3.0.0.0/24	1	-
5.0.0.0/24	1	-
7.0.0.0/24	2	5.0.0.2
9.0.0.0/24	3	5.0.0.2

Dst	Hop	Next
3.0.0.0/24	2	5.0.0.1
5.0.0.0/24	1	-
7.0.0.0/24	1	-
9.0.0.0/24	2	7.0.0.1

Dst	Hop	Next
3.0.0.0/24	3	7.0.0.2
5.0.0.0/24	2	7.0.0.2
7.0.0.0/24	1	-
9.0.0.0/24	1	-

Adott egy szép kis hálózat...

Counting to infinity



Dst	Hop	Next
3.0.0.0/24	1	-
5.0.0.0/24	1	-
7.0.0.0/24	2	5.0.0.2
9.0.0.0/24	3	5.0.0.2

Dst	Hop	Next
3.0.0.0/24	2	5.0.0.1
5.0.0.0/24	1	-
7.0.0.0/24	1	-
9.0.0.0/24	2	7.0.0.1

Dst	Hop	Next
3.0.0.0/24	3	7.0.0.2
5.0.0.0/24	2	7.0.0.2
7.0.0.0/24	1	-
9.0.0.0/24	1	-

B (5.0.0.2)

Dst	Hop
3.0.0.0/24	2
5.0.0.0/24	1
7.0.0.0/24	1
9.0.0.0/24	2

A (5.0.0.1)

Dst	Hop
3.0.0.0/24	1
5.0.0.0/24	1
7.0.0.0/24	2
9.0.0.0/24	3

C (7.0.0.1)

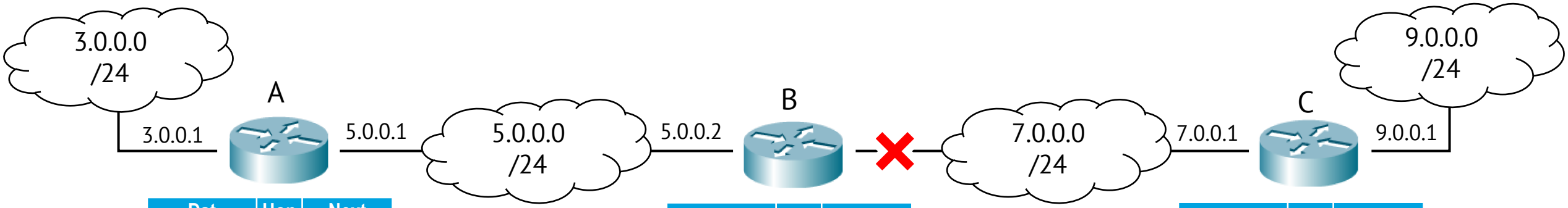
Dst	Hop
3.0.0.0/24	3
5.0.0.0/24	2
7.0.0.0/24	1
9.0.0.0/24	1

B (7.0.0.2)

Dst	Hop
3.0.0.0/24	2
5.0.0.0/24	1
7.0.0.0/24	1
9.0.0.0/24	2

Normál esetben mindenki meghirdeti a szomszédja felé a saját routing tábláját, és a routerek a kapott táblákból számolnak.

Counting to infinity



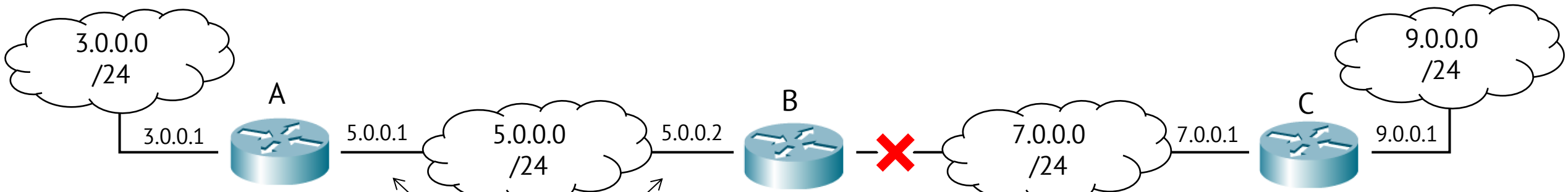
Dst	Hop	Next
3.0.0.0/24	1	-
5.0.0.0/24	1	-
7.0.0.0/24	2	5.0.0.2
9.0.0.0/24	3	5.0.0.2

Dst	Hop	Next
3.0.0.0/24	2	5.0.0.1
5.0.0.0/24	1	-
7.0.0.0/24	1	-
9.0.0.0/24	2	7.0.0.1

Dst	Hop	Next
3.0.0.0/24	3	7.0.0.2
5.0.0.0/24	2	7.0.0.2
7.0.0.0/24	1	-
9.0.0.0/24	1	-

Tegyük fel, hogy megszakad a B és C közti összeköttetés.

Counting to infinity



Dst	Hop	Next
3.0.0.0/24	1	-
5.0.0.0/24	1	-
7.0.0.0/24	2	5.0.0.2
9.0.0.0/24	3	5.0.0.2

Dst	Hop	Next
3.0.0.0/24	2	5.0.0.1
5.0.0.0/24	1	-
7.0.0.0/24	1	-
9.0.0.0/24	2	7.0.0.1

Dst	Hop	Next
3.0.0.0/24	3	7.0.0.2
5.0.0.0/24	2	7.0.0.2
7.0.0.0/24	1	-
9.0.0.0/24	1	-

B (5.0.0.2)

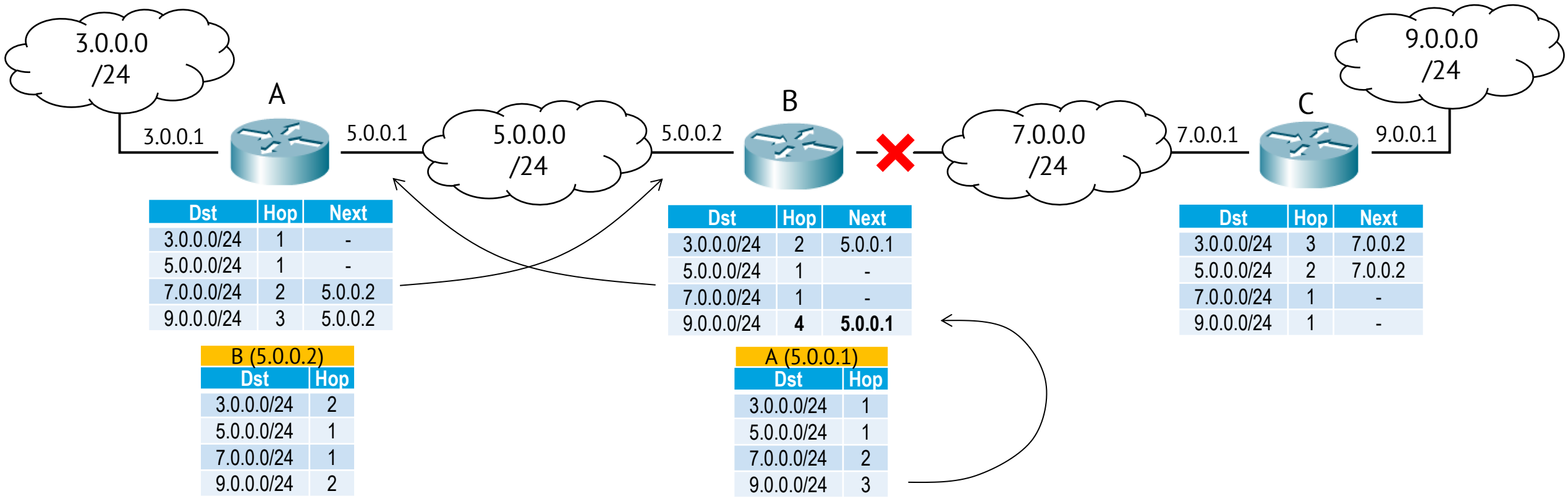
Dst	Hop
3.0.0.0/24	2
5.0.0.0/24	1
7.0.0.0/24	1
9.0.0.0/24	2

A (5.0.0.1)

Dst	Hop
3.0.0.0/24	1
5.0.0.0/24	1
7.0.0.0/24	2
9.0.0.0/24	3

Ekkor a „B” eszköz nem kapja meg a „C” hirdetését, viszont az „A” eszköz továbbra is hirdeti, hogy rajta kereszül 3 hoppal elérhető a 9.0.0.0/24 hálózat.

Counting to infinity



Dst	Hop	Next
3.0.0.0/24	1	-
5.0.0.0/24	1	-
7.0.0.0/24	2	5.0.0.2
9.0.0.0/24	3	5.0.0.2

Dst	Hop	Next
3.0.0.0/24	2	5.0.0.1
5.0.0.0/24	1	-
7.0.0.0/24	1	-
9.0.0.0/24	**4**	**5.0.0.1**

Dst	Hop	Next
3.0.0.0/24	3	7.0.0.2
5.0.0.0/24	2	7.0.0.2
7.0.0.0/24	1	-
9.0.0.0/24	1	-

B (5.0.0.2)

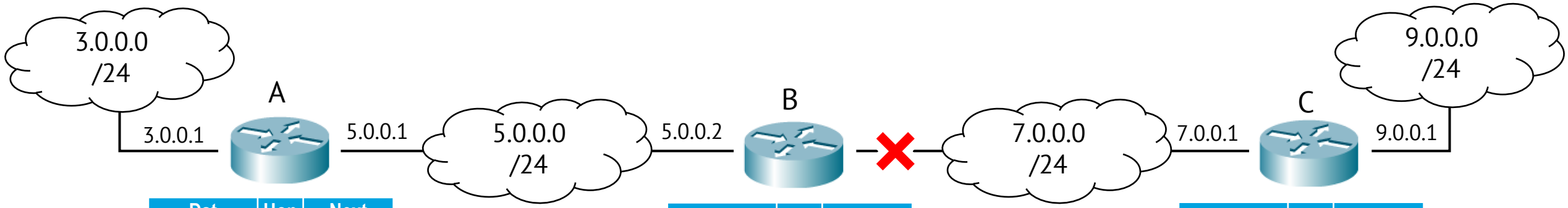
Dst	Hop
3.0.0.0/24	2
5.0.0.0/24	1
7.0.0.0/24	1
9.0.0.0/24	2

A (5.0.0.1)

Dst	Hop
3.0.0.0/24	1
5.0.0.0/24	1
7.0.0.0/24	2
9.0.0.0/24	3

Ennek fényében (mivel a „C”-től nem jött hirdetés), a „B” módosítja a routing tábláját, a 9.0.0.0/24-es hálózatot az „A”-n kersztül fogja keresni 4 hoppal.

Counting to infinity



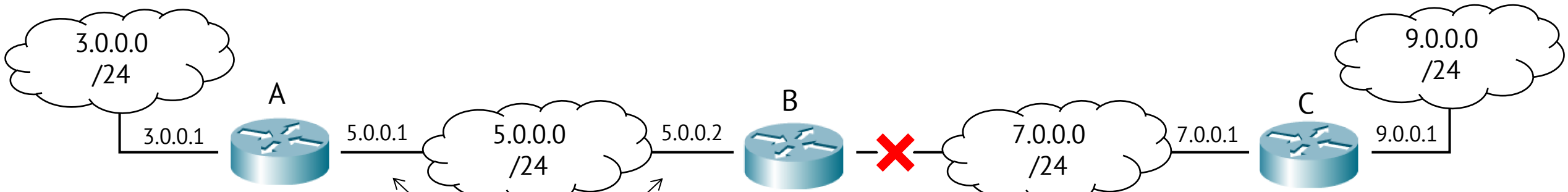
Dst	Hop	Next
3.0.0.0/24	1	-
5.0.0.0/24	1	-
7.0.0.0/24	2	5.0.0.2
9.0.0.0/24	3	5.0.0.2

Dst	Hop	Next
3.0.0.0/24	2	5.0.0.1
5.0.0.0/24	1	-
7.0.0.0/24	1	-
9.0.0.0/24	**4**	**5.0.0.1**

Dst	Hop	Next
3.0.0.0/24	3	7.0.0.2
5.0.0.0/24	2	7.0.0.2
7.0.0.0/24	1	-
9.0.0.0/24	1	-

Létrejön egy „hibás” routing tábla, amiben egy értelmetlen route szerepel.
Ezt egyébként szépen tovább is hirdeti majd a „B”...

Counting to infinity



Dst	Hop	Next
3.0.0.0/24	1	-
5.0.0.0/24	1	-
7.0.0.0/24	2	5.0.0.2
9.0.0.0/24	3	5.0.0.2

Dst	Hop	Next
3.0.0.0/24	2	5.0.0.1
5.0.0.0/24	1	-
7.0.0.0/24	1	-
9.0.0.0/24	4	5.0.0.1

Dst	Hop	Next
3.0.0.0/24	3	7.0.0.2
5.0.0.0/24	2	7.0.0.2
7.0.0.0/24	1	-
9.0.0.0/24	1	-

B (5.0.0.2)

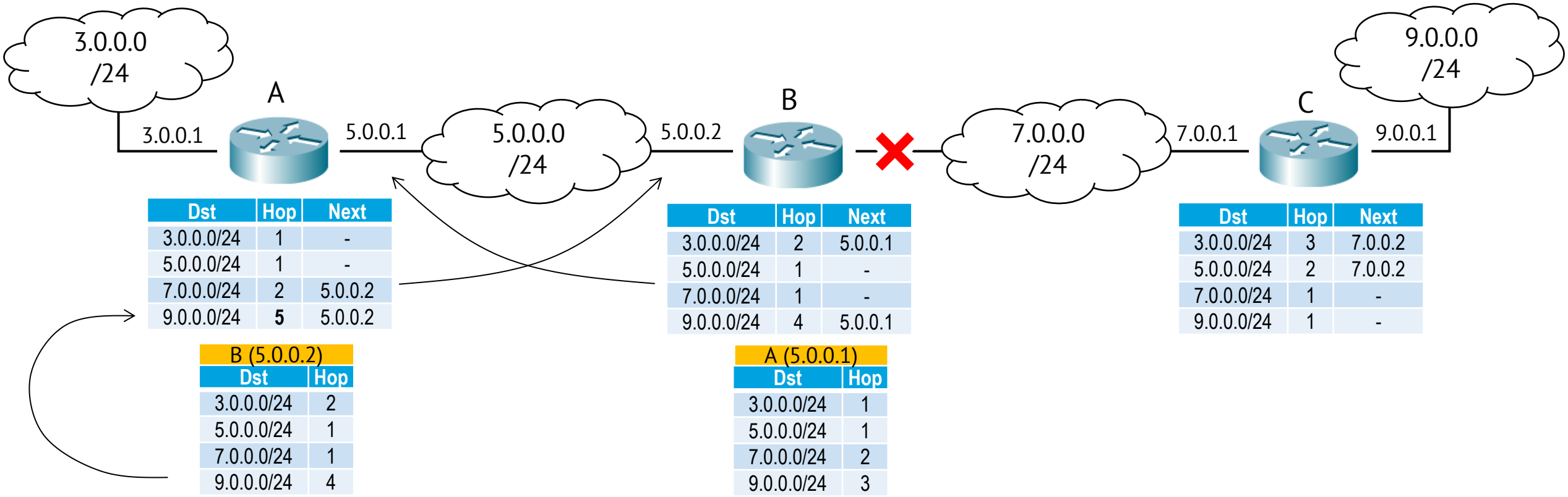
Dst	Hop
3.0.0.0/24	2
5.0.0.0/24	1
7.0.0.0/24	1
9.0.0.0/24	4

A (5.0.0.1)

Dst	Hop
3.0.0.0/24	1
5.0.0.0/24	1
7.0.0.0/24	2
9.0.0.0/24	3

A következő körben ismét üzenetet cserélnek. Ebben „B” már a nem valós route-ot hirdeti a 9.0.0.0/24 hálózathoz.

Counting to infinity



Dst	Hop	Next
3.0.0.0/24	1	-
5.0.0.0/24	1	-
7.0.0.0/24	2	5.0.0.2
9.0.0.0/24	**5**	5.0.0.2

Dst	Hop	Next
3.0.0.0/24	2	5.0.0.1
5.0.0.0/24	1	-
7.0.0.0/24	1	-
9.0.0.0/24	4	5.0.0.1

Dst	Hop	Next
3.0.0.0/24	3	7.0.0.2
5.0.0.0/24	2	7.0.0.2
7.0.0.0/24	1	-
9.0.0.0/24	1	-

B (5.0.0.2)

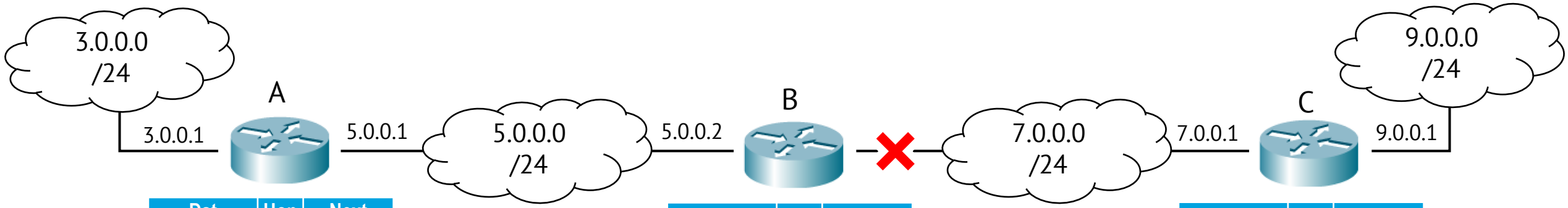
Dst	Hop
3.0.0.0/24	2
5.0.0.0/24	1
7.0.0.0/24	1
9.0.0.0/24	4

A (5.0.0.1)

Dst	Hop
3.0.0.0/24	1
5.0.0.0/24	1
7.0.0.0/24	2
9.0.0.0/24	3

Az „A”, mivel az eddig is használt „B”-től kap egy megnövekedett hop értéket, frissíteni fogja a saját hop countját a 9.0.0.0/24-hez.

Counting to infinity



Dst	Hop	Next
3.0.0.0/24	1	-
5.0.0.0/24	1	-
7.0.0.0/24	2	5.0.0.2
9.0.0.0/24	5	5.0.0.2

Dst	Hop	Next
3.0.0.0/24	2	5.0.0.1
5.0.0.0/24	1	-
7.0.0.0/24	1	-
9.0.0.0/24	4	5.0.0.1

Dst	Hop	Next
3.0.0.0/24	3	7.0.0.2
5.0.0.0/24	2	7.0.0.2
7.0.0.0/24	1	-
9.0.0.0/24	1	-

Ez a folyamat a hop count növekedését eredményezi egészen addig, amíg helyre nem áll a „B”–„C” kapcsolat, vagy a hop count el nem éri a *végtelent*.

Counting to infinity

A loopok elkerülése érdekében érdemes a végtelent kis számként megadni :P

Vagyis pl, ha azt állítjuk be, hogy hop count > 16 után már ne tekintsük értelmesnek az útvonalat, akkor kb 16 iteráció után leáll ez a ping-pong.

Így viszont nem tudunk 16 hop-nál nagyobb hálózatot kezelni :'(

További ötletek

Split horizon
Ha egy routert csak pontosan egy darab másik routertől hallunk hirdetni, akkor visszafelé nem hirdetjük azt.

Poisoned reverse
Ha egy routert csak pontosan egy darab másik routertől hallunk hirdetni, akkor visszafelé végtelen költséggel hirdetjük azt.

Ezek célja, hogy a counting to infinity eseteket „csírájában elfojtsák”.

Nem tökéletes módszerek, körök esetében meg tudnak zavarodni.

#04/5 – Összefoglalás

Fogalmak

Link state protocol

Distance vector protocol

Counting to infinity

#04/6 – IGP családba tartozó protokollok

Interior Gateway Protocol

Feladata, hogy az egy AS-en belüli gatewayek
el tudják látni a routing feladatot.

Meghatároz olyan protokollokat, amik a gatewayek közötti
routing információ kicserélését teszik lehetővé.

Két tanult implementációja:

- Routing Information Protocol (RIP)
- Open Shortest Path First (OSPF)

RIP jellemzők

A Routing Information Protocol a következő jellemzőkkel bír:

- **distance vector alapú**
  a routerek a szomszédos routerek hirdetései alapján keresnek útvonalat
- **IGP családba tartozik**
  az AS-en belüli információcserére használható
- **hop count, célhálózat és gateway címeket küld**
  egy gateway ilyen üzenetekből frissíti a saját routing tábláját
- **split horizon és poisoned reverse használata**
  a counting to infinity elkerülésére, illetve a *végtelen* értéke 16
- **triggered update**
  ha valahol „erőből” változik a routing tábla, akkor azt azonnal hirdeti
- **UDP felett működik**
  két verziója van: RIP1 és RIP2

RIP üzenet szerkezete

A RIP üzenet a következő egységekből áll:

cmd 1 byte	version 1 byte	routing domain 2 byte	RIP entry 20 byte

- cmd — command, az üzenet típusa (1 = request, 2 = response)
- version — RIP verzió (RIP1 vagy RIP2)
- r. domain — routing domain, RIP1 esetében csupa nulla, RIP2 esetében egy gépen több RIP szolgáltatás is futhat, ezek között segít választani
- RIP entry — a RIP üzenet maga. Ez a RIP verziótól függően más-más tartalmú.

RIP1 Entry szerkezete

A RIP1 entry a következő egységekből áll:

- **address family identifier**
  a címek típusának azonosítója. IPv4 címek esetén az értéke 2.
- **IP address**
  a hivatkozott IPv4 cím, ami a hálózat vagy a host címe
- **metric**
  a metrika értéke, vagyis a távolságot kifejező mennyiség

RIP2 Entry szerkezete

A RIP2 entry a következő egységekből áll:

- **address family identifier**
  a címek típusának azonosítója. IPv4 címek esetén az értéke 2.
- **route tag**
  az Autonomous System (AS) azonosító, ha ilyet is tud a küldő
- **IP address**
  a hivatkozott IPv4 cím, ami a hálózat vagy a host címe
- **subnet mask**
  a hirdetett IP címhez/tartományhoz tartozó maszk
- **next hop**
  én erre az IP címre routolom ezt a tartományt
- **metric**
  a metrika értéke, vagyis a távolságot kifejező mennyiség

RIP működése

A RIP-et használó gateway a routing során két dolgot művel:

Hirdet: A gateway a többi gateway számára jelenti, hogy mely hálózatokat látja közvetlenül.

Tanul: Ha egy másik gatewaytől olyan üzenetet kap, amiben *még számára ismeretlen* hálózatot lát, akkor felveszi azt a routing táblába.

Ha egy másik gatewaytől olyan üzenetet kap, amiben egy ismert hálózatot

kisebb költséggel elérhetőnek lát, akkor frissíti a routing táblát.

RIP korlátai

A RIP alkalmazásának vannak korlátai:

- **egy célhálózat felé csak egy útvonal lehetséges**
  (csak rövidebb út esetén cserél, alternatívát nem tárol el)

- **csak a hop count-ot használja a költség jelzésére**
  (nem jellemzi pl. az átviteli sebességet, fizikai távolságot, üzemeltetési költséget)

- **nagy hálózat esetén nem használható**
  (sokáig tart, míg szinkronizálódnak a táblák, túl sor RIP üzenet fog keringeni)

RIP egyebek

A RIP2 támogatja az autentikációt
Védekezés az ellen, hogy valaki kamu RIP csomagokkal elterelje a forgalmat.

Időzítések:
update: 30 mp – ennyi időnként történik hirdetés

timeout: 180 mp – ha ennyi ideig nem kap update-et valahonnan, akkor az oda vezető utat végtelenre állítja

garbage collection: 120 mp – a törlésre szánt (végtelen költségű) utak ennyi idő után valóban törtlődnek.

OSPF jellemzők

Az Open Shortest Path First a következő jellemzőkkel bír:

- **link state protokoll alapú**
  a teljes hálózatot figyelembe veszi az útkeresésnél
- **open**
  nyílt forrású, ingyenesen hozzáférhető
- **több utat is nyilvántart ugyanoda, köztük választani lehet:**
  - **Type of Service figyelembe vétele**
    pl. fontosabb üzenet gyorsabb alternatív útvonalon
  - **load balancing**
    terhelés egyenletes elosztására törekszik
- **hierarchikus felépítésű**
  méretnövekedést jól kezeli
- **authentikációt használ**
  véd a hamis(ított) routing információktól

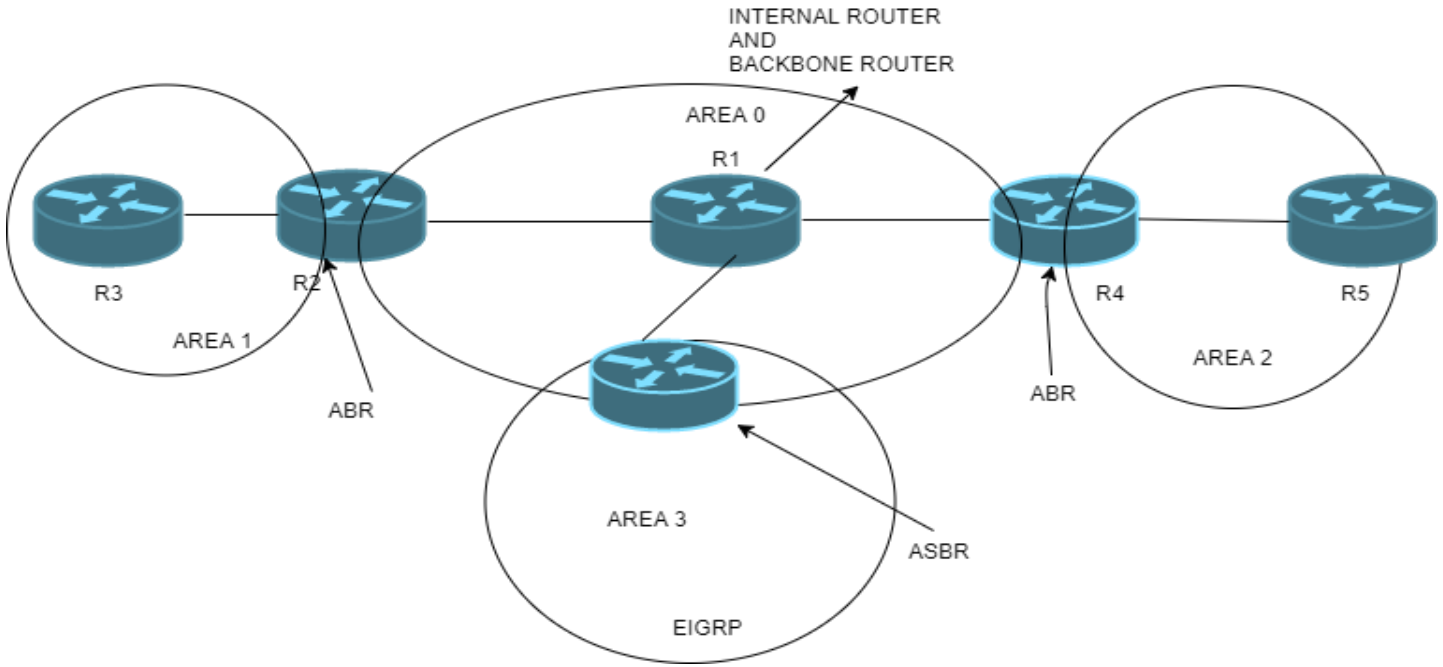
OSPF felépítése

Az AS-eket területekre osztja (area).

Minden area kapcsolódik a backbone (gerinc) area-hoz.

Az összes gateway érzékeli a hozzá kapcsolt hálózatok állapotát (link state), és ezt hirdeti az area-n belül. Az area-n belüli összes router látja az area felépítését, ezen belül Dijkstra algoritmussal legrövidebb utat számolnak.

Az area-k között border router-ek teremtenek összeköttetést.



#04/6 – Összefoglalás

Fogalmak

RIP működése és korlátai

OSPF jellemzői és felépítése

#04/7 – EGP családba tartozó protokollok

Exterior Gateway Protocol

Feladata, hogy az egy AS-ek határainál lévő gatewayek el tudják látni a routing feladatot.

Meghatároz olyan protokollokat, amik a különböző AS-ek határain lévő gatewayek közötti routing-ot teszik lehetővé.

A tanult implementációja:

- Border Gateway Protocol (BGP)

Border Gateway Protocol

Az interneten ezen alapul a routing, a backbone (gerinc) routerek ezt használják.

Distance vector szerű protokoll, sok egyéb kiegészítéssel:

- **hirdetések szűrése**
  pl. nem fogadunk el /24-nél kisebb blokkokat, ezeket oldják meg a szolgáltatók
- **AS path**
  a célhoz vezető AS-eket tartjuk számon (hop count és router helyett).
- **BGP peers**
  a szomszédok (peer-ek) kézzel vannak megadva, statikus út köti össze őket
- **időzítések**
  nem hirdetünk rendszeresen, elég csak „még élek” üzenetet küldeni ütemesen.

Border Gateway Protocol

- **TCP alapú**
  nem boradcast (vagy multicast), hanem 1-1 kapcsolat a TCP 179-es porton
- **BGP dampening**
  a gyakran változó hirdetéseket nem vesszük figyelembe („ne ugráltass”)
- **útvonalválasztás elve:**
  - a specifikusabb (hosszabb netmaszkú) útvonal a preferált
  - a lokális (AS-en belüli) útvonal a preferált
  - a rövidebb AS-path a preferált

Az AS-eket az internetben való részvételük alapján három csoportba sorolja:

- **stub** – csak egy bekötése van, végfelhasználókat tartalmaz
- **multi-connected** – több bekötése van, de átmenő forgalmat nem enged
- **transit** – több bekötéssel rendelkező hálózat, kifejezetten átmenő forgalom céljára

Looking glass, Route serverek

A **looking glass** egy diagnosztikai eszköz: http://www.bgp4.as/looking-glasses

Az interneten elszórtan fellelhető backbone routerektől kérdezhető meg mindenféle információ:

- BGP információ
- traceroute
- ping

A **route szerverek** az interneten telnettel elérhető routerek, szintén diagnosztikai célokra: http://www.traceroute.org/

#04/7 – Összefoglalás

Fogalmak

BGP

Looking glass

Route serverek

#04/8 – NAT, VPN

Network Address Translation

Előfordulhat, hogy a helyi hálózatban használt IP címet nem akarjuk kiküldeni a router másik oldalán lévő hálózatba, mert...

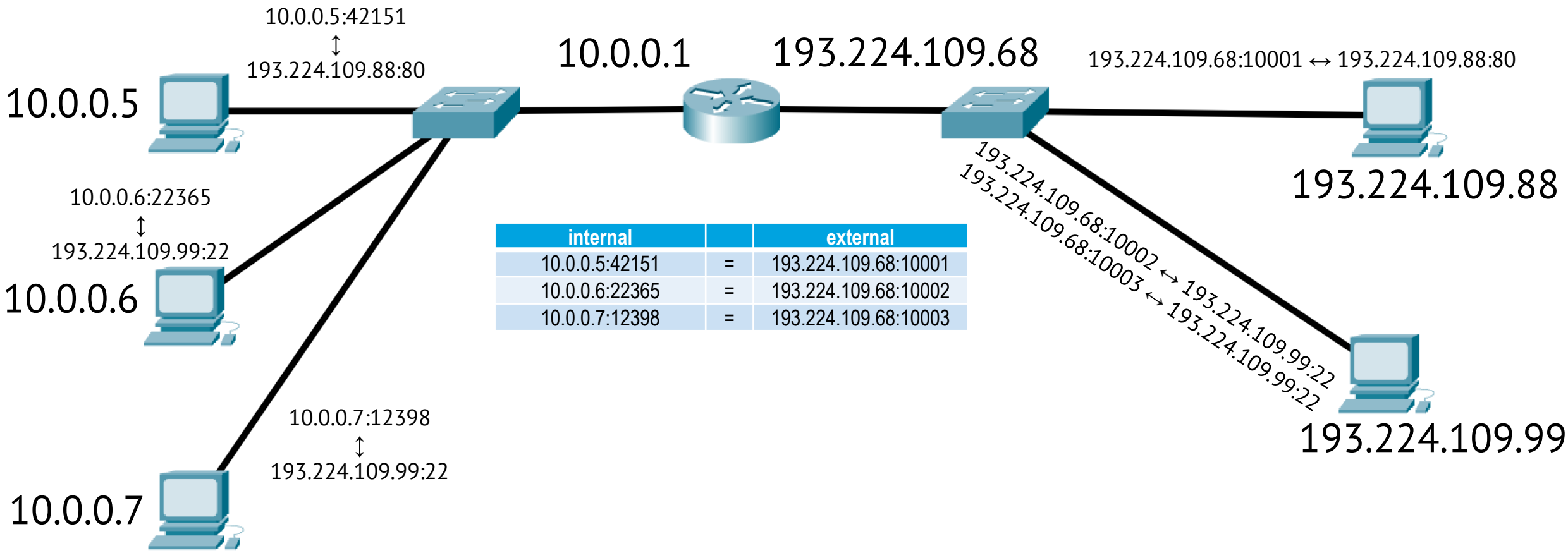
- az egy globálisan tiltott cím (pl. 192.168.0.0/16)
- az egy olyan cím, amit nem akarunk mutogatni (biztonsági okokból)
- furán használjuk a hálózatot, és a másik oldalon is van egy pont ilyen IP cím, amivel össze tudnánk akadni, és ez nem lenne jó

Megoldás: a routeren kifelé menő csomagban kicseréljük a belső IP címet a router címére; kifelé úgy tűnik, mintha minden csomag a routertől jött volna.
A router (tűzfal) mögötti hálózat így rejtve marad a külvilág számára.

Network Address Translation

NAT-olás esetén a routernek emlékezni kell, hogy mit mire cserélt ki.

Táblázatot kell vezetni az élő kapcsolatokról.



internal		external
10.0.0.5:42151	=	193.224.109.68:10001
10.0.0.6:22365	=	193.224.109.68:10002
10.0.0.7:12398	=	193.224.109.68:10003

Virtual Private Network

Előfordulhat, hogy „helyi” hálózatot szeretnénk látni, de nem tudunk ahhoz közvetlenül kapcsolódni (messze vagyunk).

Ekkor megtehetjük, hogy a „nem helyi” hálózaton küldünk adatot, amit a két hálózatot összekapcsoló eszköz kibont, és bejuttat a helyi hálózatba.

Ez az eszköz lesz a VPN szerver, ami csak akkor engedi látni a belső hálózatot, ha mi megfelelően azonosítottuk magunkat. A VPN szerver és VPN kliens közötti adatátvitel titkosított formában történik.

Virtual Private Network

VPN használata esetén az „A” eszköz által helyi címre küldendő adatokat a VPN kliens titkosítja, és ezt a titkosított adatot küldi a VPN szervernek. A szerver kicsomagloja azt, és ha jók az azonosító adatok (pl. felhasználónév, jelszó), akkor beküldi a kititkosított adatot a helyi hálózatba.

A titkosított forgalmat a gonosz világ nem tudja elolvasni. A „B” számára (és az „A” számára is) úgy tűnik, mintha ugyanazon a helyi hálózaton lennének.

#04/8 – Összefoglalás

Fogalmak NAT

VPN

VÉGE

PÁZMÁNY

Pázmány Péter Katolikus Egyetem

Információs Technológiai és Bionikai Kar