Az internet ökoszisztémája és evolúciója

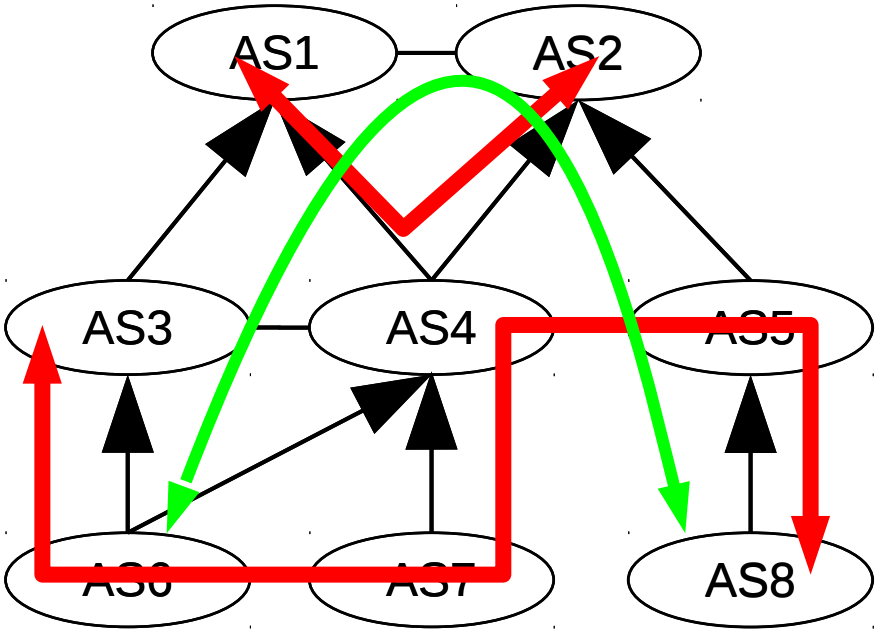
Tartalom

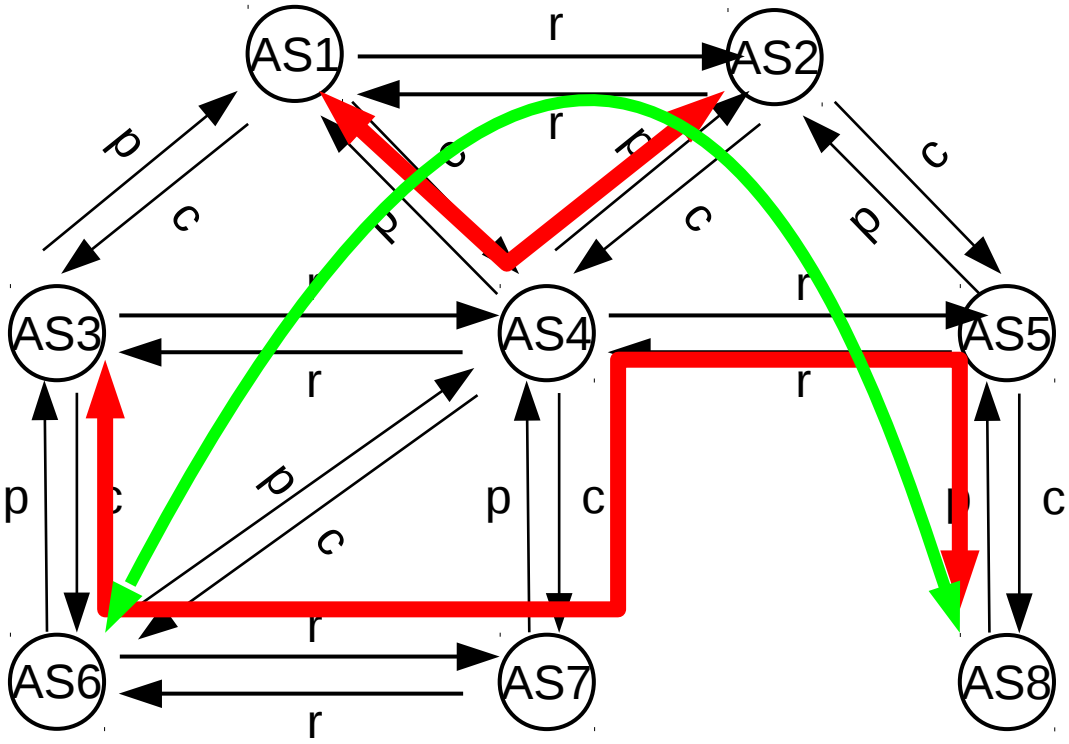
- A BGP a gyakorlatban
  - "prefer customer" szabály lokális preferencia beállításával és a legrövidebb AS út
  - prefix hijacking és prefix szűrés
  - AS útvonalak szűrése
  - backup routing és AS-path prepending
  - hot-potato routing
  - forgalommenedzsment
- Az internet útvonalválasztás stabilitása
  - BGP oszcillációk

BGP a gyakorlatban

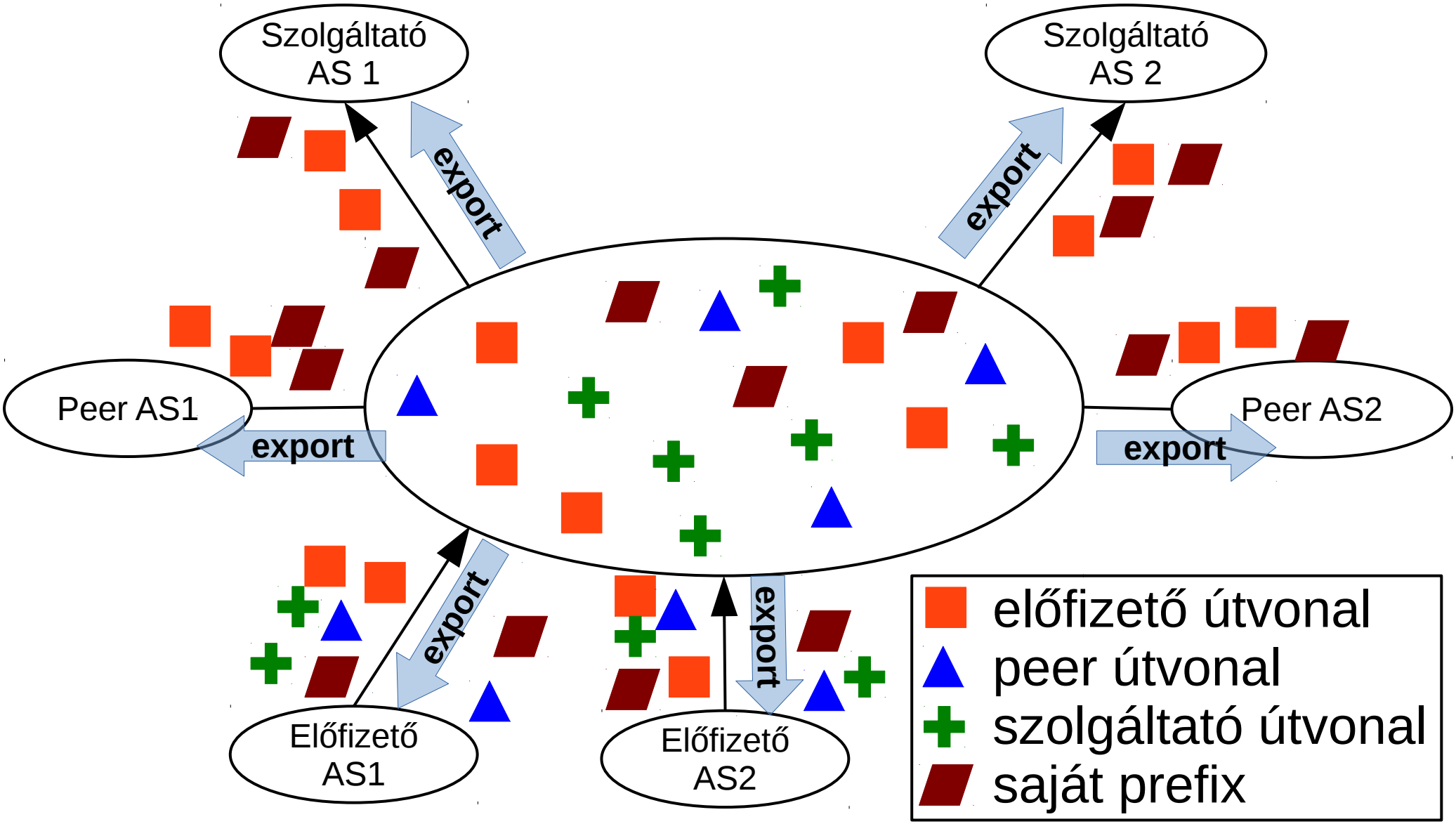
Ismétlés: AS-szintű útválasztás

- Alapvető AS–AS üzleti viszony: **tranzit/peer**
- Engedélyezett/tiltott utak: **valley-free routing**
- BGP konfigurációval kell megvalósítani





Ismétlés: „valley-free” BGP szűrők



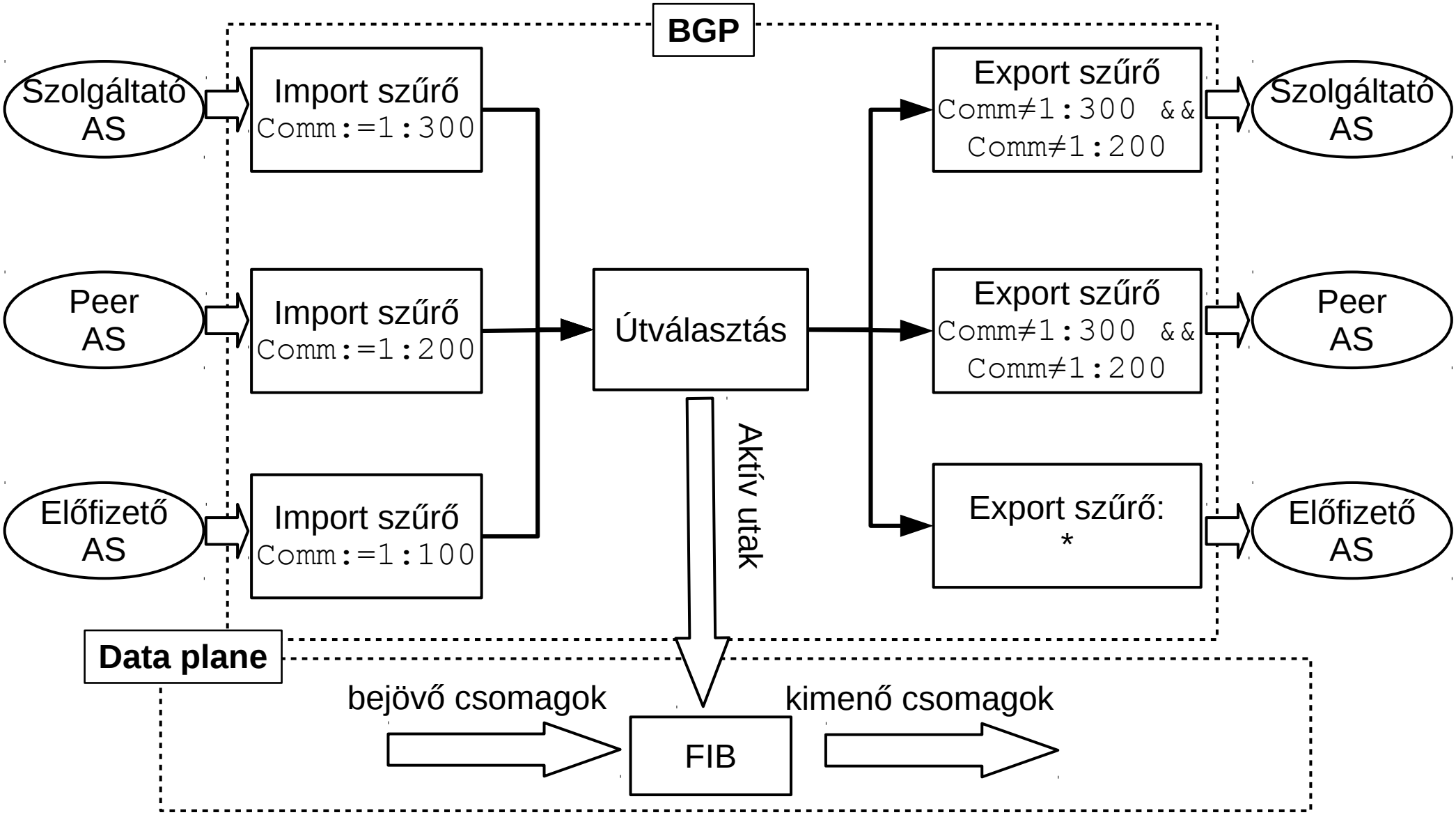
Ismétlés: BGP

- A BGP routerek hirdetéseket generálnak:

  hirdetés = prefix + attribútumok

- Fontos attribútumok: AS_PATH, NEXT_HOP, LOCAL_PREF, COMMUNITIES
- Kapott hirdetések → **import szűrő** → BGP RIB → **aktív útvonal kiválasztása** → FIB
  - magasabb lokális preferenciájú hirdetés nyer
  - döntetlen esetén a rövidebb AS úthossz dönt
- Az aktív utat továbbadjuk a szomszédoknak
  - **export szűrőn** szűrhető

Ismétlés: BGP valley-free routing

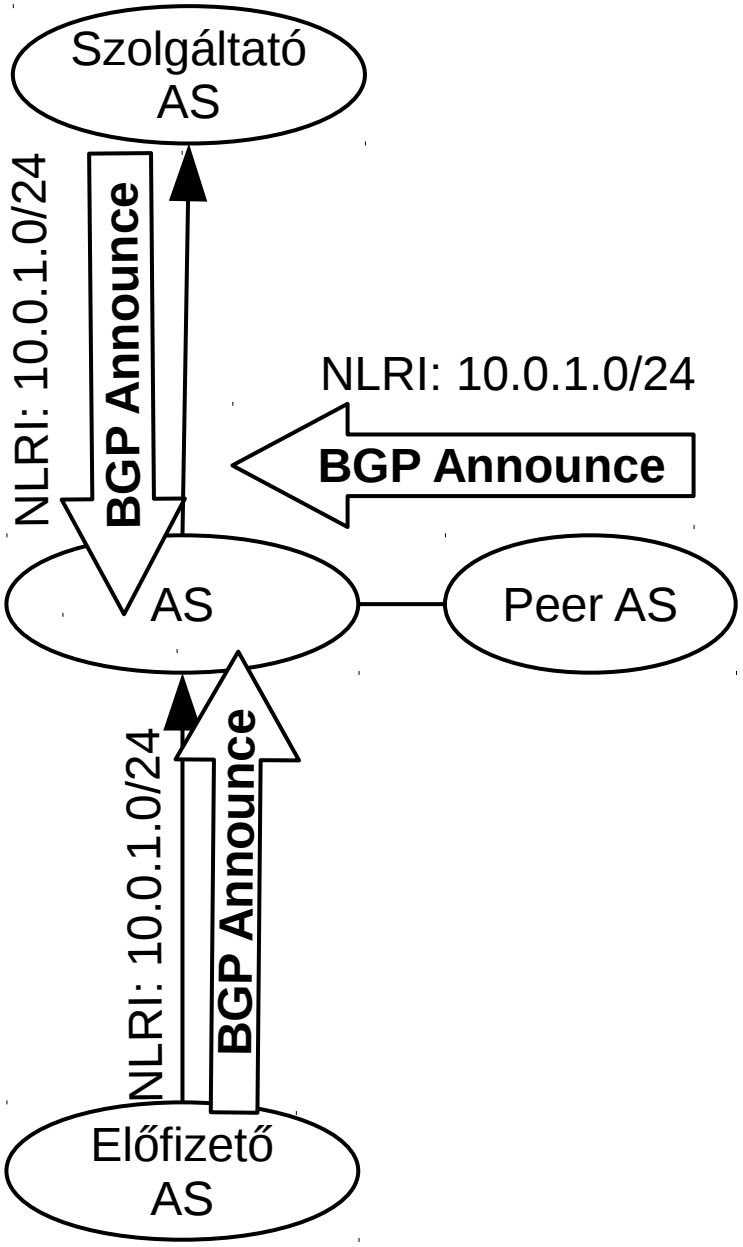


További útválasztási stratégiák

- Mivel az ISP-k piaca igen kompetitív
  - szuboptimális útvonal profitveszteséget okoz
  - sőt, támadási felületet is (prefix hijacking)
- Rendkívül finom ISP-szintű útválasztási stratégiák szükségesek (túl a valley-free szabályokon)
- Az alábbiakban áttekintünk néhány tipikus stratégiát és azok BGP konfigurációját

Prefer customer BGP felett

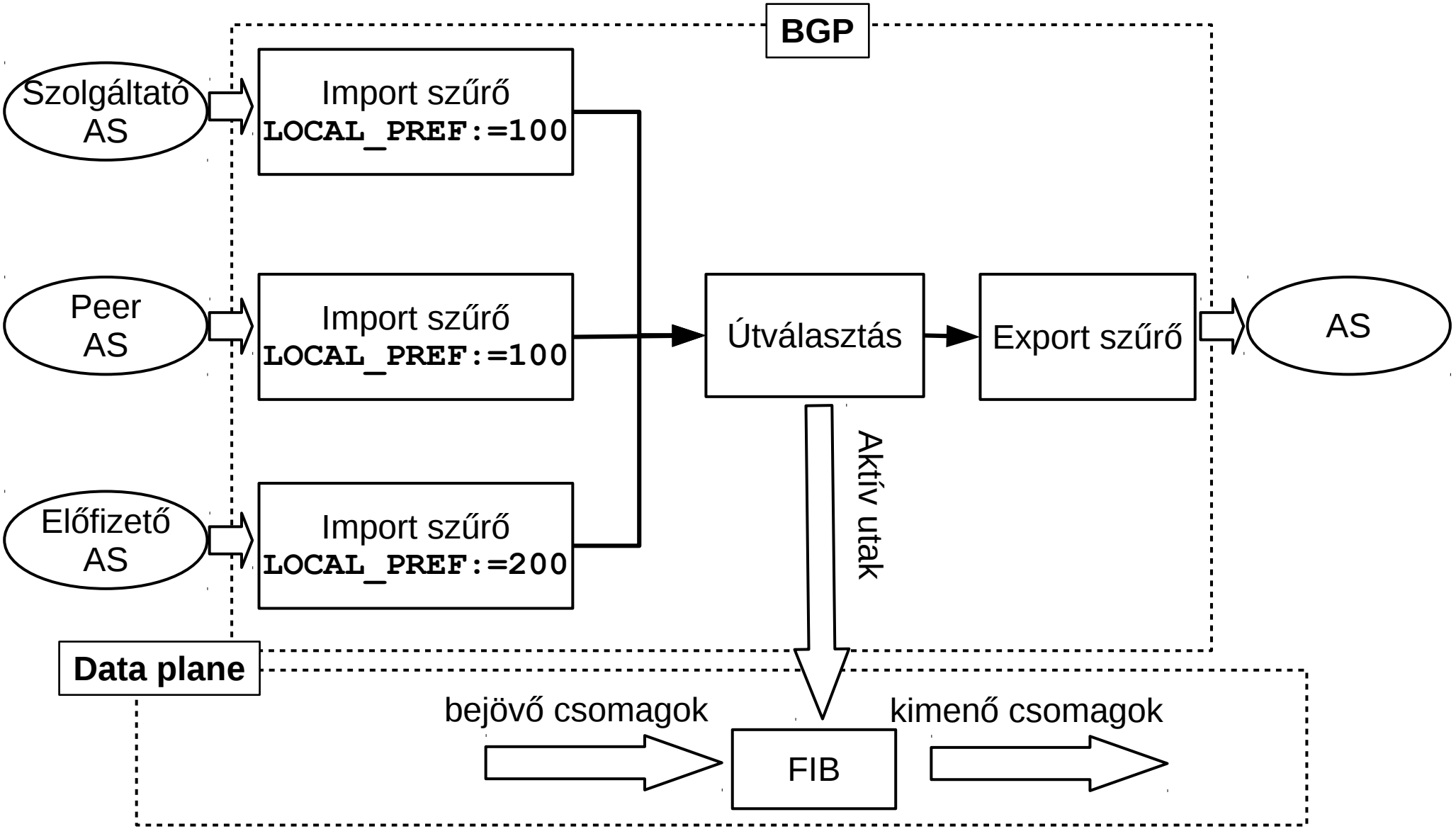
- A **prefer customer** szabály értelmében az előfizetőktől érkező hirdetések preferáltak az útválasztásban
- Különben szolgáltatón át továbbítunk az előfizető felé: anyagi veszteség
- A BGP **lokális preferencia** attribútumát fogjuk használni a szabály érvényesítésére



Prefer customer: import szűrő

- Emlékeztető: a **lokális preferencia** attribútum értéke a legnagyobb prioritású tényező a BGP döntési folyamatában
  - ha a BGP egy prefixre több hirdetést kap
  - a nagyobb LOCAL_PREF attribútumú nyer
- **BGP konfiguráció:** az előfizetőktől érkező hirdetések **lokális preferencia attribútumát** egy **import szűrőn magas értékre** állítjuk
- Így a BGP útvonalválasztásban automatikusan preferáltak az előfizetői utak (magas lok. pref.)

Prefer customer: import szűrő

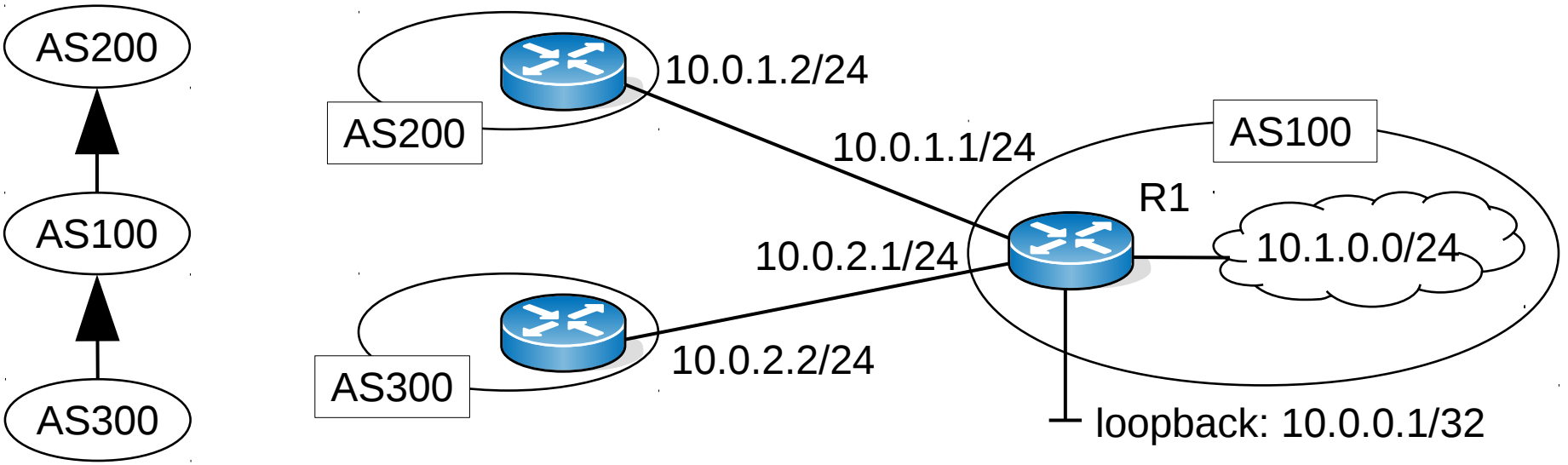


A BGP döntési mechanizmusa

Prioritás	BGP attribútum	Felhasználás
1.	Magasabb Lokális Preferencia (LOCAL_PREF)	• Elsődleges szolgáltató kiválasztása • BGP határrouterek összehangolása iBGP-n
2.	Rövidebb AS út (AS_PATH)	• Forgalom menedzselése AS-en belül a határrouterek között
3.	Alacsonyabb Multi-Exit Discriminator (MED)	
4.	iBGP-től kapott hirdetés preferált eBGP-től kapott hirdetés előtt	
5.	Kisebb IGP költség a határrouterig	
6.	Alacsonyabb router-id	• Ha még mindig több azonosan preferált hirdetés, akkor egyik itt boztos „nyer”

Prefer customer: BGP konfiguráció

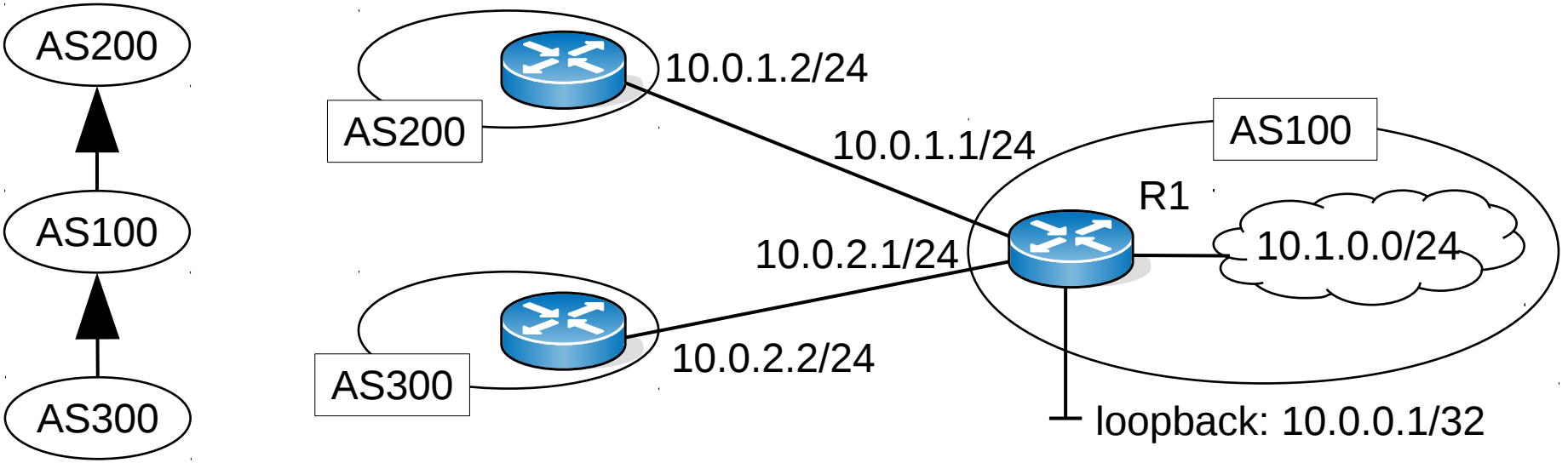
- Tegyük most fel, hogy `AS300` előfizető `AS200` pedig szolgáltató
- Ekkor a „prefer customer” szabály értelmében az `R1` router az `AS300`-tól érkező hirdetéseket preferálja



Prefer customer: BGP konfiguráció

- Alapértelmezett LOCAL_PREF beállítás: `100`
- Elég az előfizetők hirdetésein a LOCAL_PREF értékét `200`-ra állítani egy import szűrőn:

```
route-map rm-loc-pref-200 permit 10
  set local-preference 200
```

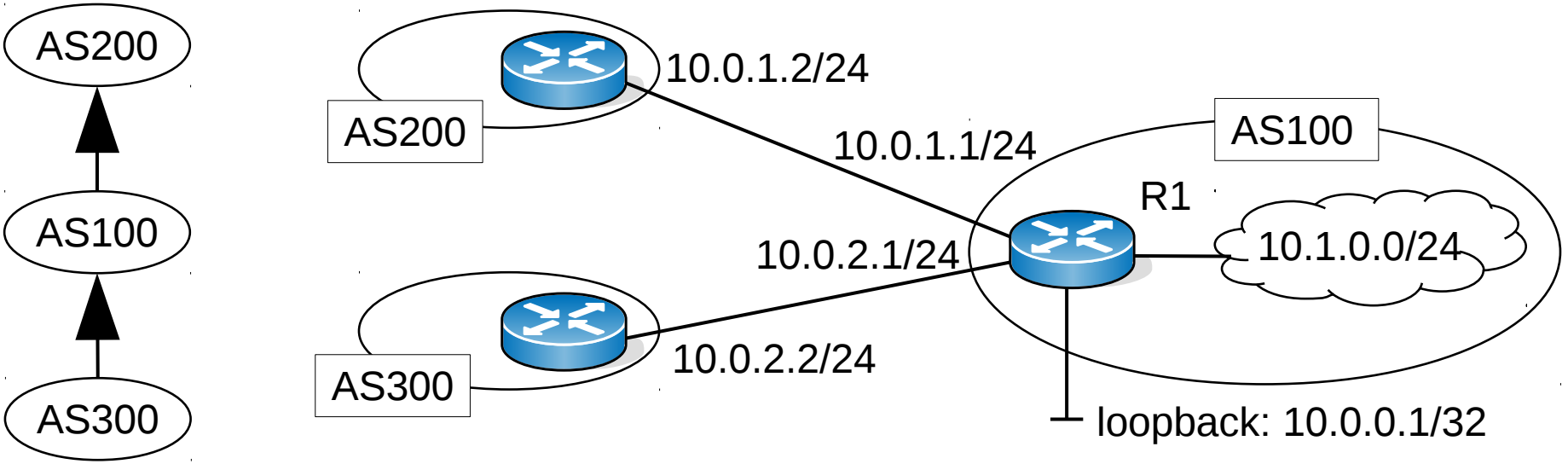


Prefer customer: BGP konfiguráció

- Csatlakoztassuk a szűrőt az előfizető AS-hez

  `neighbor 10.0.2.2 route-map rm-loc-pref-200 in`

- Az utolsó `in` klauza jelöli ki a szűrő irányát
- Más teendőnk nincs, a BGP automatikusan a magas lokális preferenciájú utat választja



Prefer customer: példa

```
!!! BGP router konfigurációja
!!! Comunity-k:
!!!    1:100: előfizető
!!!    1:200: peer
!!!    1:300: szolgáltató
router bgp 100
 bgp router-id 10.0.0.1
 network 10.1.0.0/24
 neighbor 10.0.1.2 remote-as 300
 neighbor 10.0.1.2 route-map rm-prov-set-cm in
 neighbor 10.0.1.2 route-map rm-no-export out
 neighbor 10.0.2.2 remote-as 200
 neighbor 10.0.2.2 route-map rm-cust-set-cm in
 neighbor 10.0.2.2 route-map rm-loc-pref-200 in

!!! folytatás a következő oldalon
```

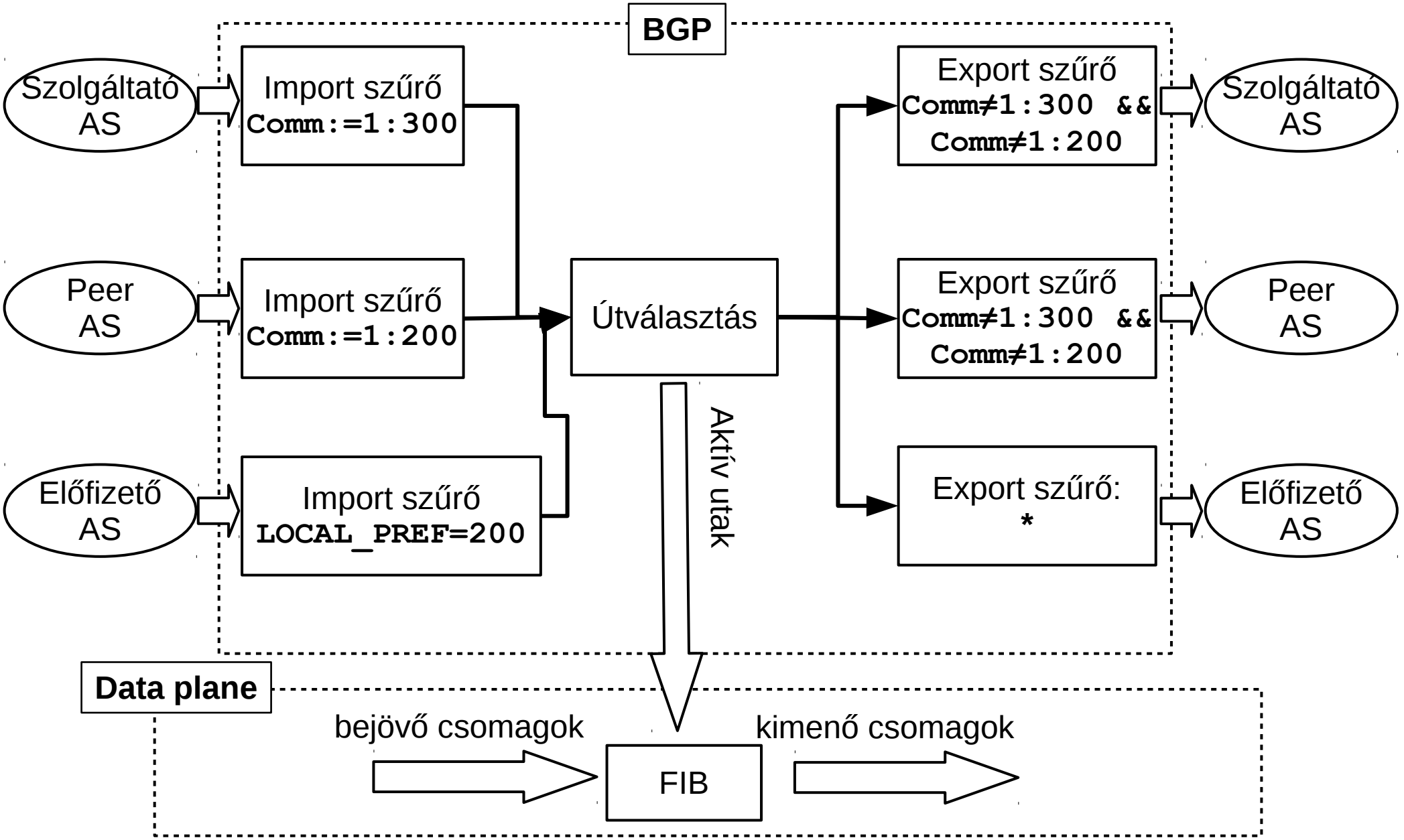
Prefer customer: példa

```
route-map rm-prov-set-cm permit 10
 set community 1:300

route-map rm-peer-set-cm permit 10
 set community 1:200

route-map rm-cust-set-cm permit 10
 set community 1:100

ip community-list standard cm-no-export permit 1:200
ip community-list standard cm-no-export permit 1:300

route-map rm-no-export deny 10
 match community cm-no-export

route-map rm-no-export permit 20

route-map rm-loc-pref-200 permit 10
 set local-preference 200
```

Legrövidebb AS út: BGP konfig

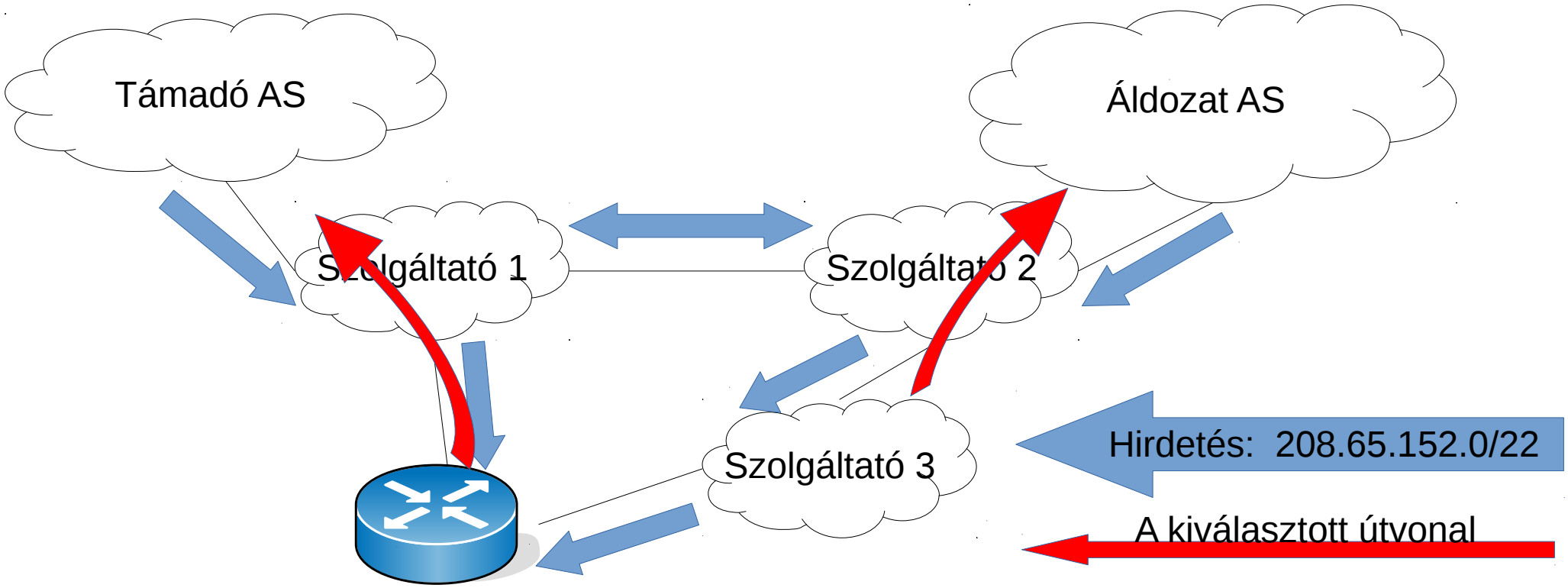
- A fenti konfigurációval beállított BGP router
  - csak **valley-free** utat közöl a szomszédokkal
  - megvalósítja a **prefer customer** szabályt
  - érdekes mód, bár nem konfiguráltuk külön, a **legrövidebb AS útvonal** szabályt is
- Ha több előfizetői út van: ezeken azonos a lokális preferencia, ilyenkor pedig az AS-út hossza dönt a BGP-ben
- De a peer és provider utak közt nem válogat!
  - ezt külön konfigurálnunk illene

BGP: valley-free+prefer-customer



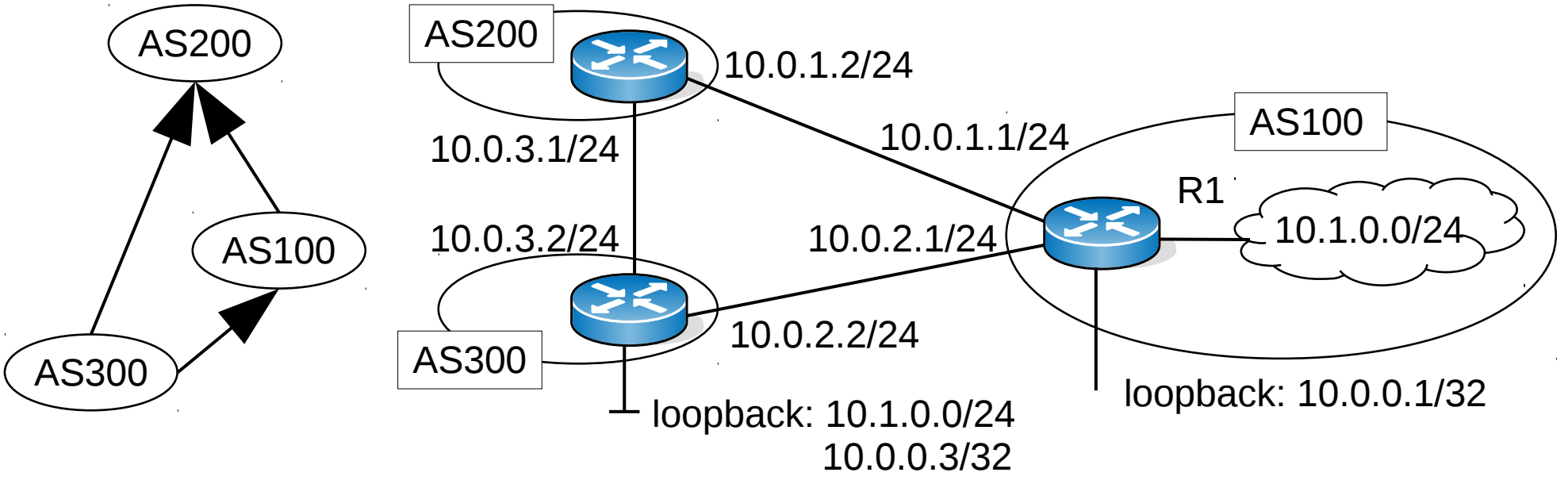
Prefix hijacking: Youtube incidens

- Egy címtartományt egyszerre többen hirdetnek
- Nem dönthető el egyértelműen, melyik a valódi
- A Támadóhoz rövidebb az AS-út: **prefix hijack**



Prefix hijacking

- Az alábbi példában `AS100` és `AS300` az `AS200` szolgáltató előfizetője, és `AS100` „legitim módon” hirdeti a `10.1.0.0/24` prefixet
- Mi van, ha `AS300` is (rosszindulatúan vagy tévedésből) meghirdeti ugyanezt a prefixet?



Prefix hijacking

```
! AS100
router bgp 100
   bgp router-id 10.0.0.1
   network 10.1.0.0/24
   neighbor 10.0.1.2 remote-as 200
   neighbor 10.0.2.2 remote-as 300
```

```
! AS300
router bgp 300
   bgp router-id 10.0.0.3
   network 10.1.0.0/24
   neighbor 10.0.3.1 remote-as 200
   neighbor 10.0.2.1 remote-as 100
```

Prefix hijacking

- AS200 BGP routerének nincs módja eldönteni, hogy melyik AS-től kapott hirdetés legális
  - szerencsés esetben a legitim hirdetést preferálja és helyes routing AS100-ba
  - rossz esetben viszont az illegális hirdetést választja aktív útként és helyezi a FIBbe
  - a 10.1.0.0/24 prefix teljes forgalmát AS300 (a támadó) felé irányítja
- Másféle módon is előidézhető prefix hijacking
  - pl. a Támadó specifikusabb prefixet hirdethet

Man-in-the-middle támadás

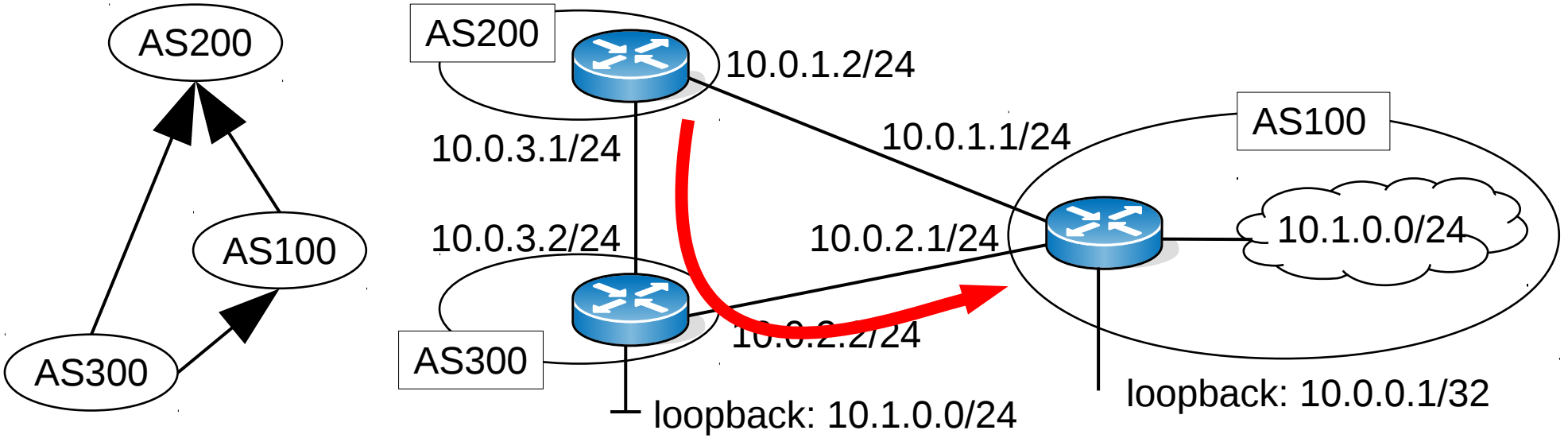
- Ha a támadó „lenyeli” `10.1.0.0/24` prefix forgalmát: a prefix elérhetetlen az internerről
  - gyakran hibás konfiguráció az ok (például a Youtube-incidens esetében)
- Ha azonban a támadás rosszindulatú: a prefix teljes forgalmát lehallgathatja anélkül, hogy az „áldozat” (a prefix tulajdonosa) ezt észlelné
- Sőt, akár káros forgalmat is „beilleszthet” (pl. vírussal fertőzheti a „lehallgatott” emaileket)
- Ráadásul az áldozat ezt észre sem veszi!

Man-in-the-middle támadás

- A MITM támadás megvalósítása a példában
- AS300 egy statikus route-ot helyez a FIBbe:

```
ip route add 10.1.0.0/24 via 10.0.2.1
```

- Az áthaladó forgalmat kedvére módosíthatja

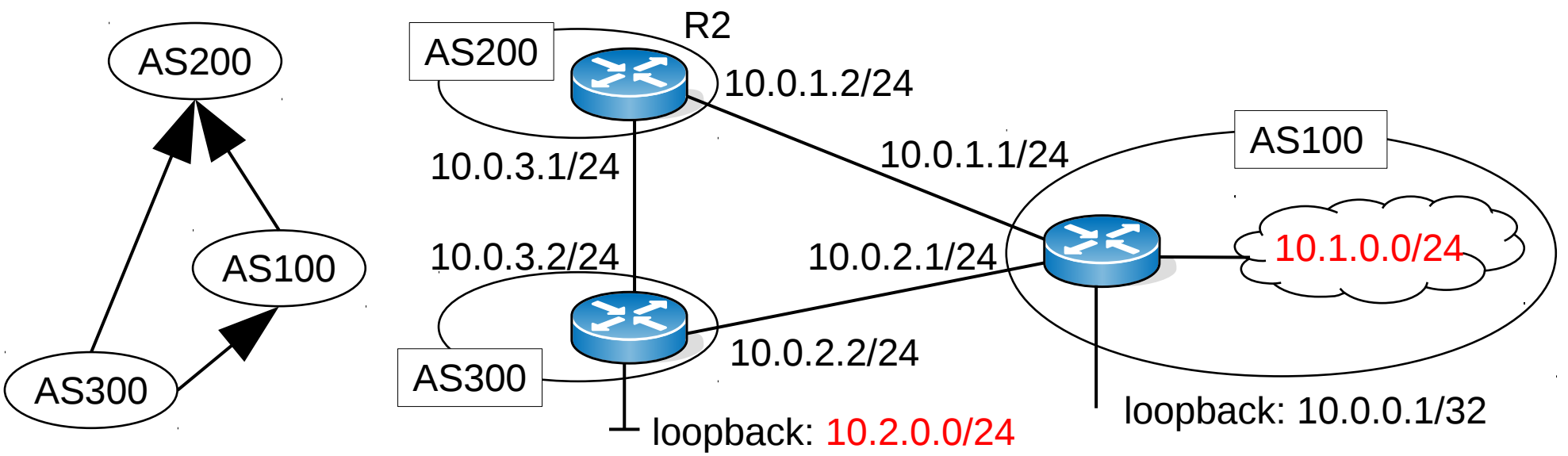


A MITM támadás megelőzése

1. **SecureBGP:** BGP hirdetések kriptografikusan aláírva
   - ellenőrizhető AS-szám–IP-prefix összerendelés
   - egyelőre nem terjedt el
2. **Jogos tulajdonos specifikusabb IP prefixeket hirdet**
   - ha például a `10.1.0.0/24` prefixet eltérítik, érdemes plusz meghirdetni a `10.1.0.0/25` és `10.1.0.128/25` prefixeket
   - több biten illeszkedik, „felülírja” a támadó bejegyzéseit a routerek FIBjeiben
3. **A jogosulatlan hirdetések szűrése**
   - AS publikálja a legális prefixeit egy „megbízható” adatbázisban: **Internet Routing Registry (IRR)**
   - szomszédos AS ez alapján BGP szűrőket konfigurál

Prefix szűrés: BGP

- Tegyük fel, hogy `AS100` legálisan hirdeti a `10.1.0.0/24` prefixet, `AS300` pedig a `10.2.0.0/24` prefixet
- `AS200` csak ezeket a prefixeket kívánja elfogadni az egyes AS-ekből

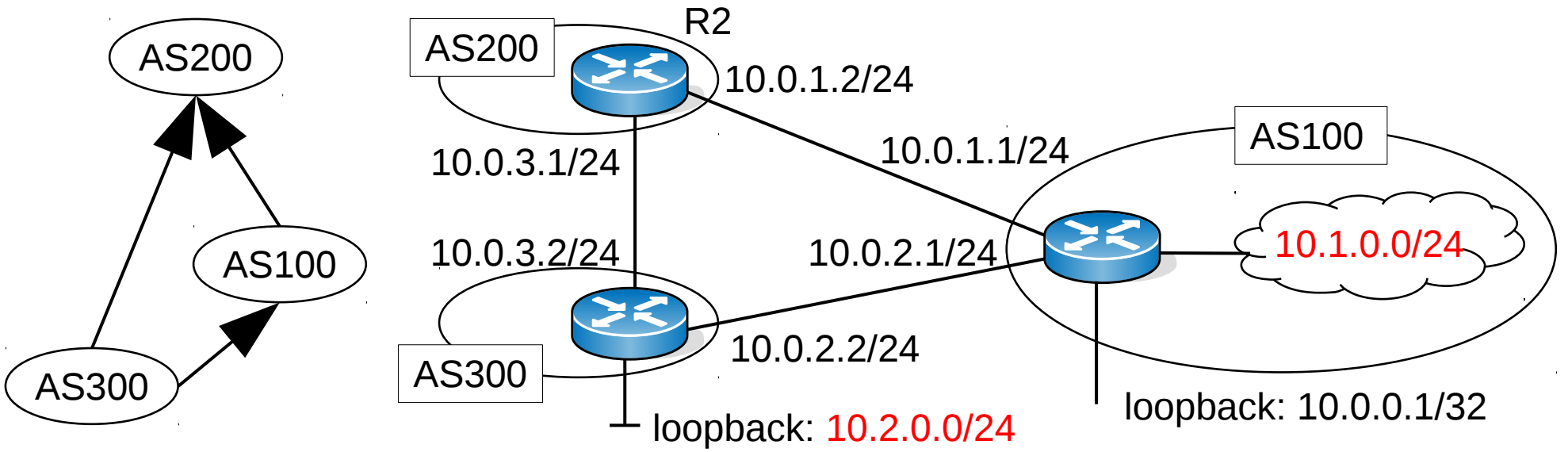


Prefix szűrés: BGP

- `R2` (`AS200` határroutere) definiálja az `AS100`-tól elfogadott prefixek listáját

```
ip prefix-list AS100 seq 5 permit 10.1.0.0/24
ip prefix-list AS100 seq 10 deny 0.0.0.0/0
```

- `seq` szerinti sorrendben, megengedett prefixek (`permit`) és visszautasított prefixek (`deny`)

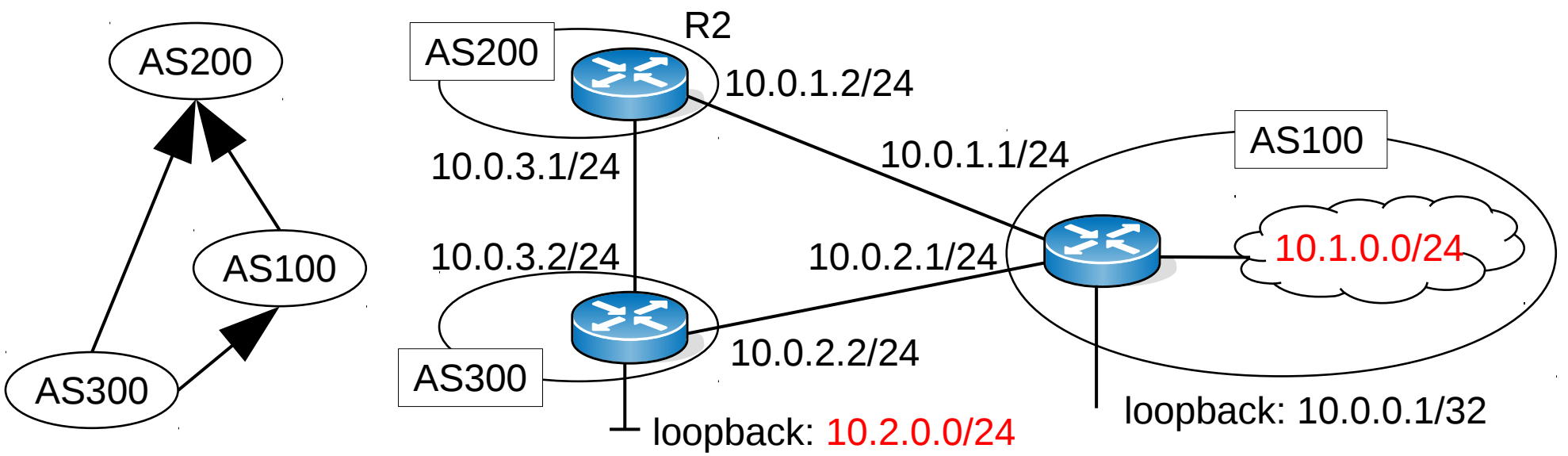


Prefix szűrés: BGP

- Majd a prefix listát hozzárendeli az `AS100`-hoz

```
neighbor 10.0.1.1 remote-as 100
neighbor 10.0.1.1 prefix-list AS100 in
```

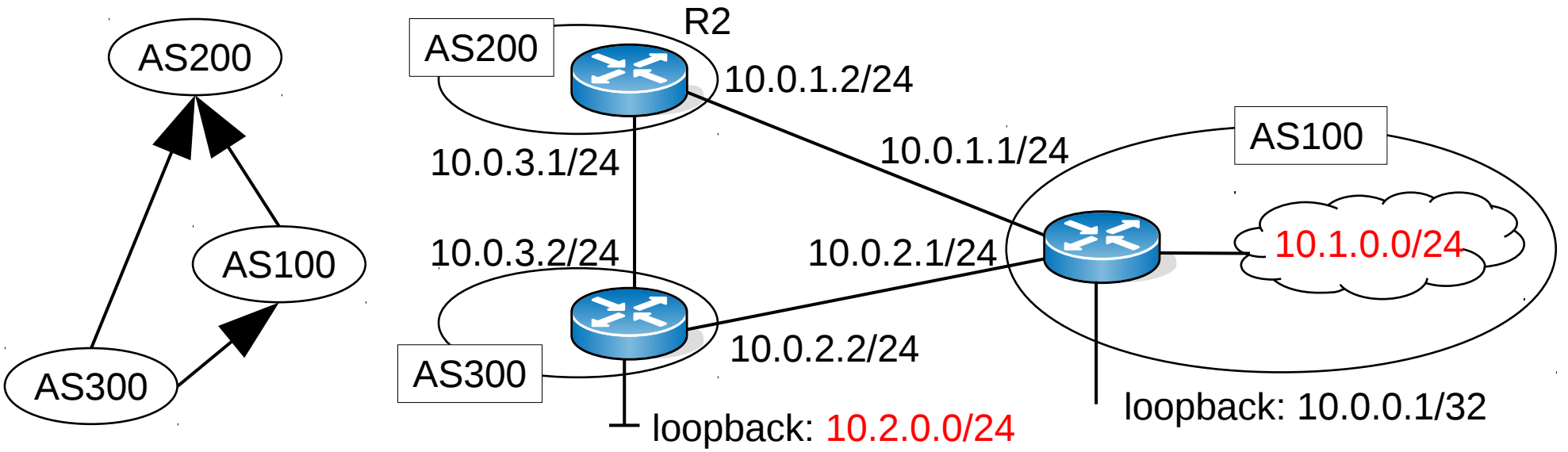
- A `prefix-list` gyakorlatilag egy speciális `route-map`, ami prefixek listájára szűr



Prefix szűrés: BGP

- `R2` hasonlóan jár el `AS300` esetében is

```
ip prefix-list AS300 seq 5 permit 10.2.0.0/24
ip prefix-list AS300 seq 10 deny 0.0.0.0/0

neighbor 10.0.3.2 remote-as 300
neighbor 10.0.3.2 prefix-list AS300 in
```



Prefix szűrés: BGP

```
router bgp 200
 bgp router-id 10.0.0.2
 neighbor 10.0.1.1 remote-as 100
 neighbor 10.0.1.1 prefix-list AS100 in
 ...
 neighbor 10.0.3.2 remote-as 300
 neighbor 10.0.3.2 prefix-list AS300 in
 ...

!!! AS100 prefixeinek szűrője
ip prefix-list AS100 seq 5 permit 10.1.0.0/24
ip prefix-list AS100 seq 10 deny 0.0.0.0/0

!!! AS300 prefixeinek szűrője
ip prefix-list AS300 seq 5 permit 10.2.0.0/24
ip prefix-list AS300 seq 10 deny 0.0.0.0/0
```

Prefix szűrés: Martians

- **Martian prefix:** speciális célra foglalt prefix
  - `0.0.0.0/8`: „This network” (RFC1122)
  - `127.0.0.0/8`: loopback címtartomány (RFC1122)
  - `192.0.2.0/24`: TEST-NET példahálózatokhoz
  - `10.0.0.0/8`, `172.16.0.0/12`, `192.168.0.0/16`: privát címtartományok intranetekhez (RFC1918)
  - `169.254.0.0/16`: auto-konfiguráció
  - `224.0.0.0/4`: multicast
  - `240.0.0.0/4`: foglalt jövőbeli felhasználásra
- Nem jelenhet meg az inter-domain routingban!

Prefix szűrés: Bogon szűrők

- Az IANA rendszeresen publikálja az AS-ek számára kiosztott IP címtartományok listáját
- A listán nem szereplő prefixek: **bogon**
  - speciális címek, ki nem osztott címek
- Érdemes BGP import szűrőkön eldobni a bogon címekre vonatkozó hirdetéseket: **bogon szűrő**
  - sok DDoS támadás bogon forráscímről
  - bogon címet tartalmazó csomagokat is szűrni érdemes a tűzfalakon
- Sok szolgáltató elvárja a prefix-szűrést az előfizetőitől

AS utak szűrése

- Egy ISP gazdasági/politikai érdekei néha megkívánják, hogy bizonyos AS-eken keresztüli útvonalakat elkerüljön
  - például az USA kormányzata nem kívánja az érzékeny forgalmát Kínán keresztül route-olni
- A BGP segítségével egy ISP precízen beállíthatja az aktív útvonalait
  - a hirdetések szűrhetők a teljes AS-útvonal alapján
  - ez lehetővé teszi, hogy bizonyos nem megbízható AS útvonalakat elkerüljünk

AS utak szűrése: BGP konfig

- Például `10.10.10.10` BGP router felől minden útvonal kiszűrése, amely tartalmazza `AS200`-at

```
ip as-path access-list 1 deny ^.*200.*$

router bgp 100
…
 neighbor 10.10.10.10 filter-list 1 in
```

- Az `ip as-path` konstrukcióval létrejön a szűrő
- A szokásos `neighbor` paranccsal illesztjük a megfelelő szomszédra

AS utak szűrése: BGP konfig

- Maga a szűrő reguláris kifejezést ad meg:
  - az AS-útvonalat alkotó AS számok aláhúzással (_) elválasztott listájára illeszthető
- `AS100-AS200` részútvonal engedélyezése

```
ip as-path access-list 1 permit 100_200
```

- Bármi, ami érinti `AS100`-at VAGY `AS200`-at

```
ip as-path access-list 1 deny _(100|200)_
```

- `AS300` path-prepending eldobása (lásd később)

```
ip as-path access-list 1 deny _300_300_
```

Backup routing

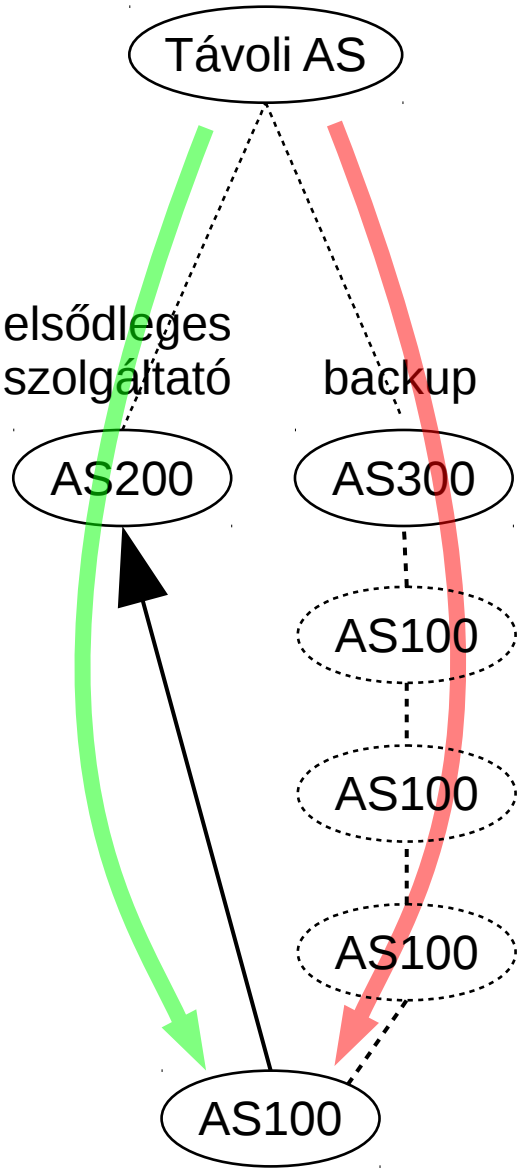
- Multi-homed AS-ek esetében gyakori felállás:
  - preferált elsődleges szolgáltató
  - másodlagos szolgáltató (**backup**)
  - csak ha az elsődleges szolgáltató elérhetetlen
- Kimenő forgalomra triviális: lokális preferencia beállításával
- De a bejövő forgalom irányát nagyon nehéz befolyásolni!

Backup routing

- **Cél:** a többi AS az **elsődleges szolgáltatón beérkező utat preferálja a backup helyett**
- Hogyan befolyásolja `AS100` azt, hogy egy Távoli AS milyen útvonalat választ hozzá??
- **Naív ötlet:** alaphelyzetben `AS100` az elsődleges szolgáltatón keresztül hirdeti a prefixeit, hiba esetén a backup-on küld hirdetést
  - meg kell várni, míg az új hirdetés elterjed az interneten
- Mindkét szolgáltatón kéne hirdetni és elérni, hogy befelé az elsődleges utat preferálják a távoli AS-ek

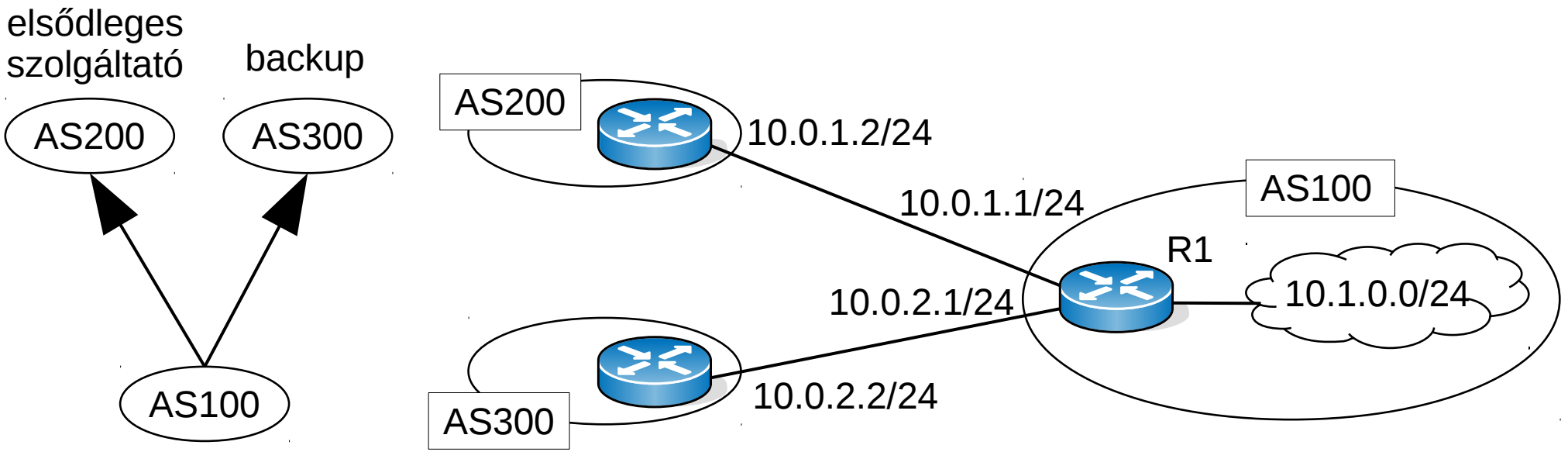
Backup: AS-path prepending

- Trükkös BGP konfiguráció: úgy teszünk, mintha a backup-on keresztül hosszabb lenne az út
- **AS-path prepending:** a backup-on keresztül küldött hirdetésbe sokszor beleírjuk a saját AS számunkat
  - hosszabbnak tűnik a backup út
  - ezért a Távoli AS kevésbé preferálja az aktív út kiválasztásakor



Backup: AS path prepending

- Legyen most `AS100` előfizetője `AS200` elsődleges szolgáltatónak és `AS300` backupnak
- `AS300` backup felé `AS100` „hazudik”: BGP-ben export szűrőkkel oldjuk meg, természetesen

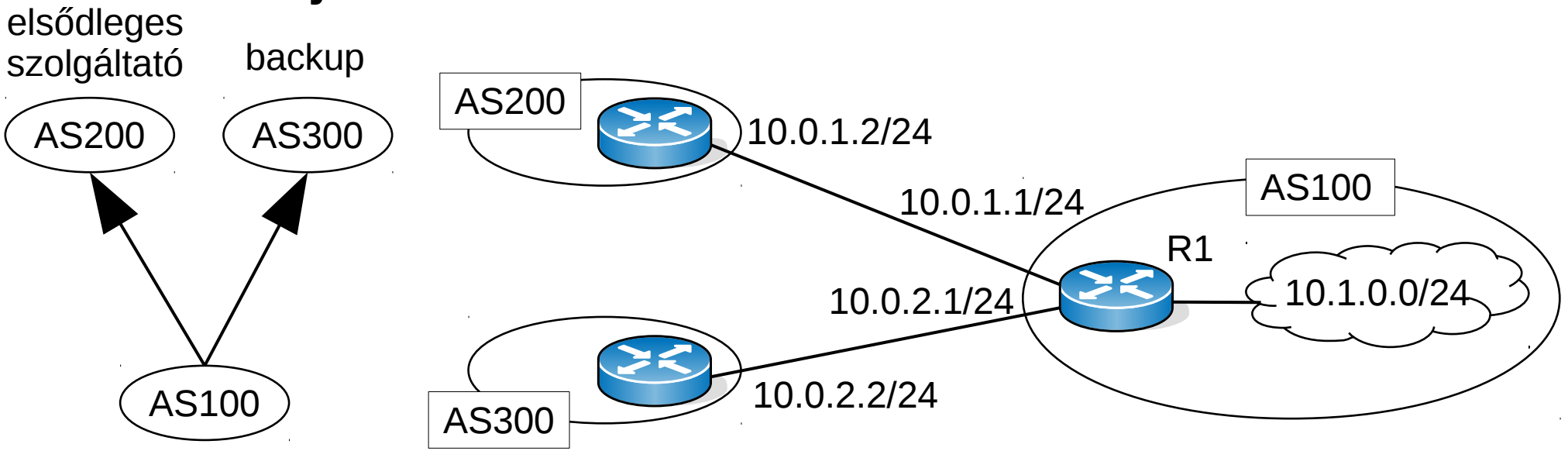


Backup: AS path prepending

- Ehhez `R1`-en gyártunk egy `route-map`-et

```
route-map rm-as-prepend permit 10
  set as-path prepend 100 100 100
```

- A `set as-path prepend` az AS-path elé beszúrja a kívánt hosszú `AS100` sorozatot

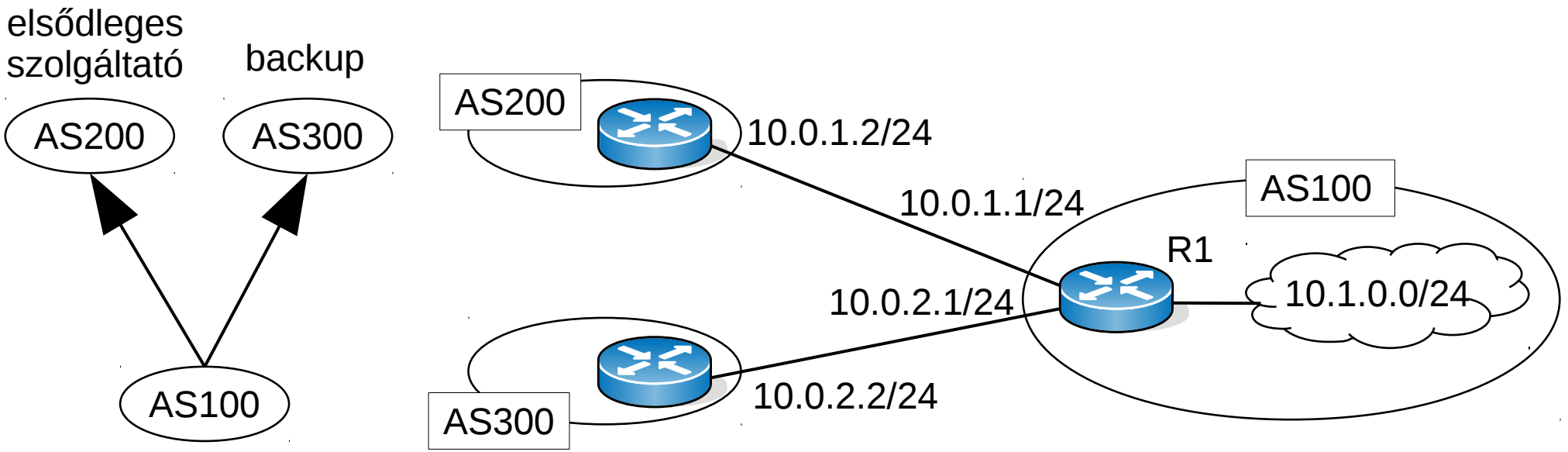


Backup: AS path prepending

- Az új `route-map`-et a `neighbor` paranccsal illesztjük a megfelelő szomszédhoz

  `neighbor 10.0.2.2 route-map rm-as-prepend out`

- Mivel export szűrőről van szó, az irány `out`



Backup: AS path prepending

```
router bgp 100
 bgp router-id 10.0.0.1
 neighbor 10.0.1.2 remote-as 200
 ...
 neighbor 10.0.2.2 remote-as 300
 neighbor 10.0.2.2 route-map rm-as-prepend out
 ...

!!! AS-path prepending szűrő
route-map rm-as-prepend permit 10
  set as-path prepend 100 100 100
```

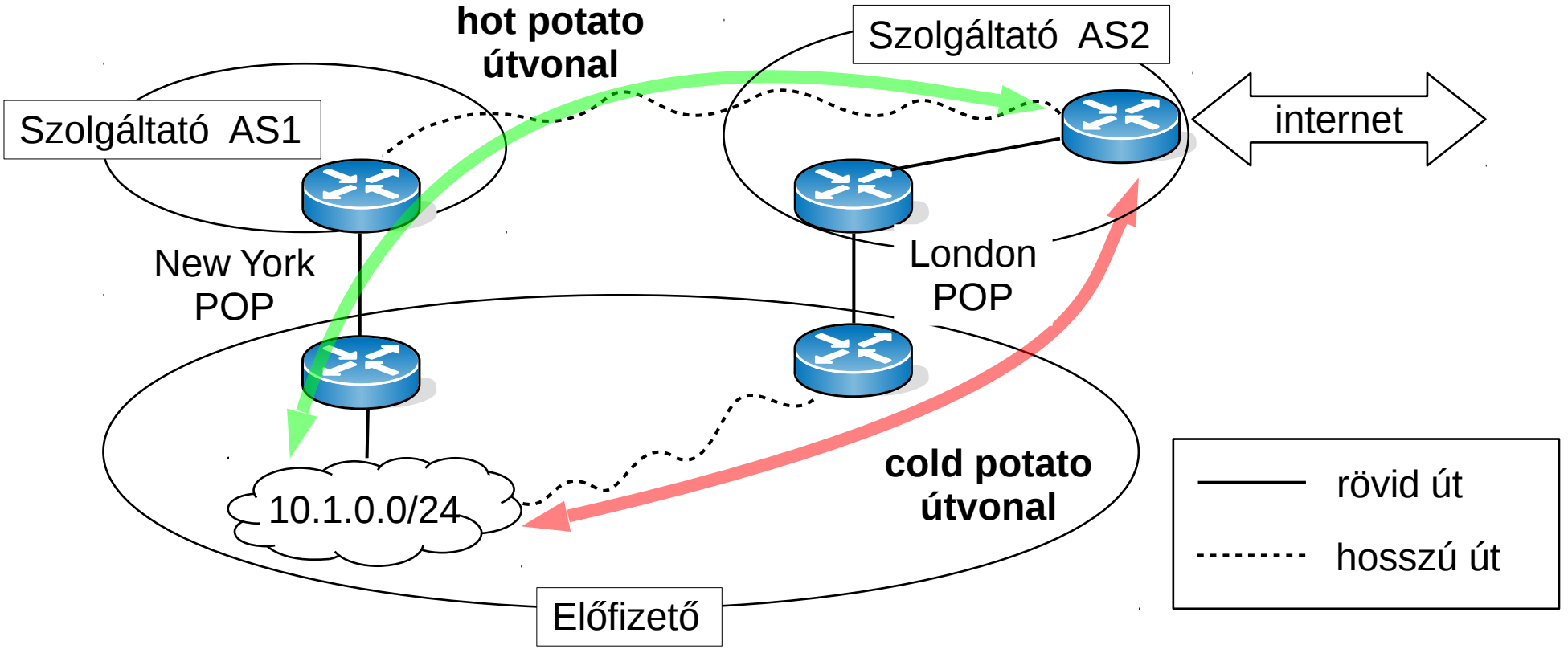
- Nyilván a korábbi esetekre definiált BGP szűrőkkel szabadon keverhető
- A szűrők sorrendjére vigyázni kell!

Backup: AS path prepending

- De `AS300` továbbra is a backup utat preferálja (a prefer-customer szabály miatt)
- Általánosságban a backup szolgáltató és annak összes előfizetője a backup-on route-ol: az AS path prepending nem mindig hatékony!
- **Megoldás:** jelezni kell a szolgáltatónak, hogy az út backup
  - „well-known” BGP community-k segítségével, jelentésükről a szomszéd ASek megállapodnak
  - a kimenő BGP hirdetésen beállítjuk a megfelelő community-t

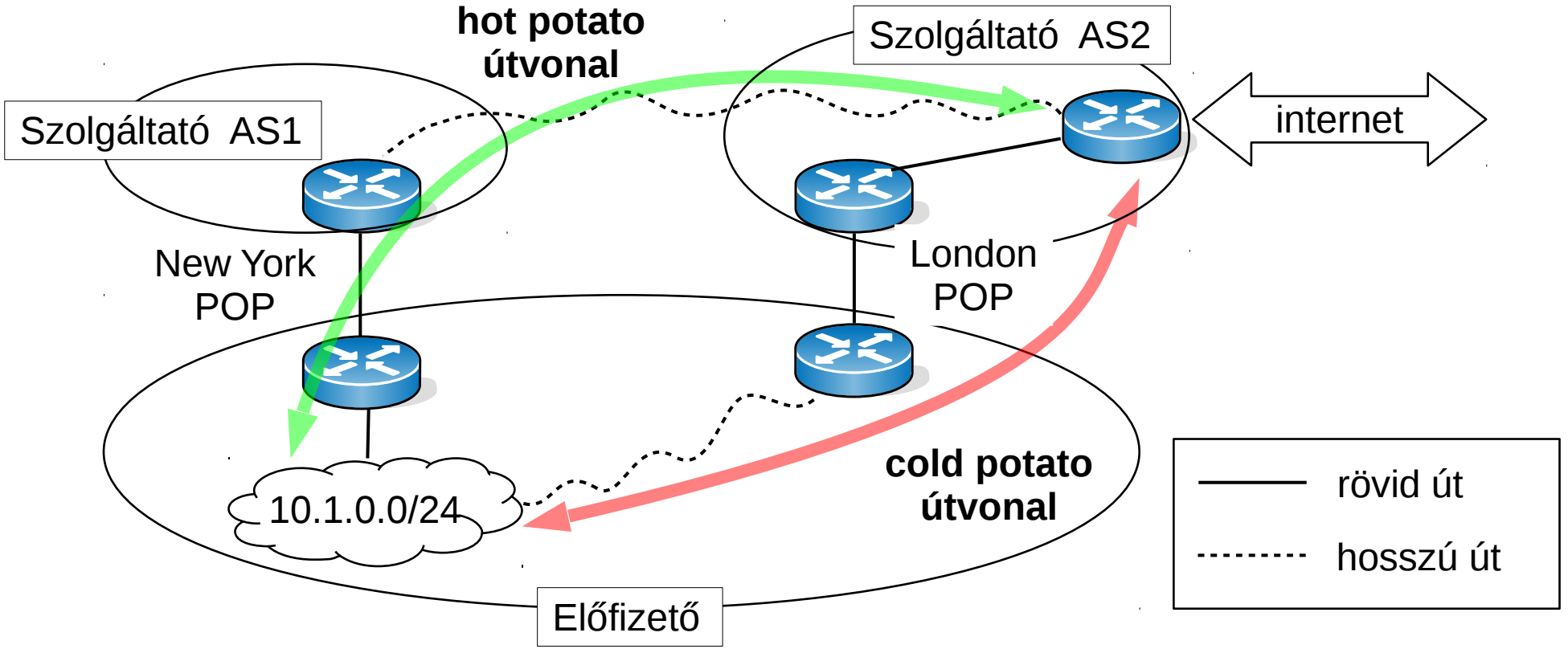
Hot potato routing

- **Hot potato routing:** a csomag a legrövidebb úton távozik az ASből
- A legkevesebb terhelést okozza **lokálisan**



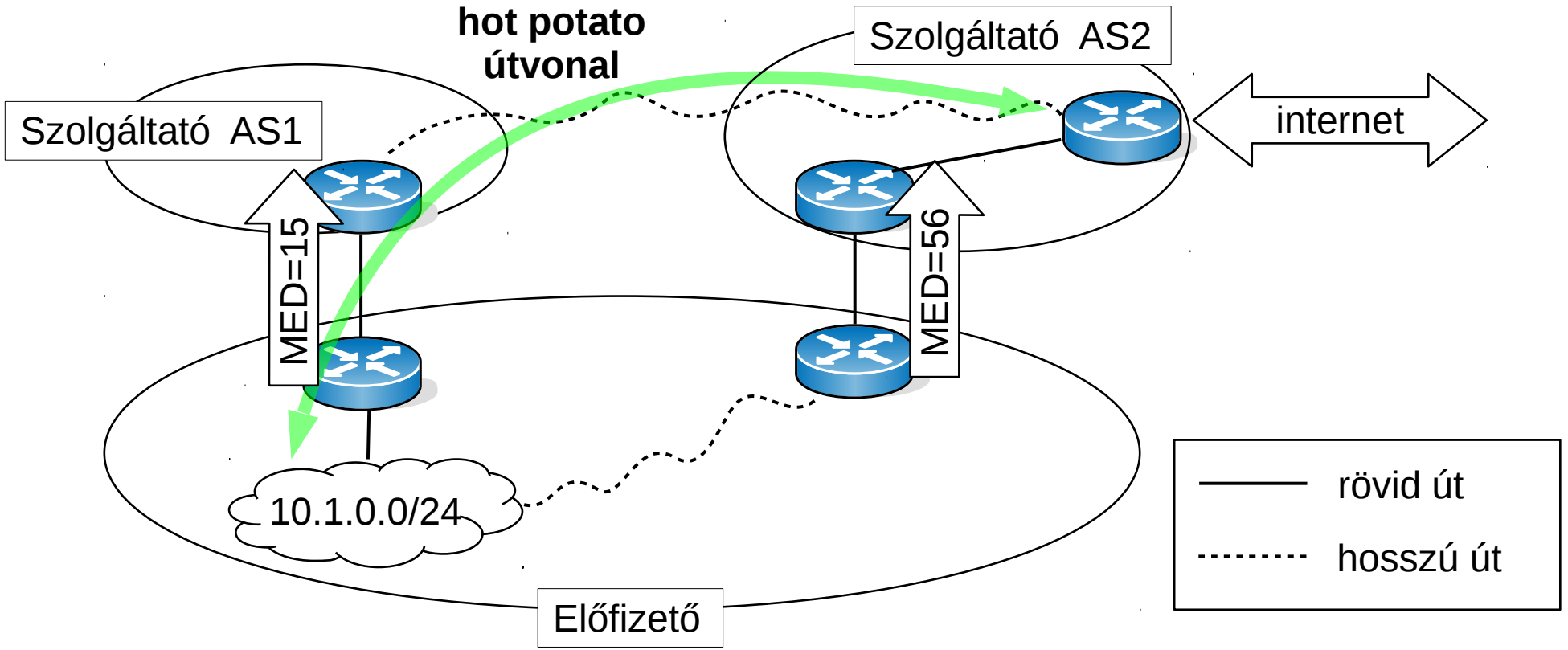
Hot potato routing

- Ami az előfizetőnek hot potato, a szolgáltatónak gyakran cold potato útvonal
- Előzetes megállapodás, ki vállalja a költséget



Hot potato routing

- Kimenő forgalom: iBGP és lokális preferencia
- Bejövő forgalom: Multi-Exit Discriminator (MED) BGP attribútum (alacsonyabb MED preferált)

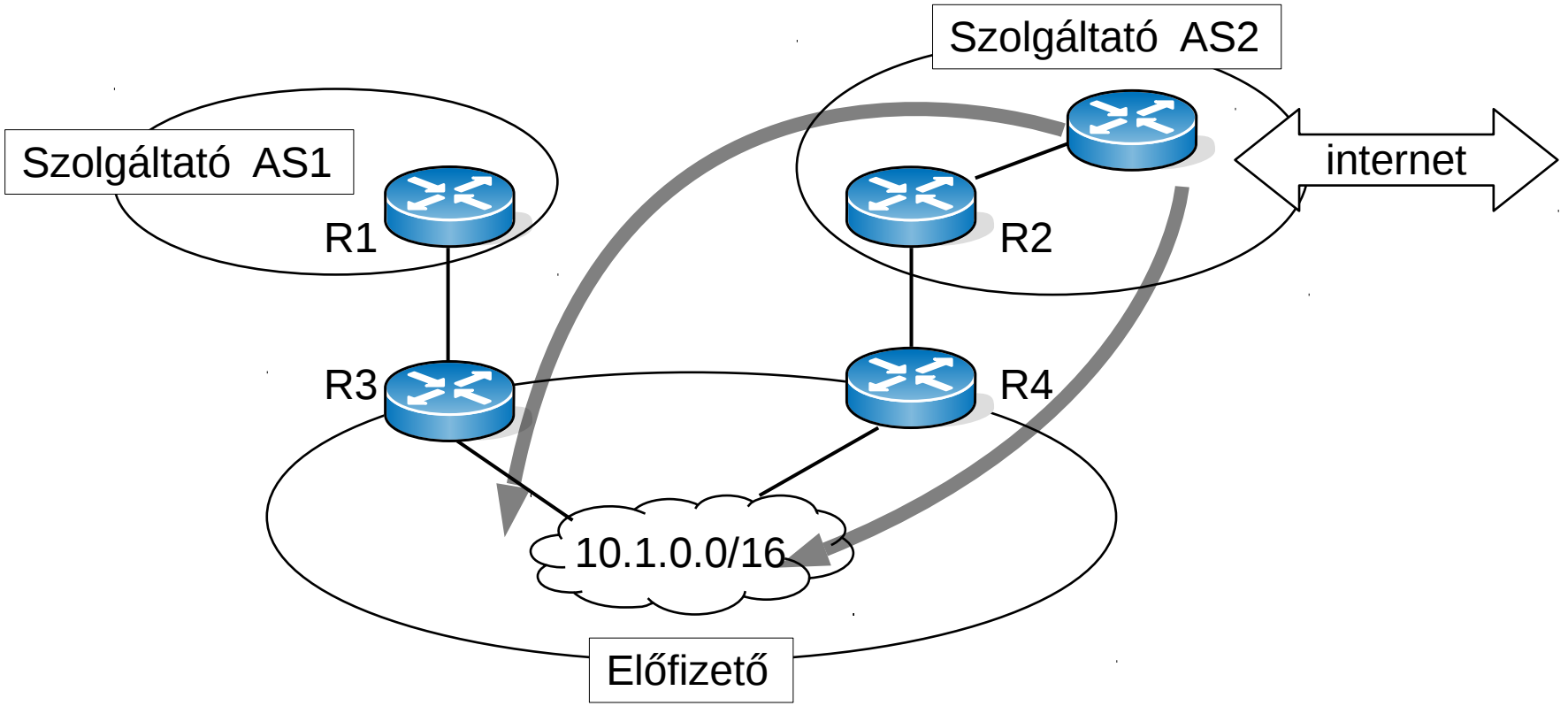


Forgalommenedzsment

- A backup egyfajta ellentéte: a forgalmi terhelés egyenlő megosztása egy multi-homed AS szolgáltatói között
- A kimenő forgalmat a BGP és a lokális preferenciák segítségével szinte tetszőlegesen megoszthatja a szolgáltatók felé
- A bejövő irány megint a nehezebb
- **Ingress Traffic Engineering (TE):** távoli ASek útválasztási döntéseinek befolyásolása úgy, hogy az egyes szolgáltatókon bejövő forgalom minél inkább kiegyenlítődjön

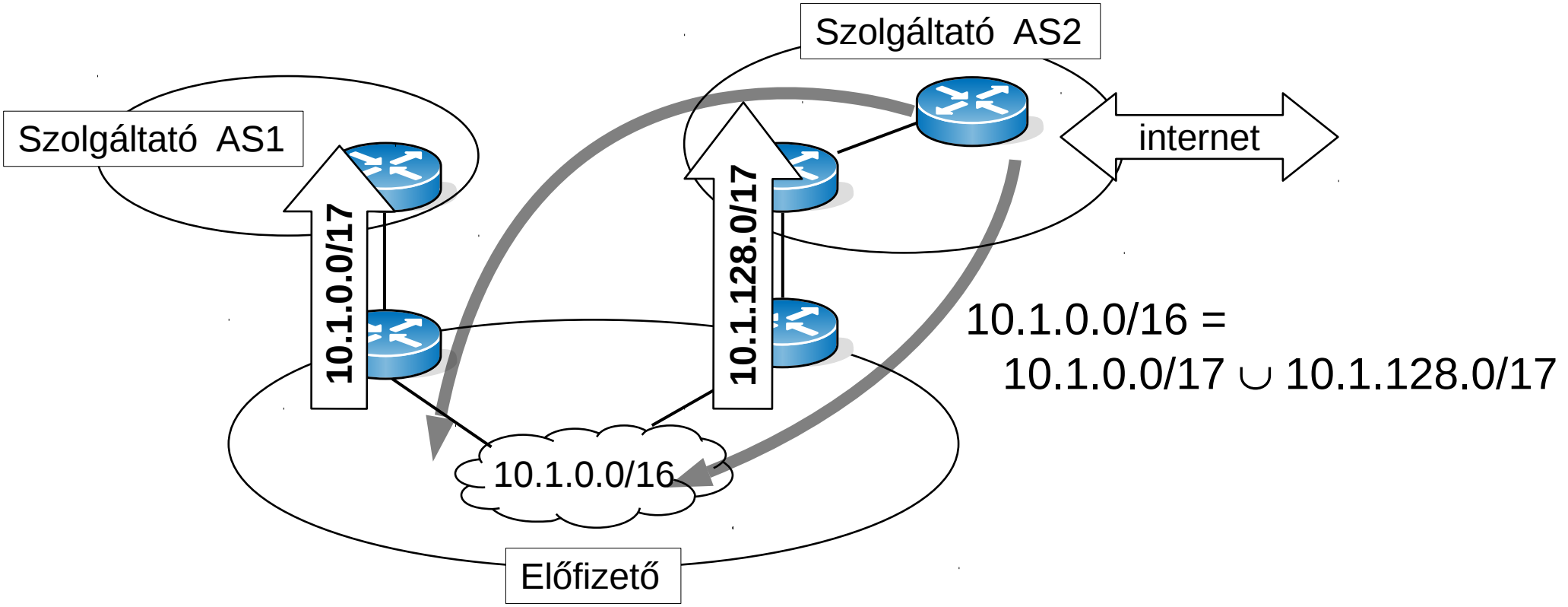
Forgalommenedzsment

- Az előfizető AS célja, hogy az `R1-R3` és az `R2-R4` linkek közt „nagyjából” egyenlően osztódjon meg a bejövő forgalma



Forgalommenedzsment

- **Deaggregáció:** az előfizető két egyenlő részre darabolja a címtartományát, egyiket `R3–R1` linken, másikat `R4–R2` linken hirdeti meg



Forgalommenedzsment

- A `10.1.0.0/17` és a `10.1.128.0/17` alhálózatokban ugyanannyi IP cím van
- Ha ezek ugyanannyi bejövő forgalmat „vonzanak”, akkor egyenlő terhelésmegosztás:
  - `10.1.0.0/17` forgalma `R1`-en lép be
  - `10.1.128.0/17` forgalma `R2`-n lép be
- **De a deaggregáció növeli minden BGP router FIBjét** (egy helyett két prefix)!
  - a hirdetések 5–10%-a deaggregált prefix
  - internet skálázhatatlanság egyik fontos oka

Az internet útvonalválasztás stabilitása

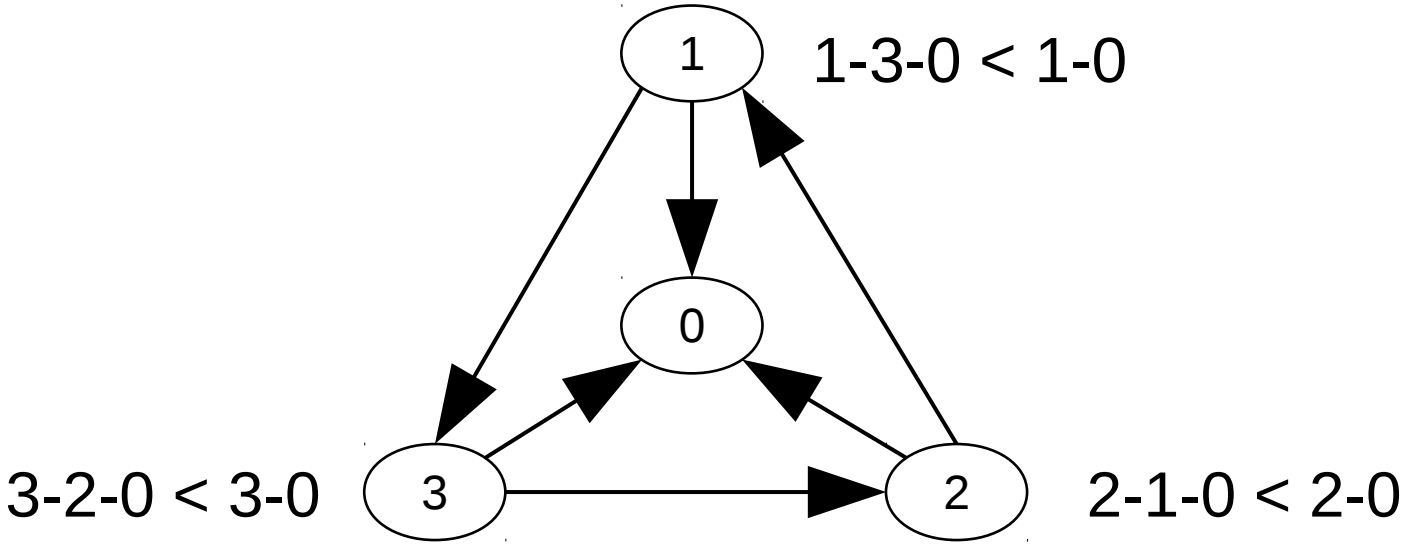
AS-szintű útválasztás: stabilitás

- A BGP-vel minden AS egymástól függetlenül megvalósíthatja a saját útválasztási stratégiáját
- De mi garantálja, hogy a függetlenül beállított policy-k végül egyértelmű és „jó” AS-szintű útvonalakhoz konvergáljanak?
- A válasz: semmi, nincs ilyen garancia
- **BGP oszcilláció:** az egyes BGP routerek nem tudnak megállapodni az útvonalakról
  - az ASek ellentmondó preferenciái miatt
  - instabilitás: az utak folyamatosan frissülnek

BGP oszcilláció

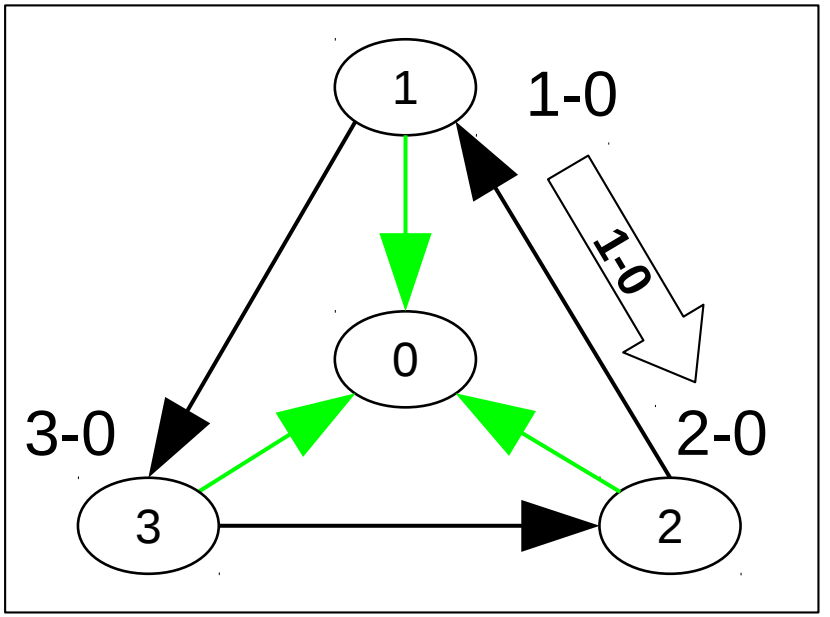
- Az alábbi példa-topológiában
  - a pontok az ASek és az élek tranzit linkek
  - cél AS a nullás azonosítójú pont
  - „<” reláció az útvonal-preferencia irányát jelzi

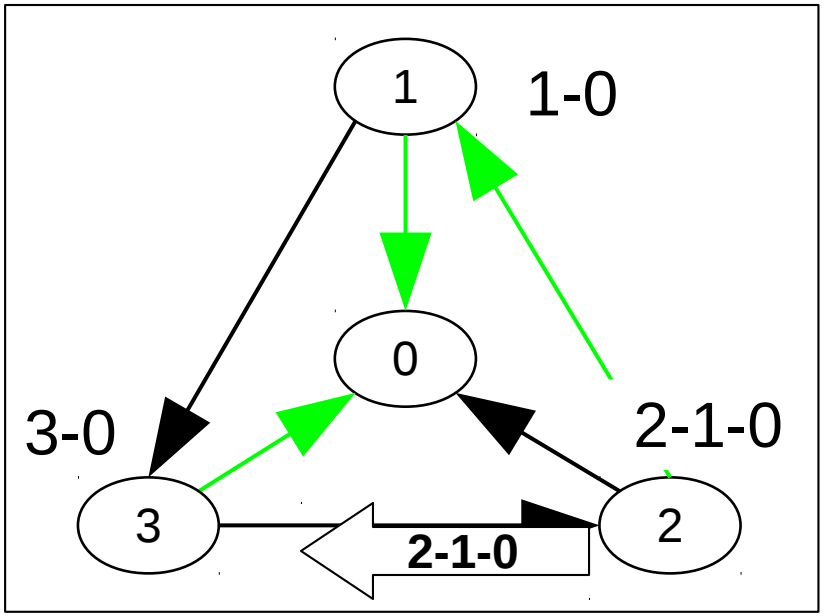
    preferált út < kevéssé preferált út < …



BGP oszcilláció

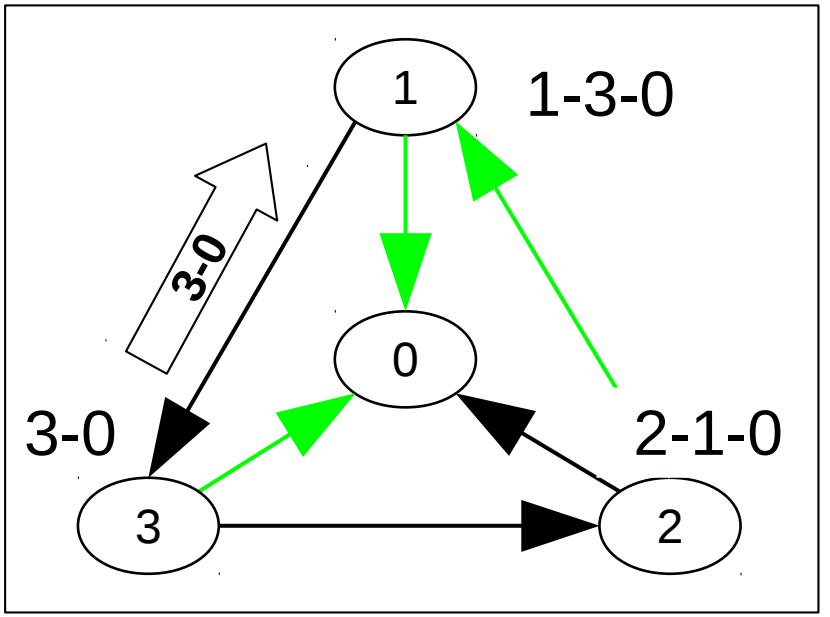
- Kezdetben *1*, *2* és *3* a közvetlen utat választja *0* irányában
- Tegyük fel, hogy *1* küld először hirdetést az *1-0* aktív útról
- Amint *2* megkapja, átvált a preferált *2-1-0* útra (*2-1-0* < *2-0*)
- *2* értesíti a *2-1-0* aktív útról *3*-at

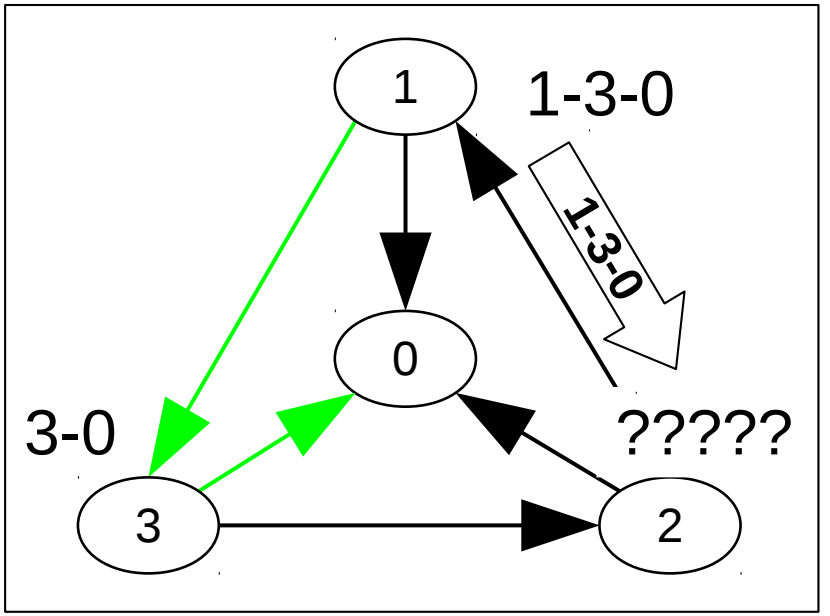




BGP oszcilláció

- Mivel *3* számára *3-2-0* út nem elérhető (*2* nem hirdeti a *2-0* utat), ezért marad a *3-0* útnál
- Értesíti *1*-et, amely a *1-3-0* > *1-0* preferencia miatt átvált *1-3-0* útra
- *2* elveszti a *2-1-0* utat (*1* nem hirdeti *1-0* utat), ezért a *2-0* útra vált
- És így tovább...





BGP oszcilláció elkerülése

- Oszcilláció alatt a BGP folyamatosan frissíti a FIBet és jelentős jelzési forgalmat generál
- Szintén oszcillációhoz vezethet a **flapping interfészek** esete
  - egy interfész másodpercenként többször resetel (például HW hiba vagy SW bug miatt)
  - minden fel→le→fel állapotváltozásról BGP üzenet megy az egész internet felé
- BGP Route Flap Damping [RFC 2439]: ismételt BGP Update üzenetek késleltetése (30 sec)