

Cursul #5

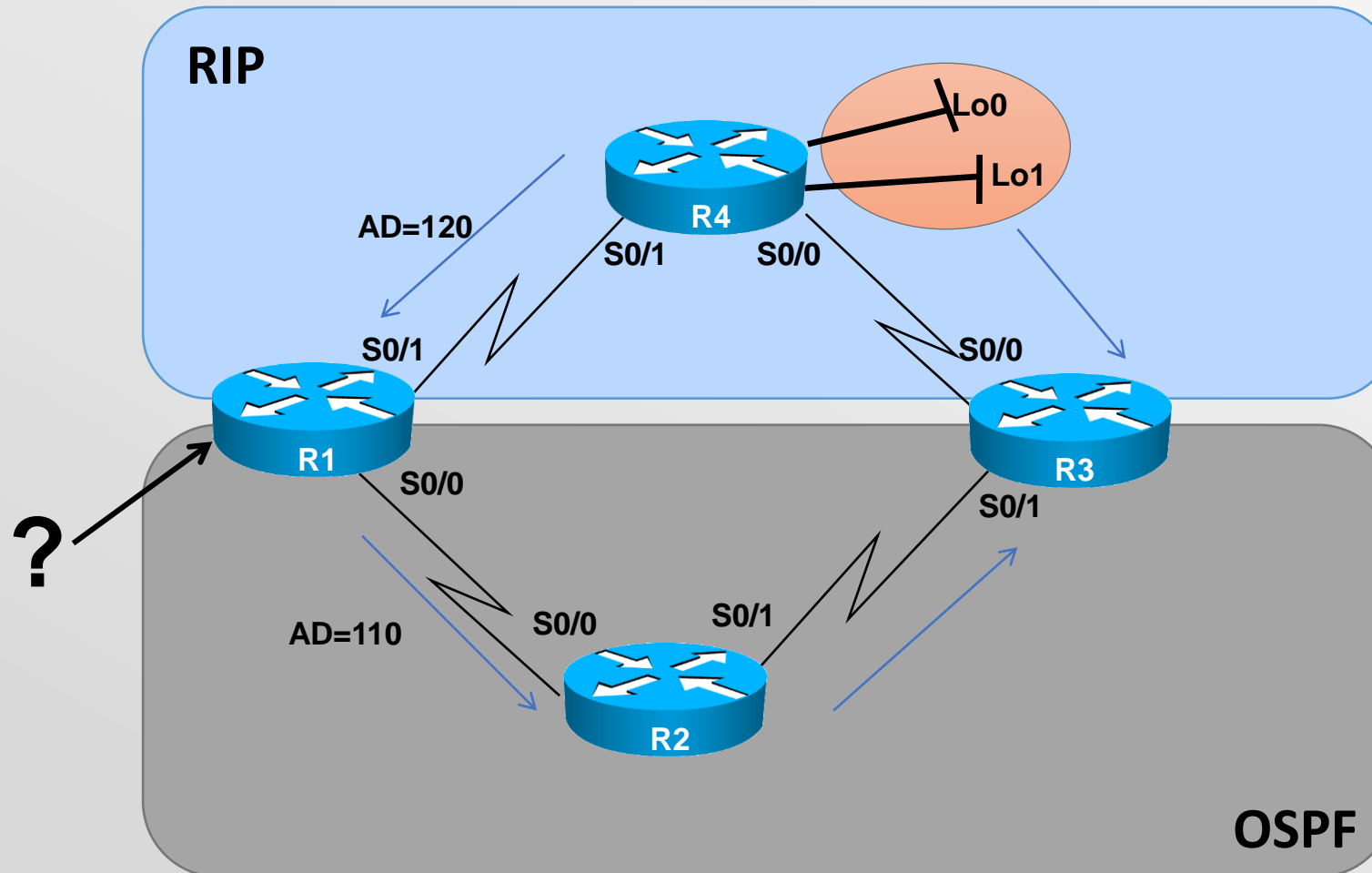
Optimizarea rutării



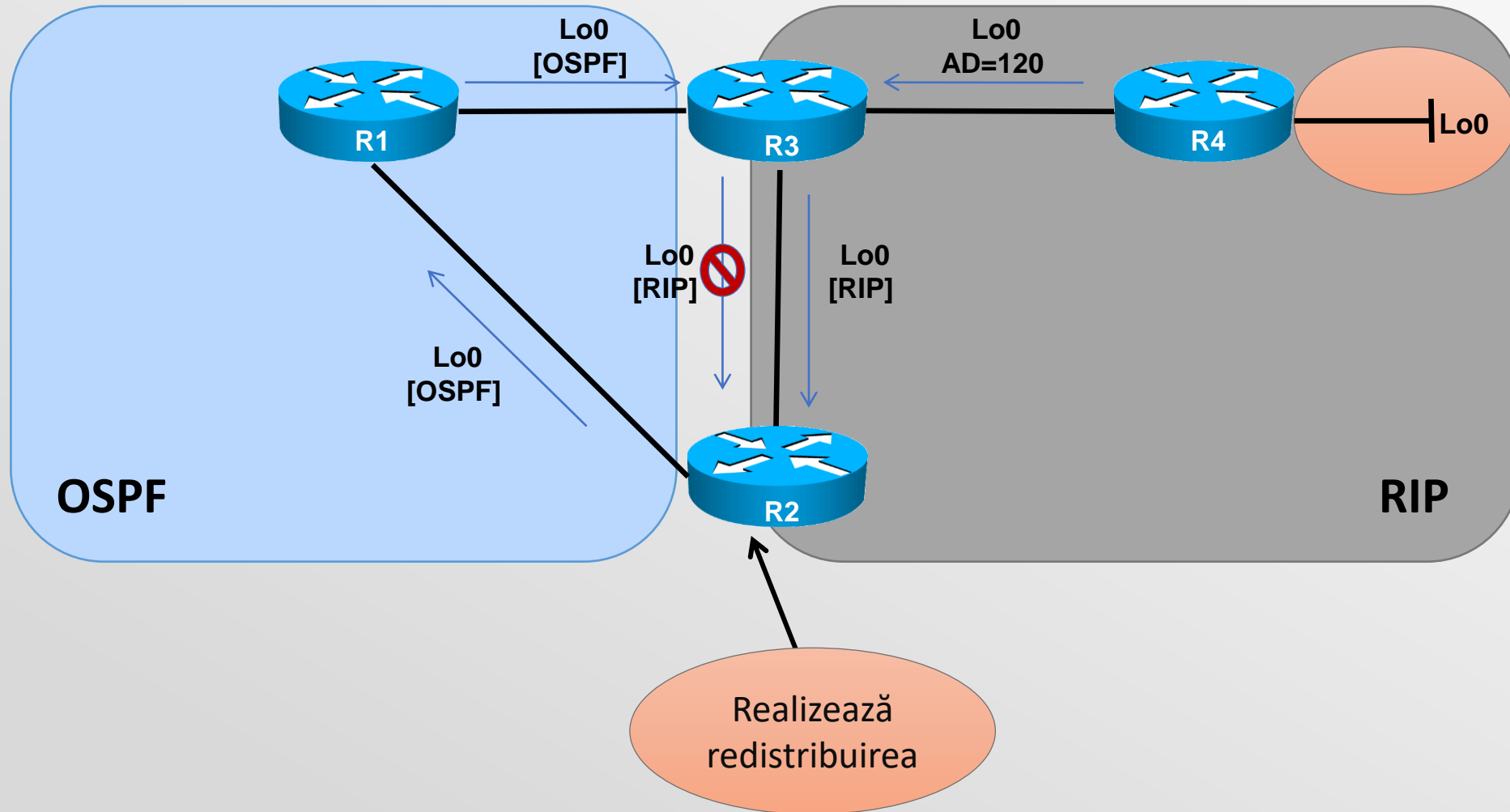
Controlul distribuției rutelor



Exemplu 1 - PoC



Exemplu 2



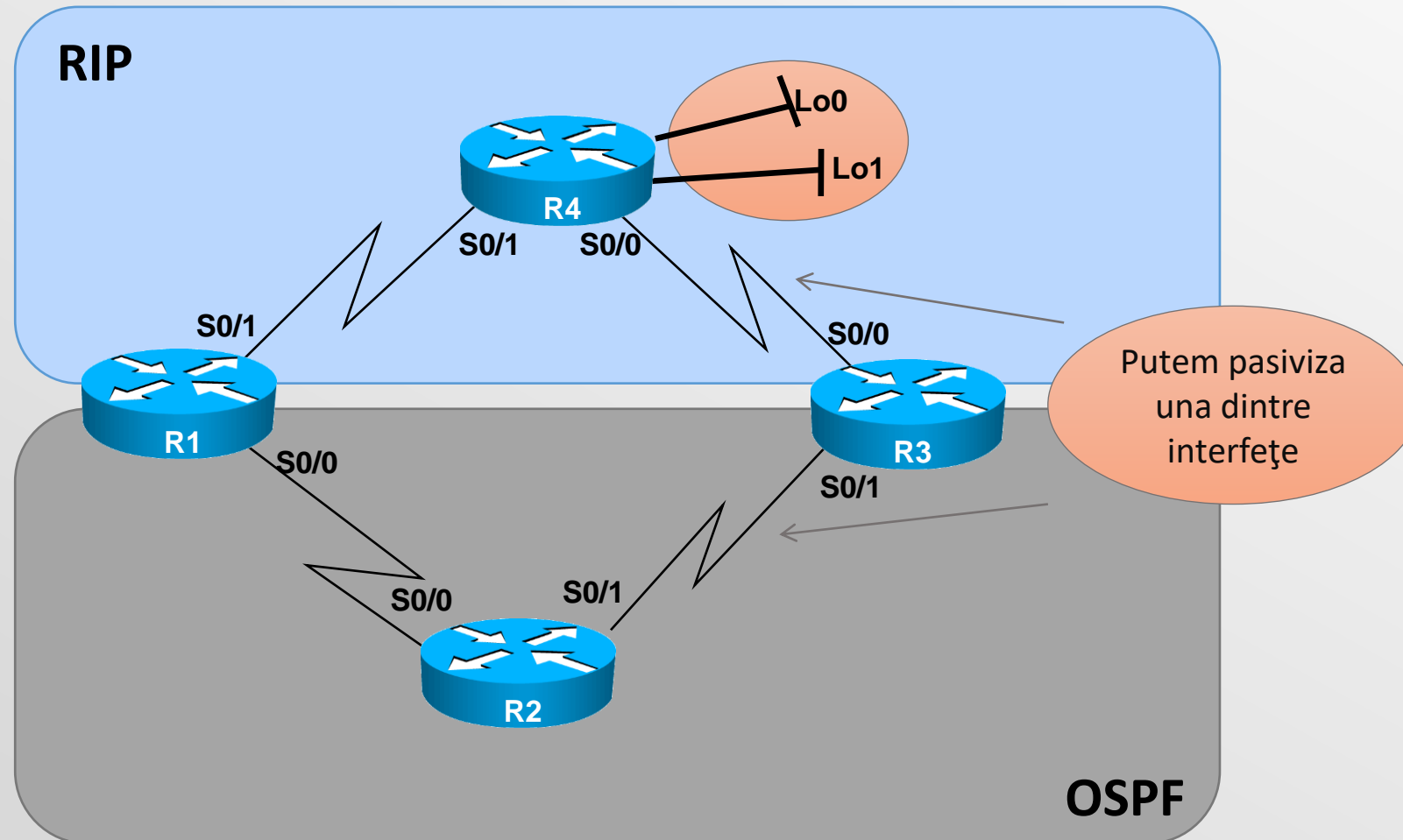
Pasivizarea interfețelor

- Din modul de configurare al protocolului de rutare:

```
passive-interface [default] {interface-type interface-number}
```

| Protocol | Efect |
|----------|---|
| RIP | Actualizările sunt primite - nu sunt trimise |
| EIGRP | Nu mai sunt trimise pachete Hello |
| OSPF | Nu mai sunt trimise pachete Hello |
| IS-IS | Nu mai sunt trimise pachete de Hello, dar sunt trimise actualizări automate despre rețeaua interfeței |

Exemplu - PoC



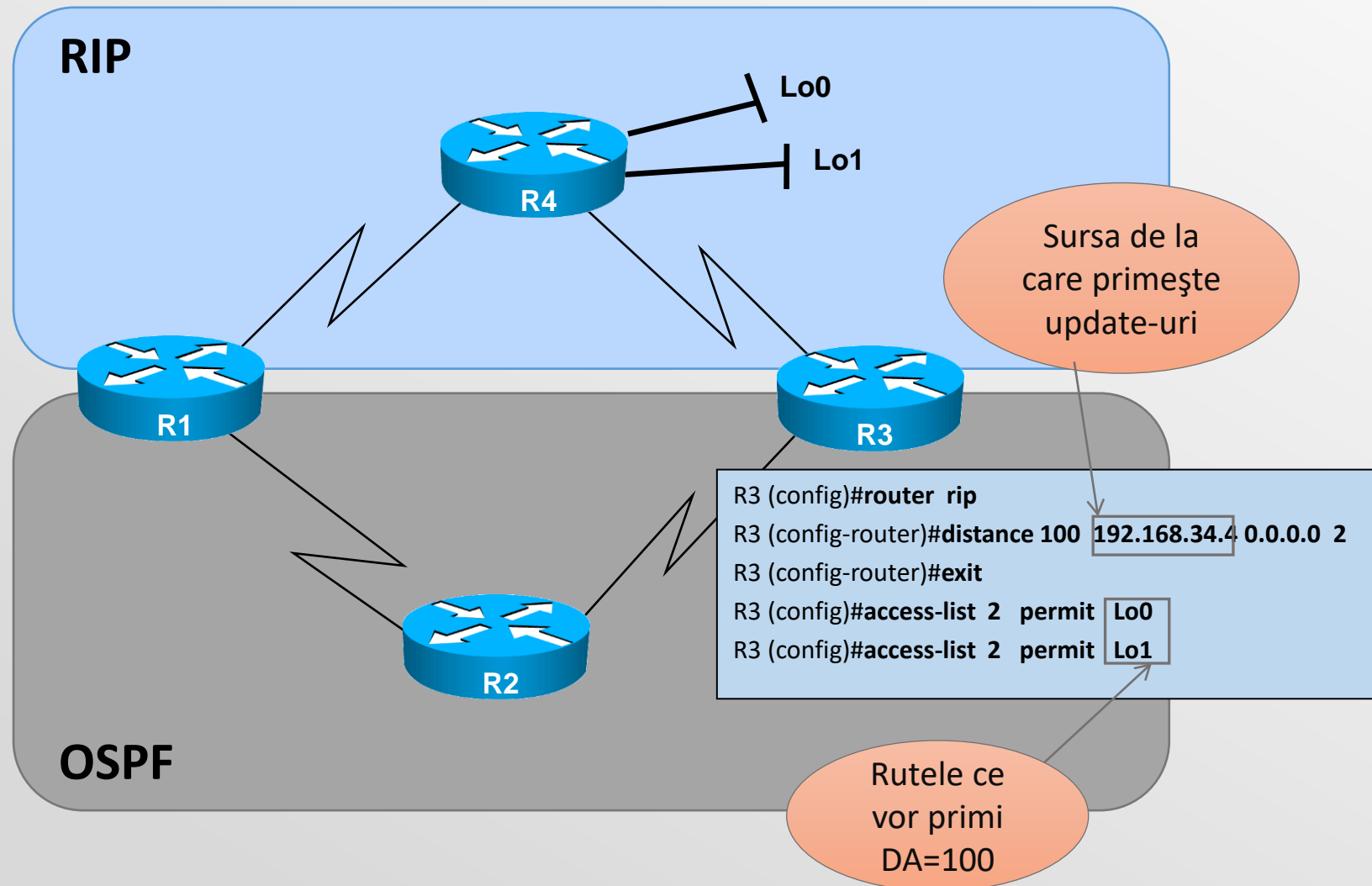
Distanțe administrative

- Din modul de configurare al protocolului de rutare

```
distance <value>
distance ospf {[intra-area <value>] [inter-area <value>] [external <value>]}
```

| Tipul rutei | Distanța administrativă |
|------------------|-------------------------------|
| Connected | 0 |
| Static | 0 (interfață) / 1 (adresă IP) |
| EIGRP summary | 5 |
| EIGRP (internal) | 90 |
| OSPF | 110 |
| IS-IS | 115 |
| RIP | 120 |
| EIGRP (external) | 170 |
| iBGP | 200 |

Exemplu - PoC



Liste distribuite

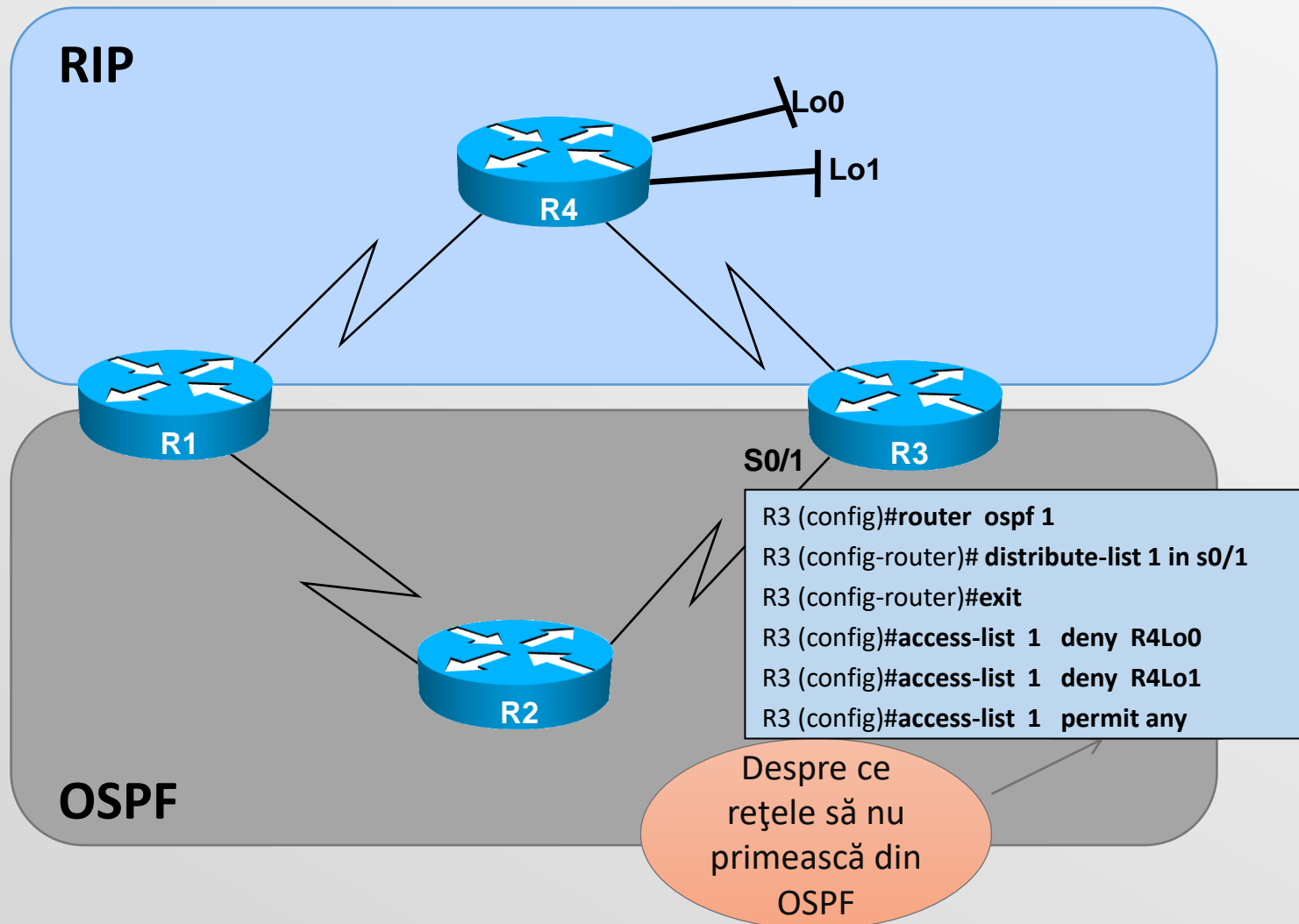
- Filtrează numai update-urile de rutare, nu și pachetele trimise

```

distribute-list {access-list-number | name} {in | out} [interface-type interface-number]
distribute-list prefix prefix_list_name {in | out} [interface-type interface-number]
distribute-list route-map route_name {in | out}
    
```

| Protocol | Efect |
|----------|---|
| RIP | Filtrează actualizările trimise/primate |
| EIGRP | Filtrează actualizările trimise/primate din tabela de topologie |
| OSPF | Filtrează rutele ce vor intra în tabela de rutare |
| IS-IS | Nu este suportat. |

Exemplu - PoC



Dezavantaje „distribute-list”

- În primul rând...
 - ... filtrarea de rute se poate aplica în orice situație; nu doar în problema rutării suboptimale
- Dezavantaje “aparente”?
 - Nescalabilă: depinde de ACL-uri
- Optimizarea distribute-list
 - Folosind tehnici de **route tagging** (nu se mai folosesc ACL-uri decât la identificarea inițială a traficului)

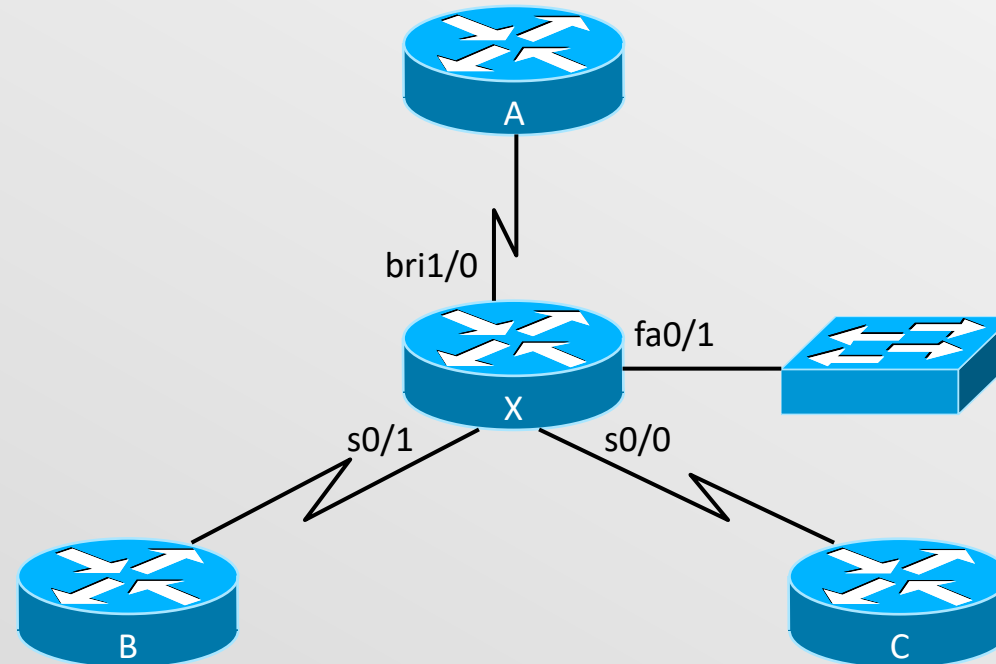
Mecanisme de optimizare a rutării



Mecanisme generice de optimizare

1. Pasivizarea interfețelor
2. Alterarea distanțelor administrative
3. Liste de distribuție
4. Route-maps

Interfețe pasive - configurare



```
X(config)#router rip
X(config-router)#network 10.0.0.0
X(config-router)#passive-interface fa0/1
```

Pasivizarea interfețelor

- Din modul de configurare al protocolului de rutare:

```
passive-interface [default] {interface-type interface-number}
```

| Protocol | Efect |
|--------------|---|
| RIP | Actualizările sunt primite - nu sunt trimise |
| EIGRP | Nu mai sunt trimise pachete Hello |
| OSPF | Nu mai sunt trimise pachete Hello |
| ISIS | Nu mai sunt trimise pachete de Hello, dar sunt trimise actualizări automate despre rețeaua interfeței |

Distanțe administrative

- Din modul de configurare al protocolului de rutare

```
distance <value>
distance ospf {[intra-area <value>] [inter-area <value>] [external <value>]}
```

| Tipul rutei | Distanța administrativă |
|------------------|-------------------------------|
| Connected | 0 |
| Static | 0 (interfață) / 1 (adresă IP) |
| EIGRP summary | 5 |
| EIGRP (internal) | 90 |
| OSPF | 110 |
| IS-IS | 115 |
| RIP | 120 |
| EIGRP (external) | 170 |
| iBGP | 200 |

Configurarea distanței administrative

```
Router(config-router) #distance weight [source-ip-address  
source-mask (access-list-number | name)]
```

```
RTZ (config) #router rip
```

```
RTZ (config-router) #distance 105 10.4.0.2 255.255.255.0
```

- Rutele invatate prin RIP vor avea o distanta administrativa diferita de cea default (105, in acest caz)
- Schimbarea are doar **semnificatie locală**, toate celelalte rutere pastrand distanta administrativa de 120
- In exemplu, toate rutele invata prin RIP de la 10.4.0.2 vor primi local d.a. 105

Modificarea distanței administrative

```
RTZ (config) #router rip
RTZ (config-router) #distance 97 10.3.0.1 255.255.255.0 (2)
RTZ (config-router) #exit
RTZ (config) #access-list (2) permit 192.168.3.0 0.0.0.255
```

Sursa de la care se primesc
updateuri RIP

Rutele care vor primi d.a. 97

- Se specifica rutele care vor primi o anumita d.a
- In exemplu, doar rutele spre 193.168.3.0/24, invatate de la 10.3.0.1

Dezavantaje „distance”

- Greu de urmărit în configurații complexe
- Nu e o soluție scalabilă: se bazează pe intrări în ACL-uri
- Modificarea este locală
 - Distanța administrativă nouă nu este comunicată altor rutere

Liste de distribuție

- Filtrează numai update-urile de rutare, nu și pachetele trimise

```
distribute-list {access-list-number | name} {in | out} [interface-type interface-number]
distribute-list prefix prefix_list_name {in | out} [interface-type interface-number]
distribute-list route-map route_name {in | out}
```

| Protocol | Efect |
|----------|---|
| RIP | Filtrează actualizările trimise/primate |
| EIGRP | Filtrează actualizările trimise/primate din tabela de topologie |
| OSPF | Filtrează rutele ce vor intra în tabela de rutare |
| IS-IS | Nu este suportat. |

Dezavantaje „distribute-list”

- În primul rând...
 - ... filtrarea de rute se poate aplica în orice situație; nu doar în problema rutării suboptimale
- Dezavantaje “aparente”?
 - Nescalabilă: depinde de ACL-uri
- Optimizarea distribute-list
 - Folosind tehnici de **route tagging** (nu se mai folosesc ACL-uri decât la identificarea inițială a traficului)

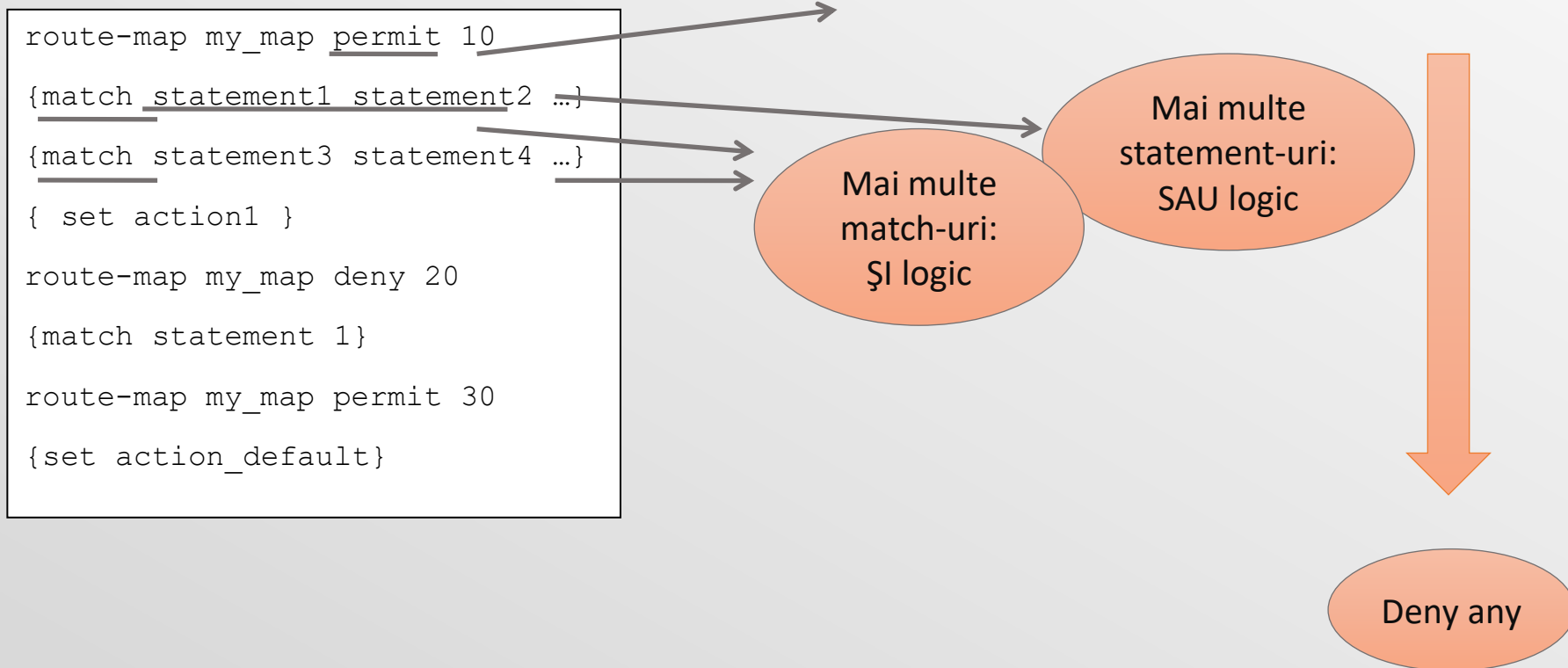
Route-maps



Route-maps

- Cel mai puternic mecanism de manipulare de rute
- Structură
 - Asemănător IF/THEN/ELSE în programare
 - **Acțiune globală** la nivelul fiecărei reguli (permit/deny)
 - Clauze **match** identifică traficul
 - Conform unui ACL
 - Conform protocolului de rutare
 - Conform dimensiunii pachetului , etc.
 - Clauze **set** specifică acțiuni asupra pachetului identificat
 - Forțarea pachetului pe o anumită interfață (PBR)
 - Manipularea atributelor BGP
 - Metrica în protocolul de rutare

Parcurgerea unui route-map

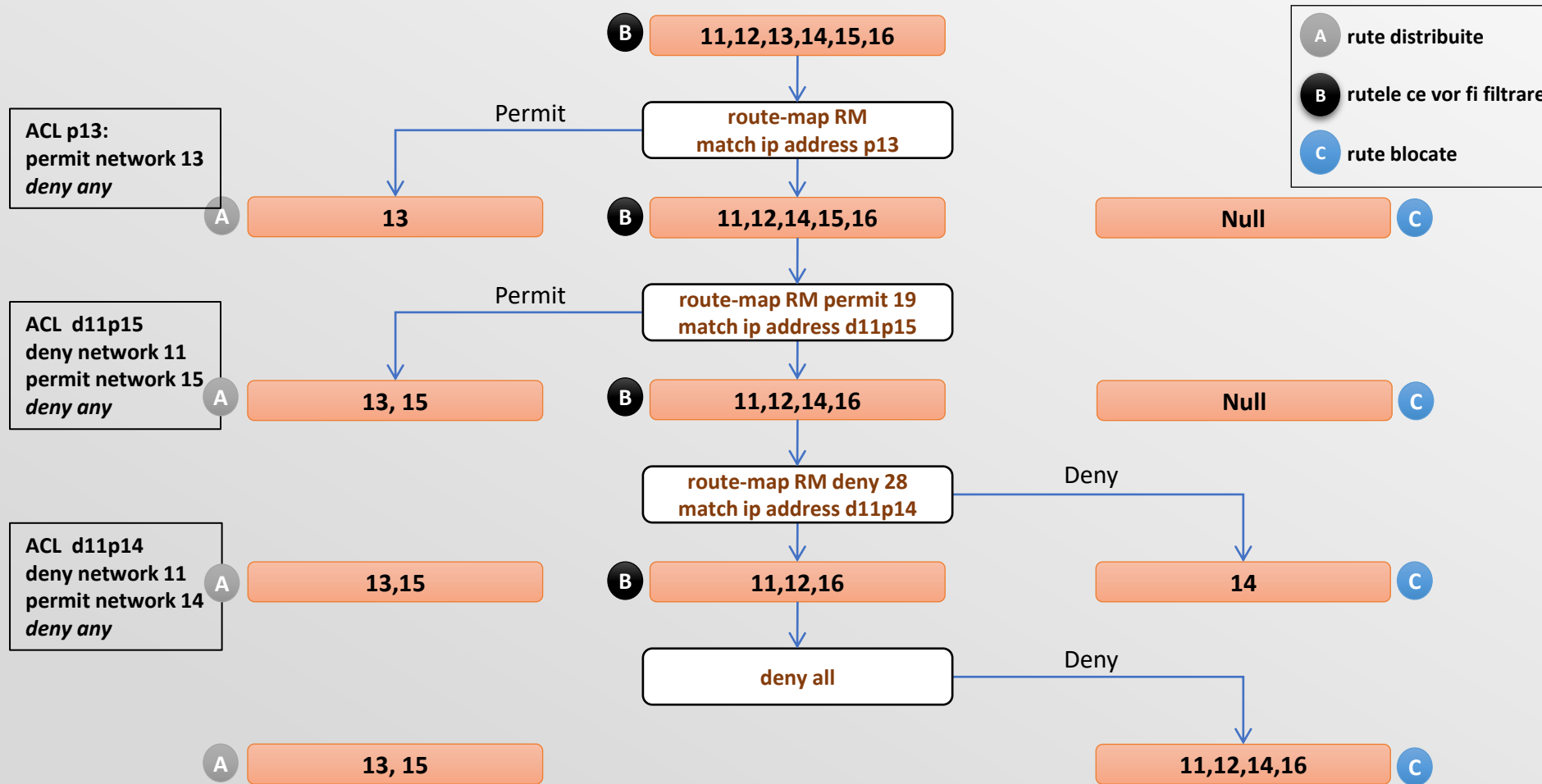


- Lipsa unei clauze match == match any

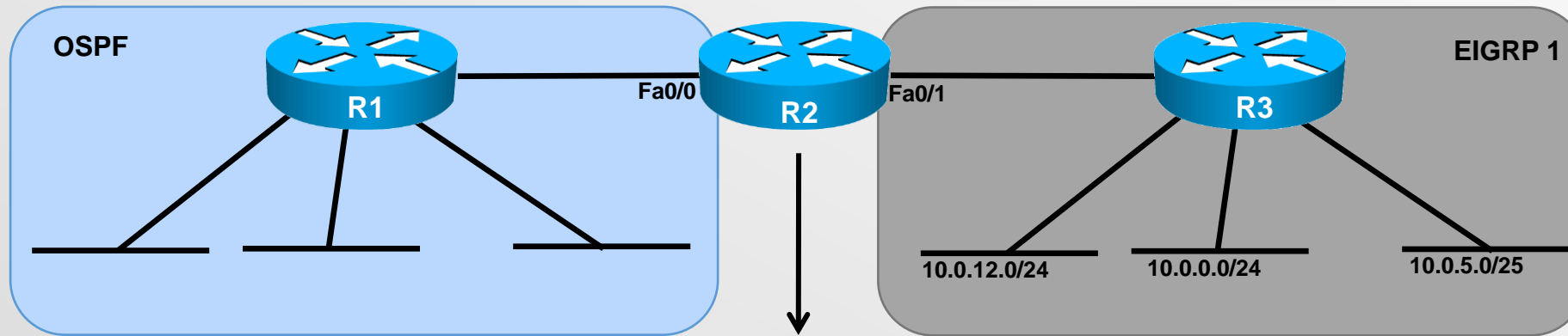
Comanda match

| Comanda | Descriere |
|------------------------------------|---|
| <code>match interface</code> | Interfața de ieșire a rutelor |
| <code>match ip address</code> | Folosește ACL și prefix-list |
| <code>match ip next-hop</code> | Adresa IP a următorului hop |
| <code>match ip route-source</code> | ACL pentru sursa ruterului care a trimis actualizarea |
| <code>match metric</code> | Metrica rutei |
| <code>match route-type</code> | Tipul rutei |
| <code>match tag</code> | Tag-ul pe care îl are ruta |

Selecția informațiilor de actualizare



Route-map în redistribuție



```
R2(config)# router ospf 1
R2(config-router)# redistribute eigrp 1 subnets route-map eigrp_to_ospf
R2(config)#route-map eigrp_to_ospf permit 10
R2(config-route-map)#match ip address eigrp_to_ospf
R2(config-route-map)#exit
R2(config-route-map)# do sh access-1 eigrp_to_ospf
Standard IP access list filter_isis
  10 permit 10.0.12.0, wildcard bits 0.0.0.255
  20 permit 10.0.0.0, wildcard bits 0.0.0.255
  30 deny any
```

- Doar rețelele permise în ACL vor face match pe regula 10 și vor fi redistribuite conform politicii globale ale regulii (permit).

Parcurgerea unui route-map

```
route-map my_map permit 10
{match statement1 statement2 ...}
{match statement3 statement4 ...}
{ set action1 }

route-map my_map deny 20
{match statement 1}

route-map my_map permit 30
{set action_default}
```

- Lipsa unei clauze match == match any

Folosirea set

- Modificarea atributelor BGP: ASP PATH, local_pref, weight
- Stabilirea metricii de redistribuție (ex în OSPF sau EIGRP)
- Stabilirea următorului hop în policy-based routing.

Comanda set

- Stabilește **următorul hop** către care să fie trimis pachetul:

```
Router(config-route-map)#set ip next-hop ip-address [... ip-address]
```

- Stabilește **interfața de ieșire** pe care să fie trimis pachetul:

```
Router(config-route-map)#set interface interface-type interface-number [... type number]
```

- Stabilește **următorul hop**, în cazul în care nu există o rută explicită către destinație:

```
Router(config-route-map)#set ip default next-hop ip-address [...ip-address]
```

- Stabilește **interfața de ieșire**, în cazul în care nu există o rută explicită către destinație:

```
Router(config-route-map)#set default interface interface-type interface-number [... type ...number]
```

Policy-based routing



Policy-based routing

- Suprascrierea deciziilor de rutare implicite
- De ce?
 - Rutarea tradițională este realizată **doar** pe baza adresei IP destinație
 - Singura modalitate de a stabili înainte calea unui pachet -> rutare statică (**ip route**)
 - Permite rutarea pe baza mai multor factori, nu numai a adresei destinație
 - Permite stabilirea de politici de rutare (în funcție de organizație sau aspecte de securitate)
- Implementare: route-maps

PBR facts

- Politicile de rutare se aplică la nivel de interfață
- Se poate aplica o singură politică pe o interfață
- Pentru a aplica o politică:

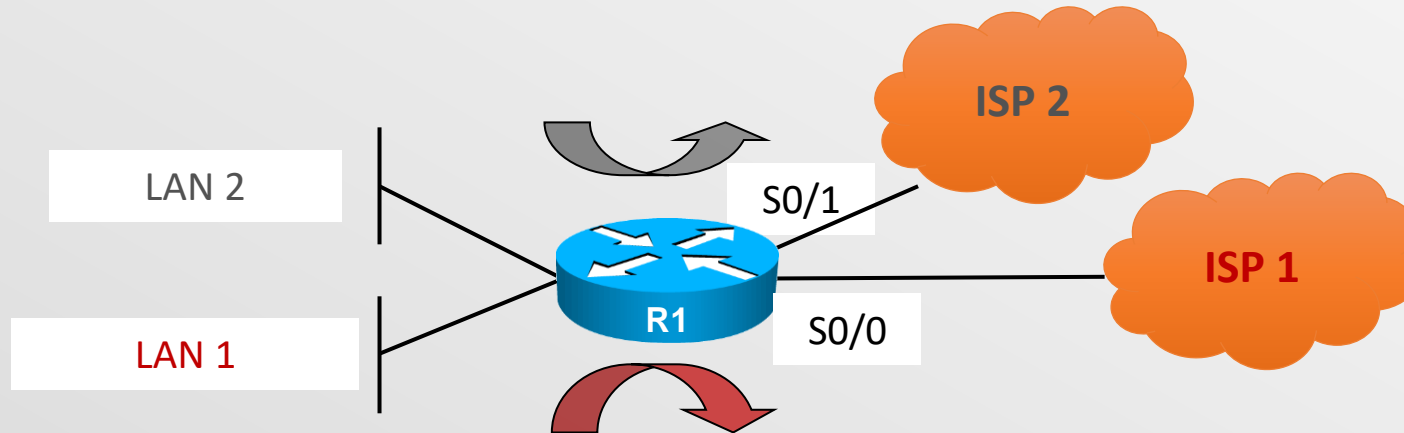
```
(config-if)# ip policy route-map <name>
```

- Pentru a afecta traficul generat de ruter:

```
(config)# ip local-policy route-map <name>
```

- Dacă un pachet nu face match pe nici o regulă de route-map, acesta este trimis în procesul de rutare normal

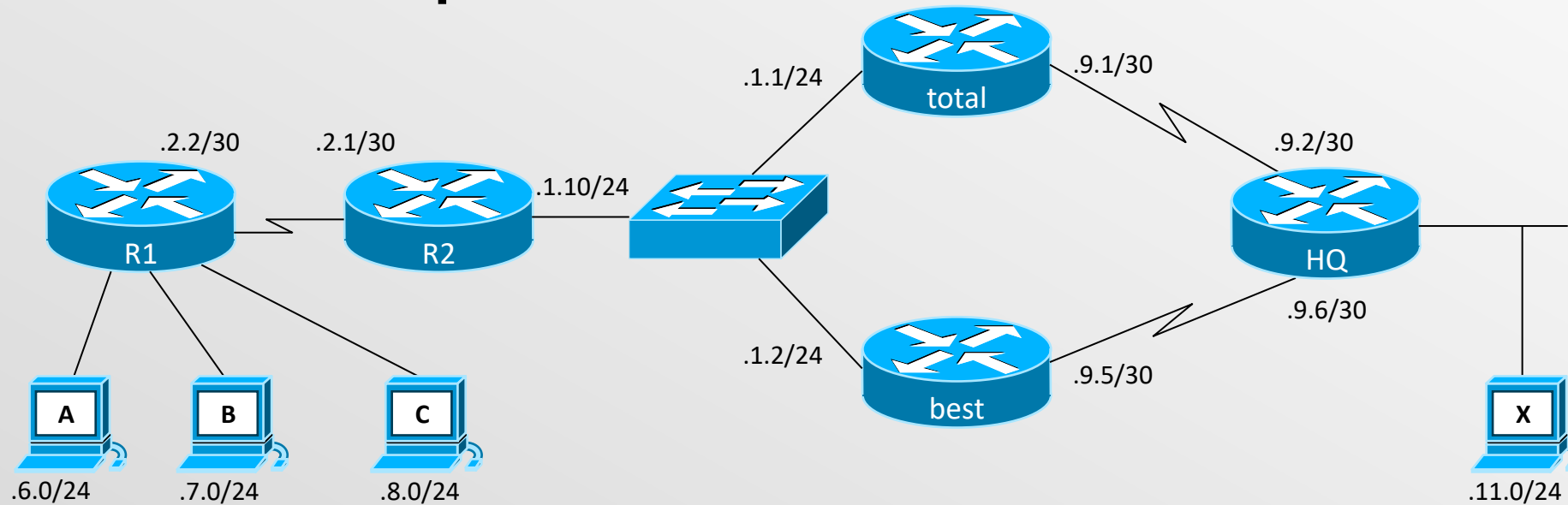
PBR – Exemplanu 1



```

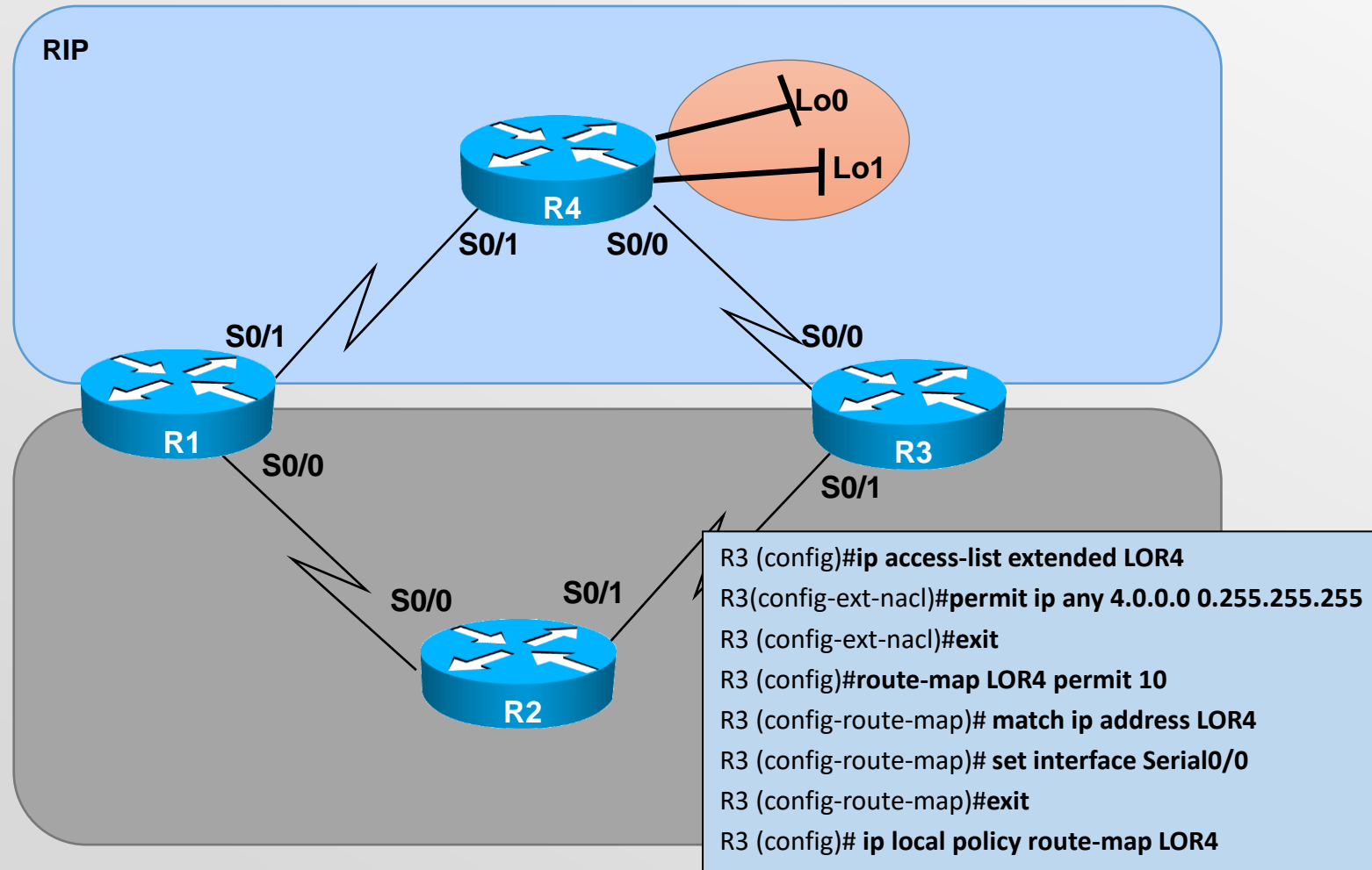
R1 (config) #interface e0/0
R1 (config-if) #ipv6 policy route-map ISP1
R1 (config) #interface e0/1
R1 (config-if) #ipv6 policy route-map ISP2
R1 (config) #route-map ISP1 permit 10
R1 (config-route-map) #match ipv6 address 1
R1 (config-route-map) #set interface s0/0
R1 (config) #route-map ISP2 permit 10
R1 (config-route-map) #match ipv6 address 2
R1 (config-route-map) #set interface s0/1
R1 (config) #access-list 1 permit LAN1
R1 (config) #access-list 2 permit LAN2
    
```

PBR – Exemplu 2



- Se cere ca:
 - traficul din 172.17.6.0/24 sa treaca doar prin ruterul “total”
 - traficul din 172.17.7.0/24 sa treaca doar prin ruterul “best”
 - traficul din 172.17.8.0/24 sa foloseasca ambele legaturi

PBR - PoC



Tunelare



Tunelare

- Ce este un tunel?
 - O legătură virtuală peste o rețea fizică
 - Încapsularea unui protocol în alt protocol
 - Ascunderea unei infrastructuri de rețea în spatele unei singure conexiuni



Tipuri de tunele

| | |
|-------------|---------------------------------------|
| Application | HTTP |
| Transport | SSL VPN |
| Network | IPIP, 6to4, SIT, IPSec, GRE |
| Data link | PPPoE, Q-in-Q |

Tunelare GRE

- Generic Routing Encapsulation
- Protocol de tunelare dezvoltat de Cisco
- Poate încapsula o varietate de protocoale de rețea
- Stateless – nu sunt menținute informații despre starea tunelului
- Un tunel GRE se ridică imediat după configurarea corectă a ambelor capete și rămâne ridicat tot timpul

Componentele unui tunel GRE

- Crearea interfeței tunel
 - Tip tunel (GRE)
 - Capăt sursă
 - Interfață sau IP local
 - Capăt destinație
 - IP la distanță
- Configurarea interfeței tunel
 - Interfața nou creată se tratează ca o legătură normală (punct la punct)
 - Adresare IP
 - Spațiu de adresă pentru domeniul tunelului

Configurare GRE

```
R2(config)# ip route 11.0.0.0 255.255.255.0 Fa0/0
R2(config)#interface Tunnel0
R2(config-if)# ip address 12.0.0.2 255.255.255.0
R2(config-if)# tunnel source FastEthernet0/0
R2(config-if)# tunnel destination 11.0.0.1
```

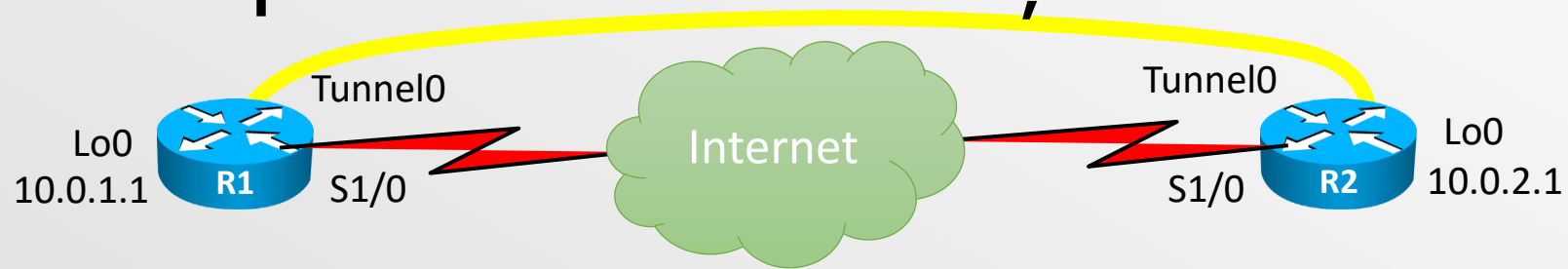


```
R1(config)# ip route 22.0.0.0 255.255.255.0 Fa0/0
R1(config)#interface Tunnel0
R1(config-if)# ip address 12.0.0.1 255.255.255.0
R1(config-if)# tunnel source FastEthernet0/0
R1(config-if)# tunnel destination 22.0.0.2
```

GRE Keepalive

- Tunelele GRE sunt **stateless** – fiecare capăt de tunel nu păstrează informații de stare despre capătul său remote
 - celălalt capăt al tunelului dispăre - ruterul nu poate determina acest lucru
- Mecanisme **keepalive**
 - Trimise de un device pe o interfață fizică sau virtuală
 - Informează un alt device din rețea că legătura dintre ele încă funcționează
 - **keepalive interval**
 - **keepalive retries**

GRE Keepalive – Funcționare



► Mesaj keepalive de la R1 către R2 :

PT = Packet Type

| Outer GRE IP | | | | Inner GRE IP | | | |
|--------------|----------|----------|-------|--------------|----------|----------|------|
| IP | IP sursă | IP dest | GRE | IP | IP sursă | IP dest | GRE |
| | 10.0.1.1 | 10.0.2.1 | PT=IP | | 10.0.2.1 | 10.0.1.1 | PT=0 |

• **Observație** : Răspunsul R2 pt. R1 este deja încapsulat în interiorul header-ului IP intern

• **Configurare:**

seconds

retries

```
R1(config)# interface Tunnel0
R1(config-if)# keepalive 5 4
```

GRE Tunnel Identification Key



- Mecanism elementar de autentificare a celor 2 capete ale unui tunel GRE
- Cheie configurată pe cele 2 capete de tunel
 - Trebuie configurată manual pe ambele rutere cu aceeași valoare
 - **Atenție: NU** trebuie folosită ca mecanism de securitate (se trimite pe tunel în clear-text)
 - Utilitate : prevenirea configurărilor incorecte sau a injectării de pachete dintr-o altă sursă
- Configurare

```
R1/R2(config)# interface Tunnel0  
R1/R2(config-if)# tunnel key key-number
```

Sumar

Manipularea
rutelor

Virtual-
Links

Distribute-
list

Redistribuția
protoalelor
de rutare

Policy
Based
Routing

