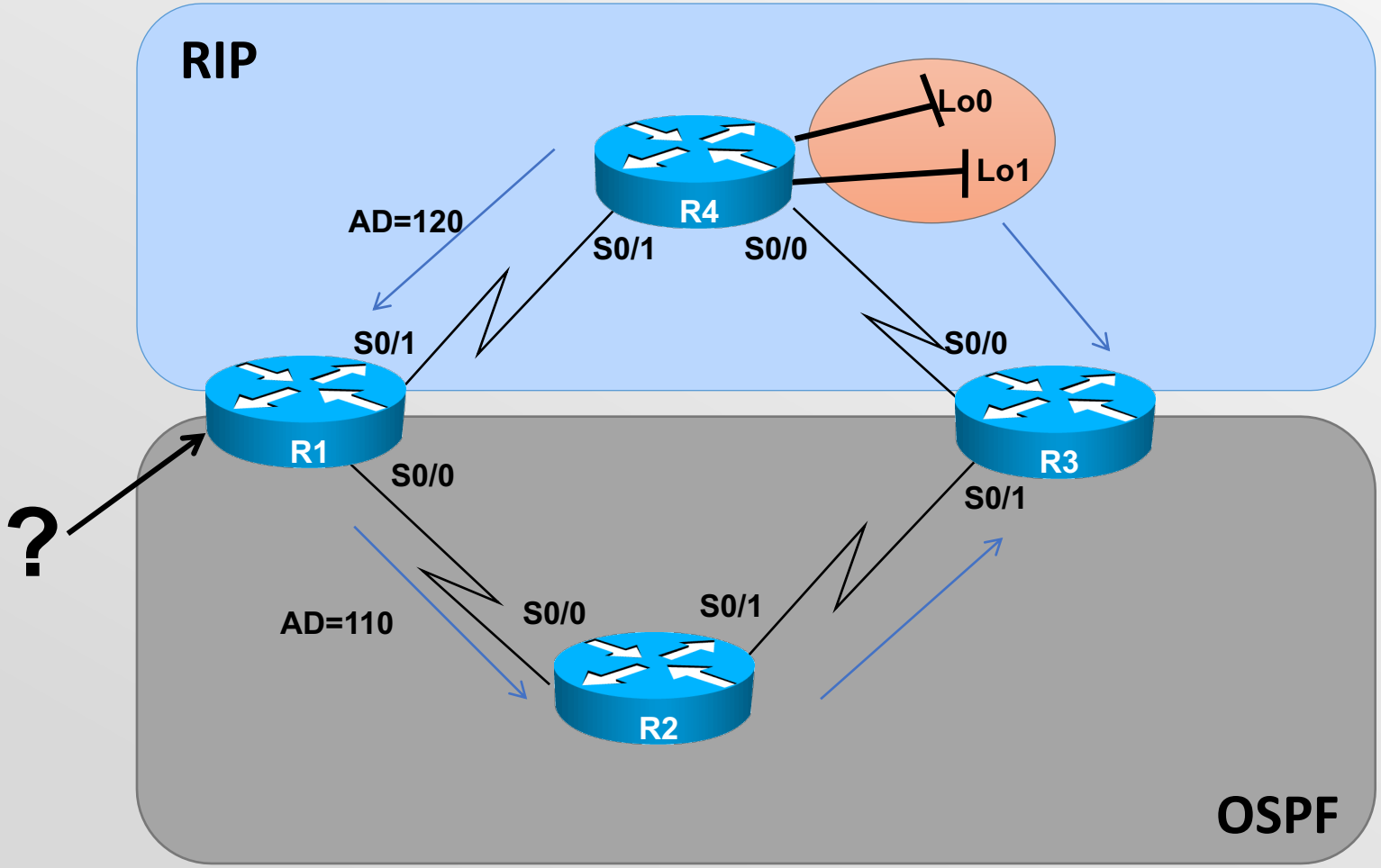
SYSTEMS LABORATORY

Cursul #5

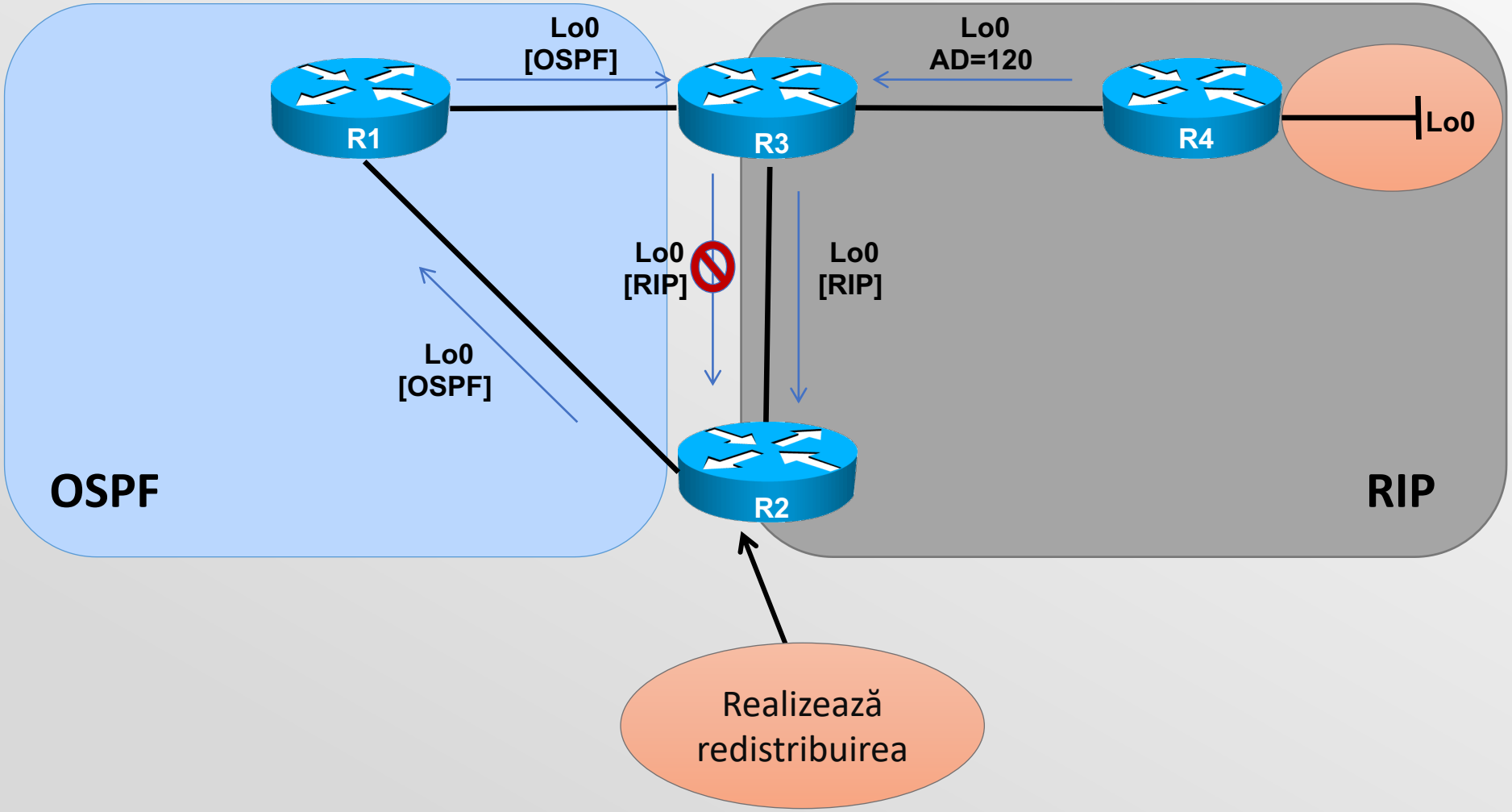
Optimizarea rutării

Controlul distribuției rutelor

Exemplu 1 - PoC



Exemplu 2


Lo0
[OSPF]
Lo0
AD=120
R1
R3
R4
Lo0
Lo0
[RIP]
Lo0
[RIP]
Lo0
[OSPF]
OSPF
R2
RIP
Realizează
redistribuirea

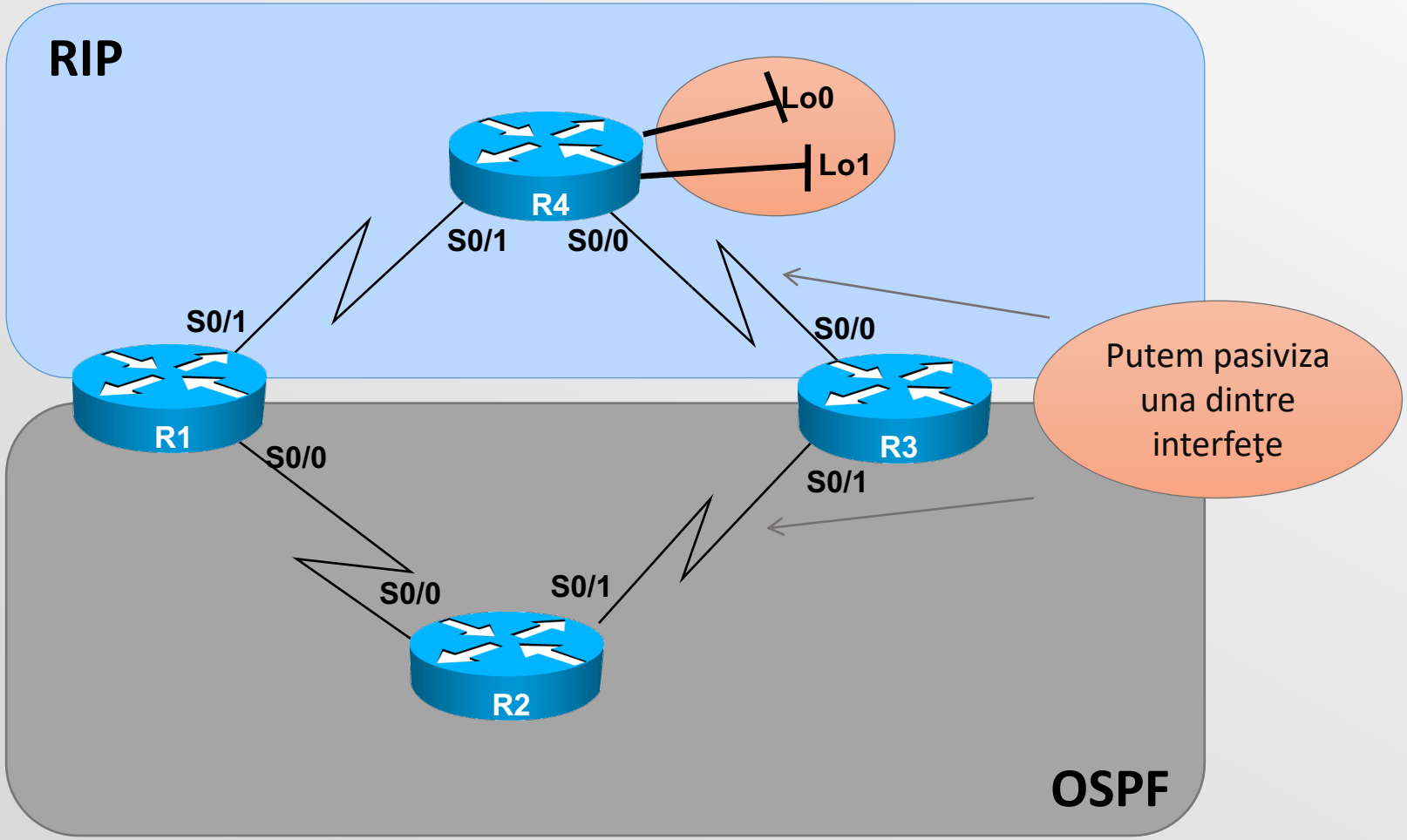
Pasivizarea interfețelor

- 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 reţeaua interfeţei

Exemplu - PoC


RIP
Lo0
Lo1
R4
S0/1
S0/0
S0/1
S0/0
R1
R3
Putem pasiviza una dintre interfeţe
S0/0
S0/1
S0/0
S0/1
R2
OSPF

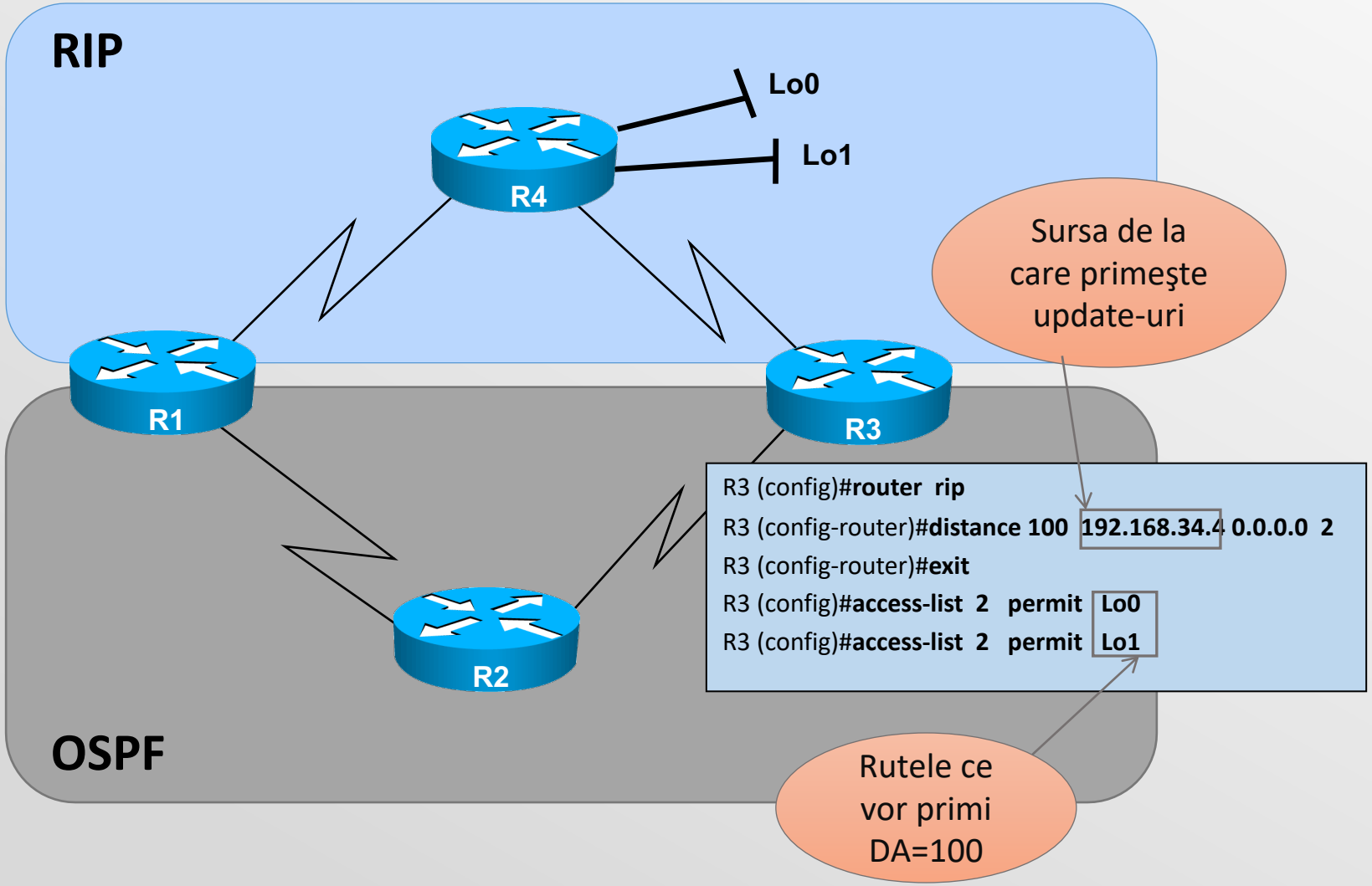
Distanțe administrative

- Din modul de configurare al protocolului de rutare

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

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

Exemplu - PoC


RIP
Lo0
Lo1
R4
R1
R3
R2
OSPF
Sursa de la care primeşte update-uri
R3 (config)#router rip
R3 (config-router)#distance 100 192.168.34.4 0.0.0.0 2
R3 (config-router)#exit
R3 (config)#access-list 2 permit Lo0
R3 (config)#access-list 2 permit Lo1
Rutele ce vor primi DA=100

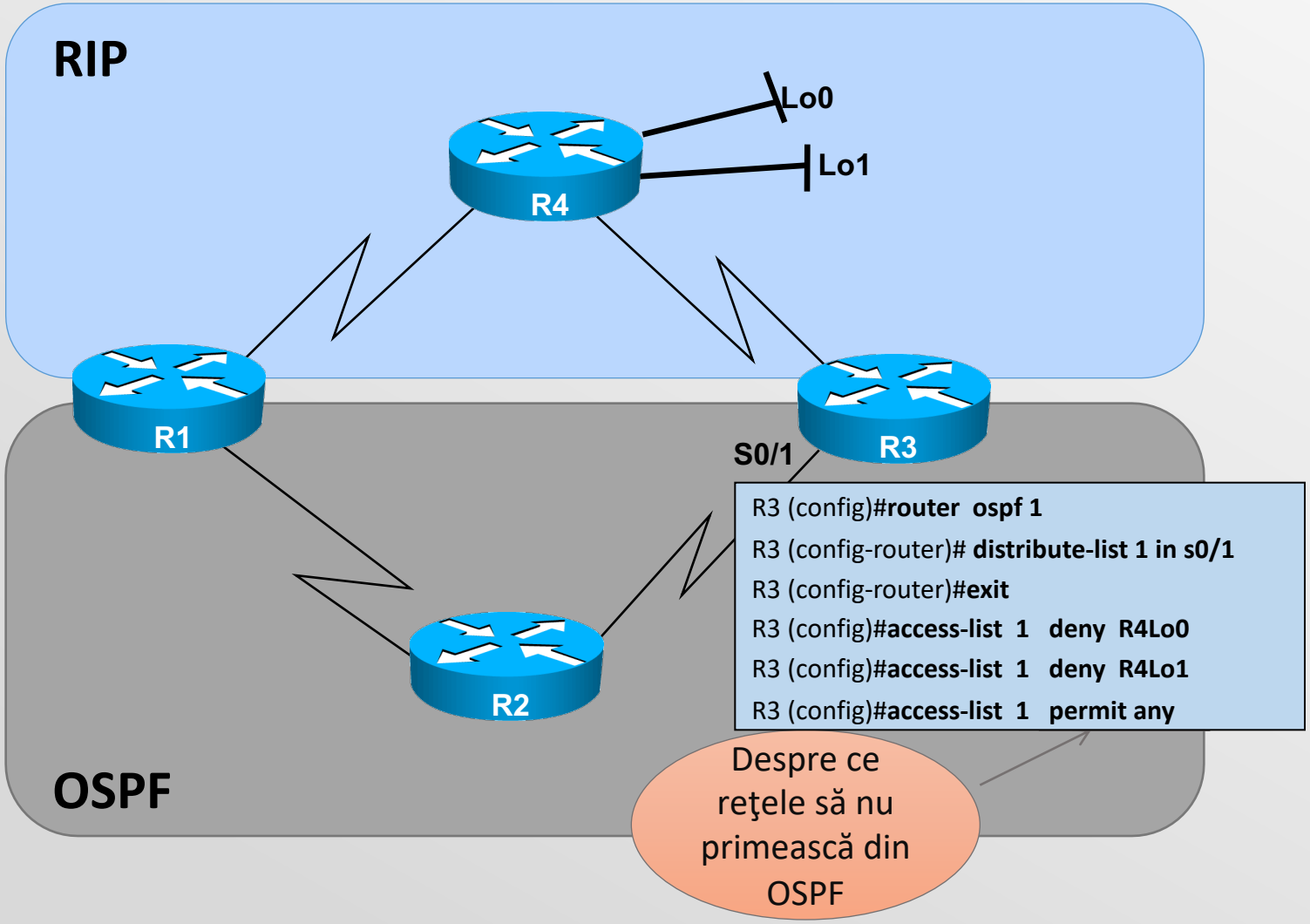
Liste distribuite

- Filtrează numai update-urile de rutare, nu şi 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/primite
EIGRP	Filtrează actualizările trimise/primite din tabela de topologie
OSPF	Filtrează rutele ce vor intra în tabela de rutare
IS-IS	Nu este suportat.

Exemplu - PoC


RIP
Lo0
Lo1
R4
R1
R3
S0/1
R2
OSPF
R3 (config)#router ospf 1
R3 (config-router)# distribute-list 1 in s0/1
R3 (config-router)#exit
R3 (config)#access-list 1 deny R4Lo0
R3 (config)#access-list 1 deny R4Lo1
R3 (config)#access-list 1 permit any
Despre ce reţele să nu primească din OSPF

Dezavantaje „distribute-list”

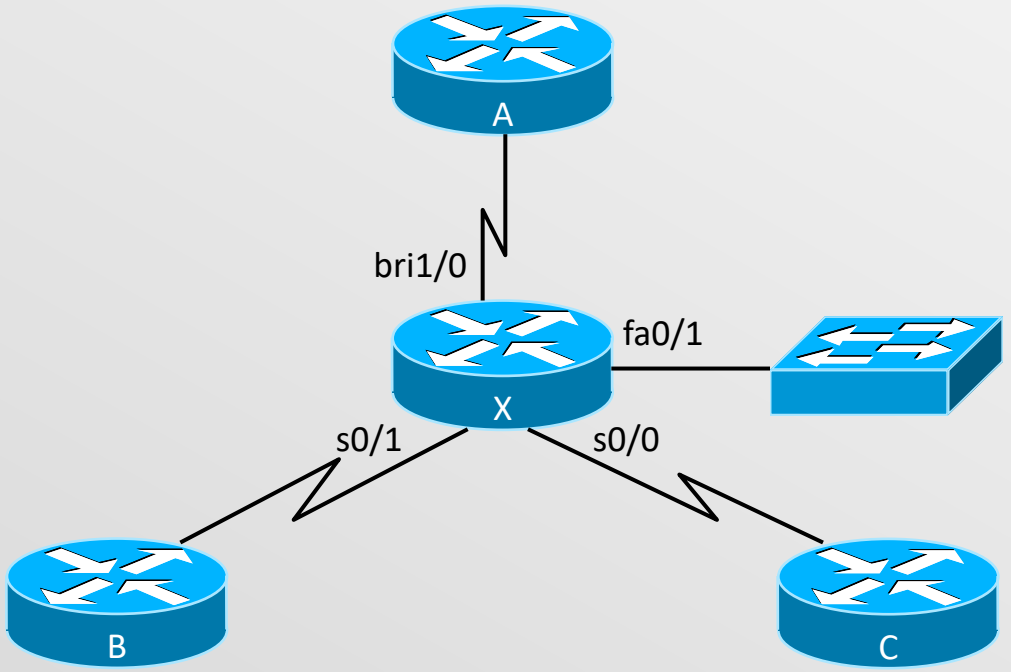
- În primul rând...
  - ... filtrarea de rute se poate aplica în orice situație; nu doar în problema rutării suboptimale
- Dezavantaje “aparente”?
  - Nescalabilă: depinde de ACL-uri
- Optimizarea distribute-list
  - Folosind tehnici de **route tagging** (nu se mai folosesc ACL-uri decât la identificarea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

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 şi 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/primite
EIGRP	Filtrează actualizările trimise/primite 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)

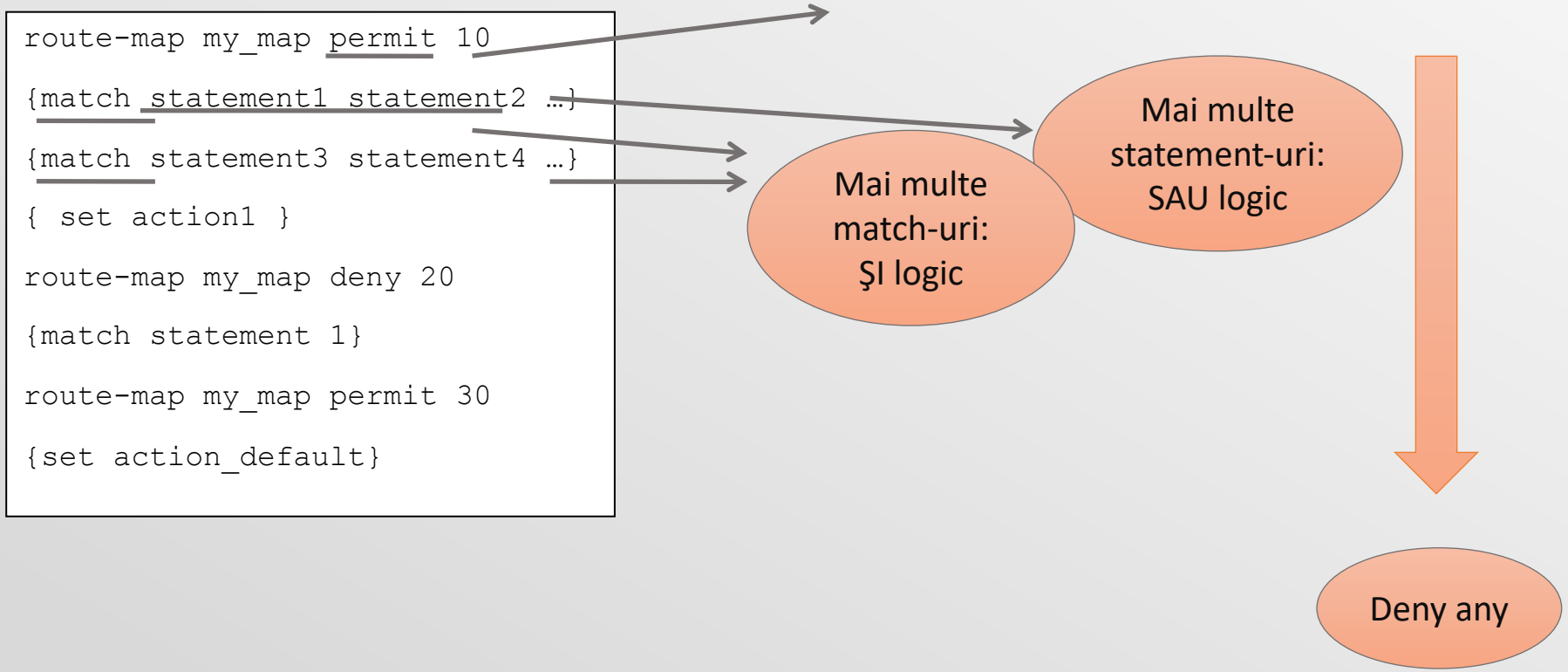
Route-maps

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

```
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}
```

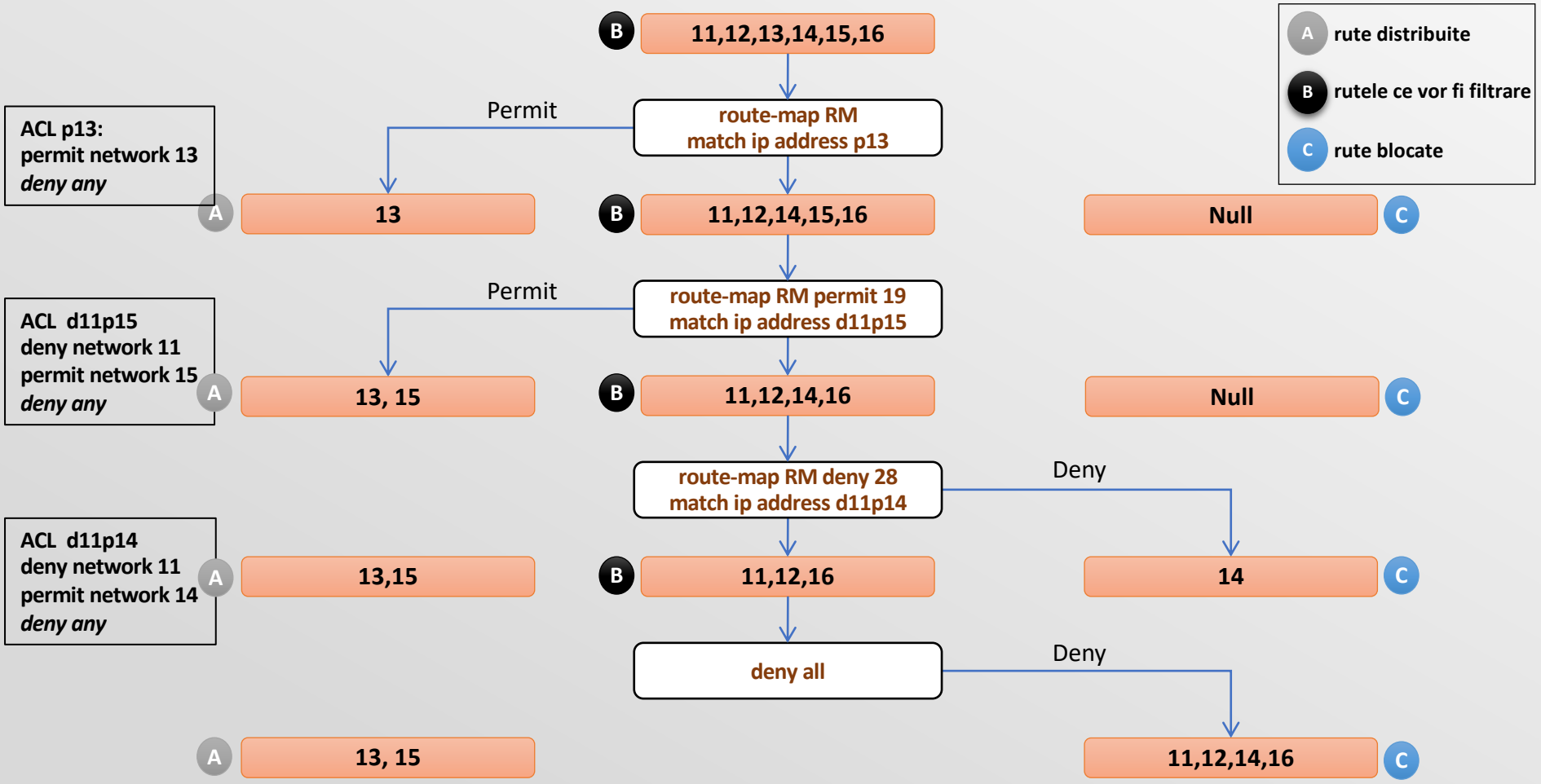


- Lipsa unei clauze match == match any

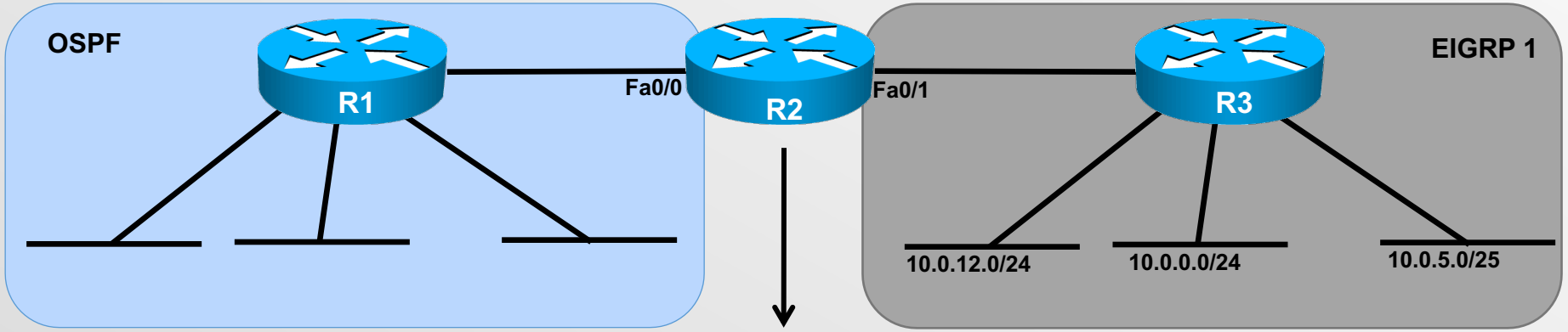
Comanda match

Comanda	Descriere
`match interface`	Interfaţa de ieşire a rutelor
`match ip address`	Folosește ACL şi prefix-list
`match ip next-hop`	Adresa IP a următorului hop
`match ip route-source`	ACL pentru sursa ruterului care a trimis actualizarea
`match metric`	Metrica rutei
`match route-type`	Tipul rutei
`match tag`	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-l 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

PR network crunch

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

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

- Stabileşte **interfaţa de ieşire** pe care să fie trimis pachetul:

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

- Stabileşte **următorul hop**, în cazul în care nu există o rută explicită către destinaţie:

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

- Stabileşte **interfaţa de ieşire**, în cazul în care nu există o rută explicită către destinaţie:

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

Policy-based routing

Policy-based routing

PR network crunch

- 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:

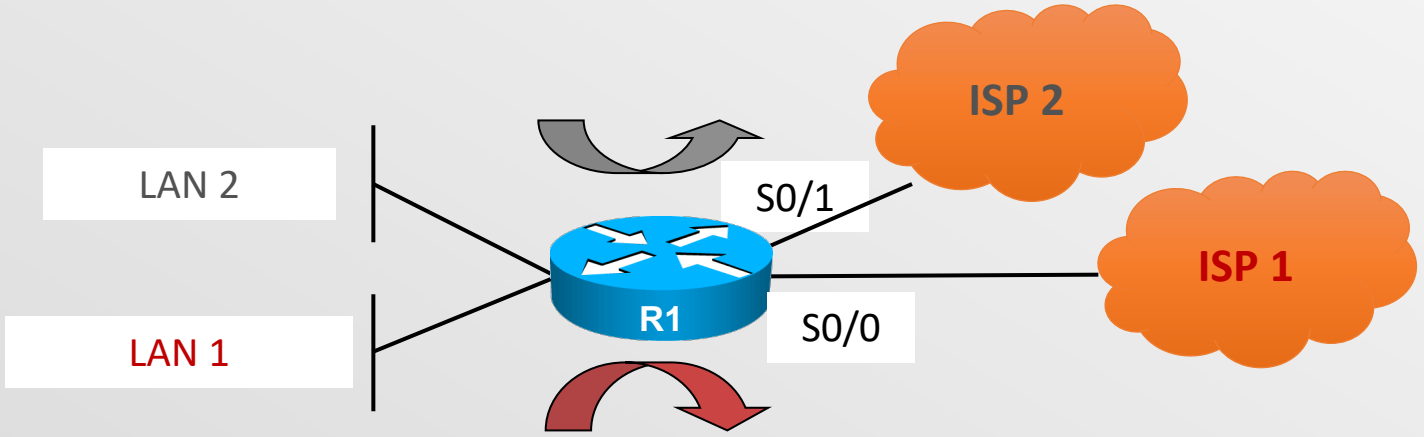
```
(config-if)# ip policy route-map <name>
```

- Pentru a afecta traficul generat de ruter:

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

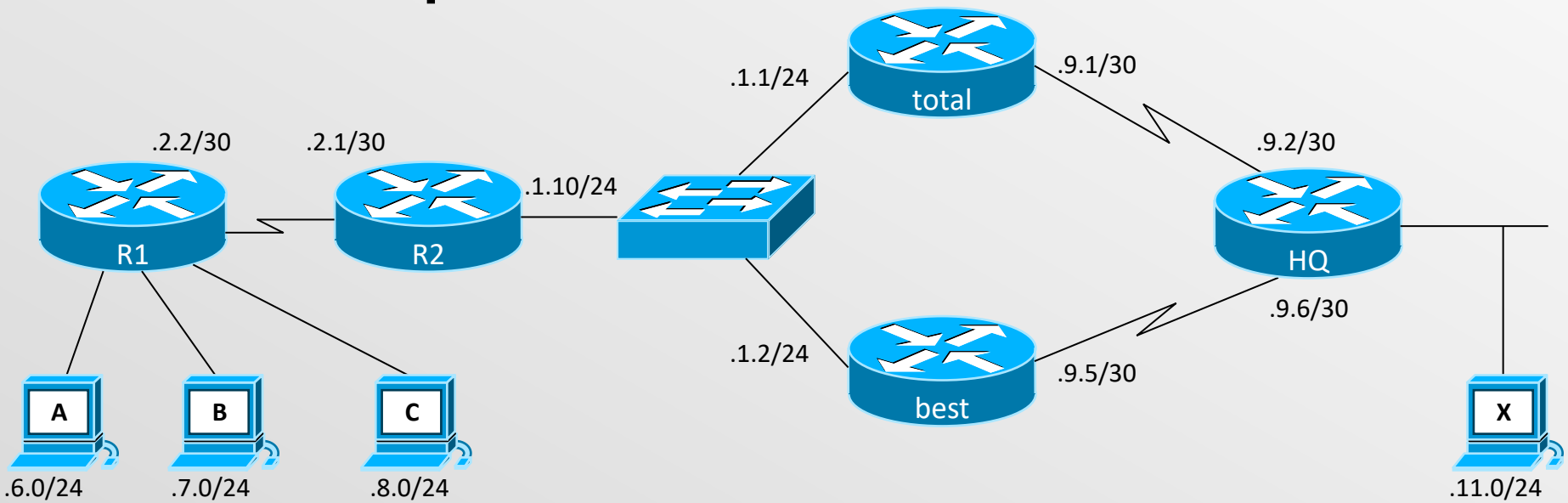
- Dacă un pachet nu face match pe nici o regulă de route-map, acesta este trimis în procesul de rutare normal

PBR – Exemplu 1



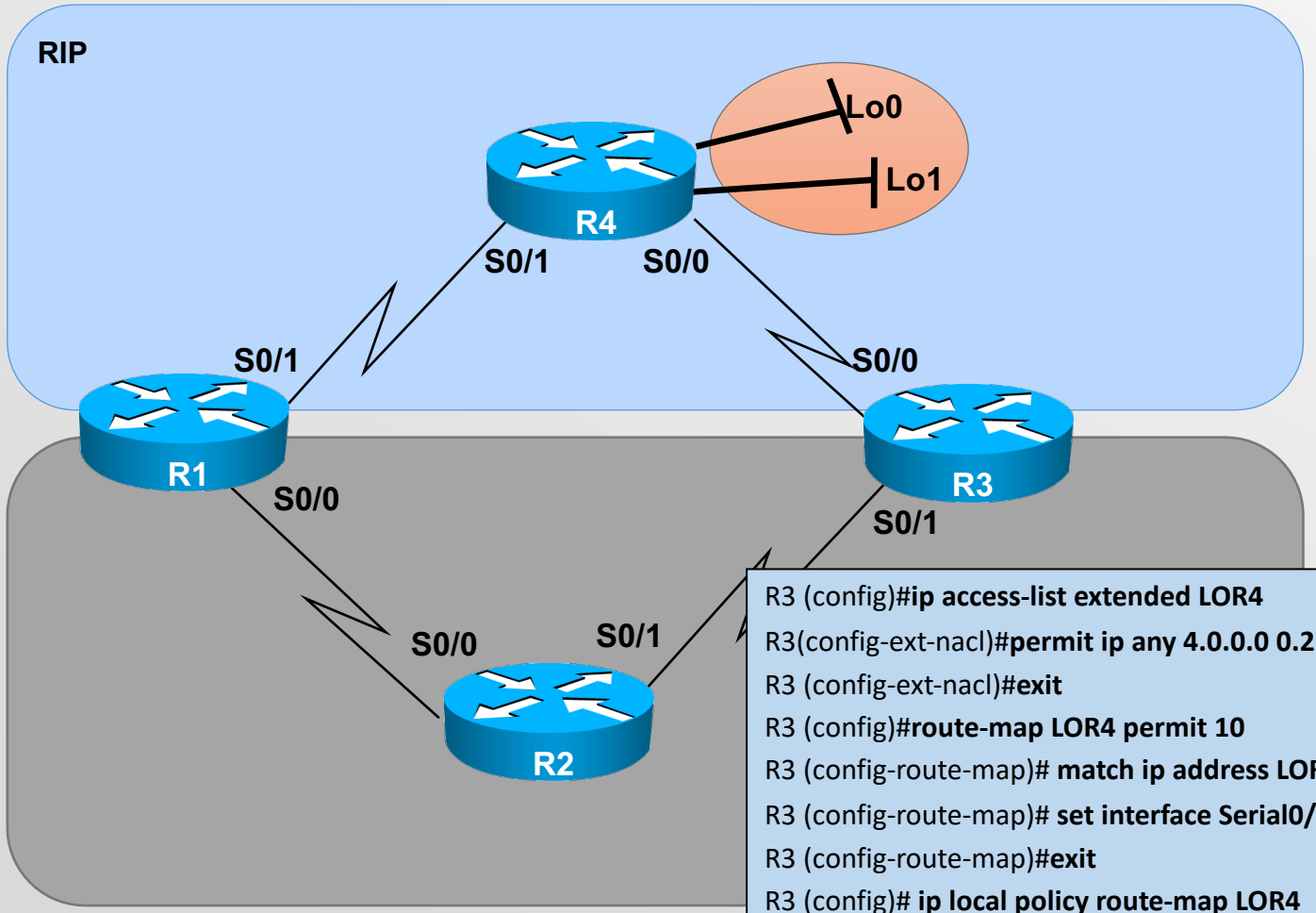

```
R1(config)#interface e0/0
R1(config-if)#ipv6 policy route-map ISP1
R1(config)#interface e0/1
R1(config-if)#ipv6 policy route-map ISP2
R1(config)#route-map ISP1 permit 10
R1(config-route-map)#match ipv6 address 1
R1(config-route-map)#set interface s0/0
R1(config)#route-map ISP2 permit 10
R1(config-route-map)#match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



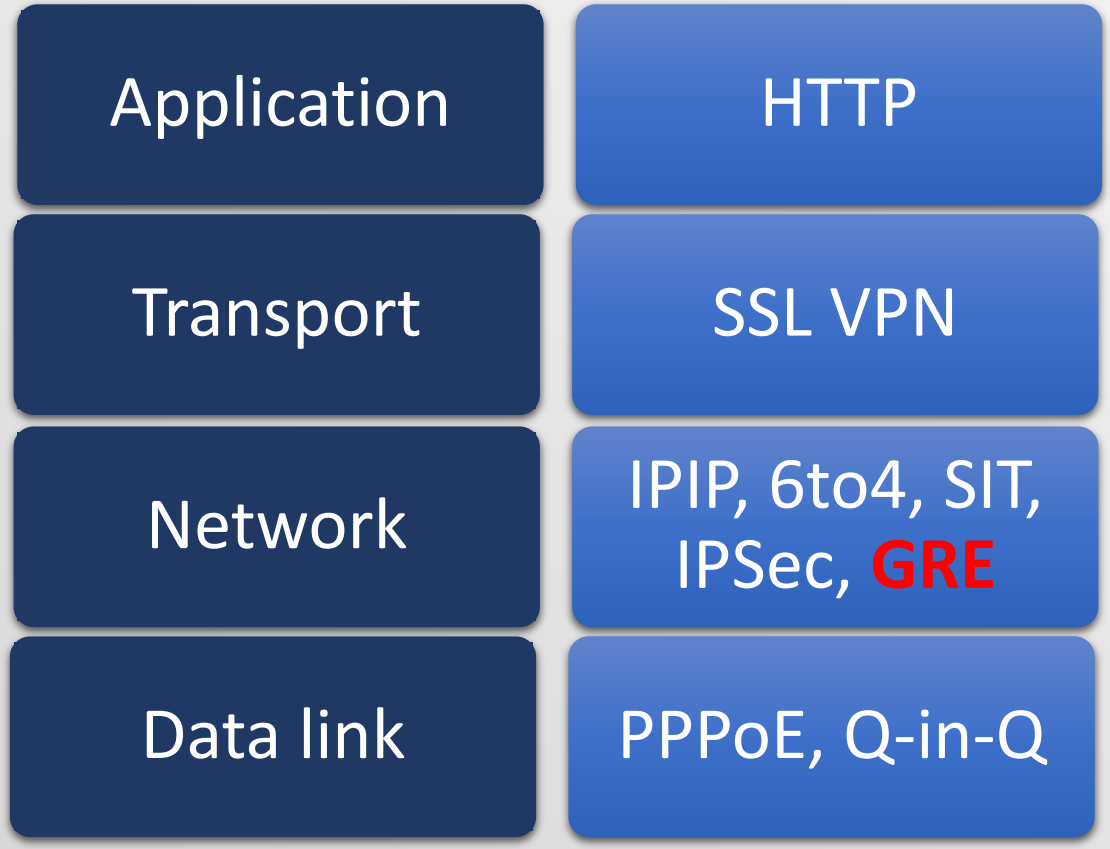

```
R3 (config)#ip access-list extended LOR4
R3(config-ext-nacl)#permit ip any 4.0.0.0 0.255.255.255
R3 (config-ext-nacl)#exit
R3 (config)#route-map LOR4 permit 10
R3 (config-route-map)# match ip address LOR4
R3 (config-route-map)# set interface Serial0/0
R3 (config-route-map)#exit
R3 (config)# ip local policy route-map LOR4
```

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



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

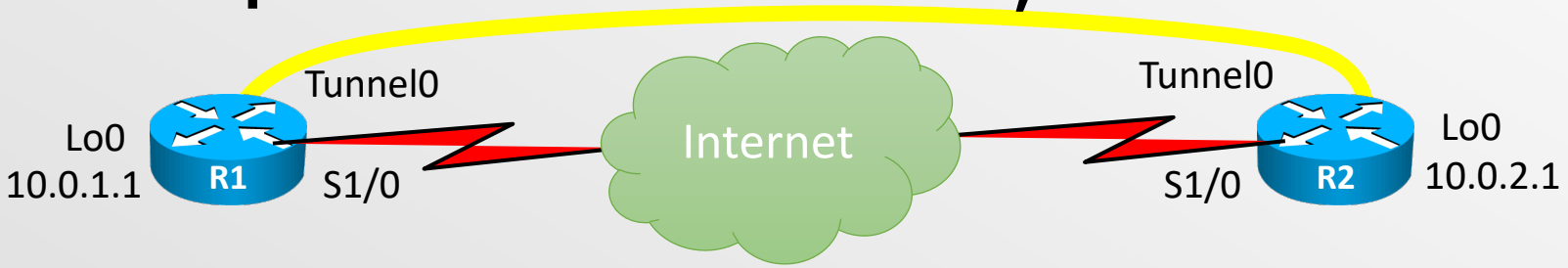
```
R2(config)# ip route 11.0.0.0 255.255.255.0 Fa0/0
R2(config)#interface Tunnel0
R2(config-if)# ip address 12.0.0.2 255.255.255.0
R2(config-if)# tunnel source FastEthernet0/0
R2(config-if)# tunnel destination 11.0.0.1
```


```
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 dispare - 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
- Redstribuția protocoalelor de rutare
- Policy Based Routing