



Optimizarea rutării

Proiectarea Rețelelor

Cuprins

- ▶ Redistribuția între protocoale de rutare
- ▶ Manipularea rutelor
 - ▶ Passive-interface
 - ▶ Distanțe administrative
 - ▶ Distribute-list
- ▶ Route-maps
 - ▶ Redistribuție
 - ▶ Policy Based Routing
 - ▶ Atribute BGP



*Redistribuția între protocoale
de rutare*

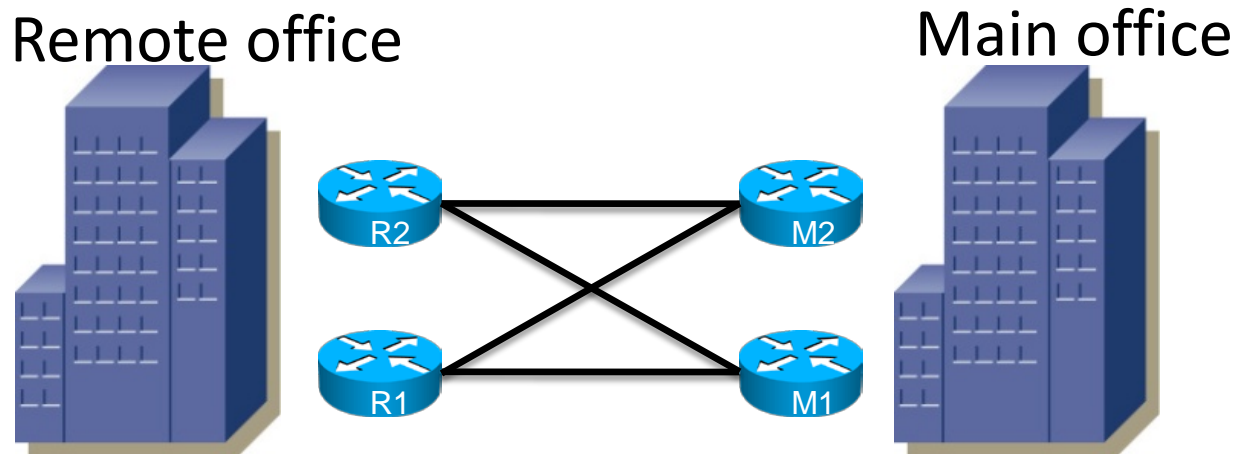


Rute statice vs Protocoale de rutare

- ▶ Protocoalele de rutare oferă:
 - ▶ Flexibilitate
 - ▶ Redundanță la nivel de rute
 - ▶ Scalabilitate
 - ▶ Automatizare
- ▶ Deci... de ce mai folosim rute statice?
 - ▶ Securitate
 - ▶ Simplitate
 - ▶ Siguranță

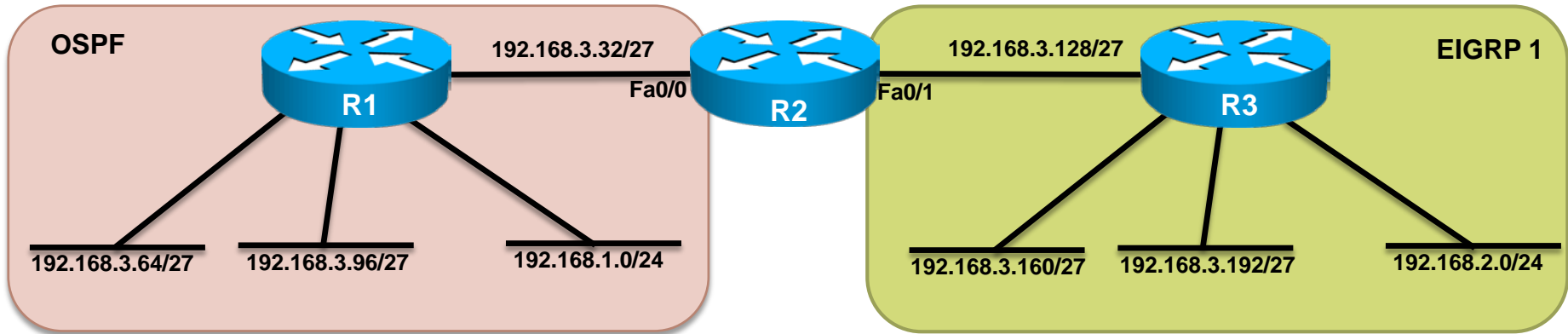
Mai multe protocoale de rutare

- ▶ Conectivitate la fuzionarea companiilor
- ▶ Oferirea de servicii integrate de către ISP către clienți
- ▶ Obiective:
 - ▶ Folosire protocoale de rutare
 - ▶ Conexiune redundantă (la nivel de rute) între cele 2 clădiri



Redistribuție

- ▶ În cadrul unui ruter pot rula mai multe protocoale de rutare
- ▶ Redistribuirea rutelor
 - ▶ Informațiile învățate printr-un protocol trebuie transmise către alte protocoale
- ▶ Se poate realiza
 - ▶ Unde
 - ▶ One point
 - ▶ Multiple points
 - ▶ Cum
 - ▶ One-way: un protocol învață informațiile unui alt protocol
 - ▶ Two-way: ambele protocoale fac schimb de informații



► Comanda “redistribute”

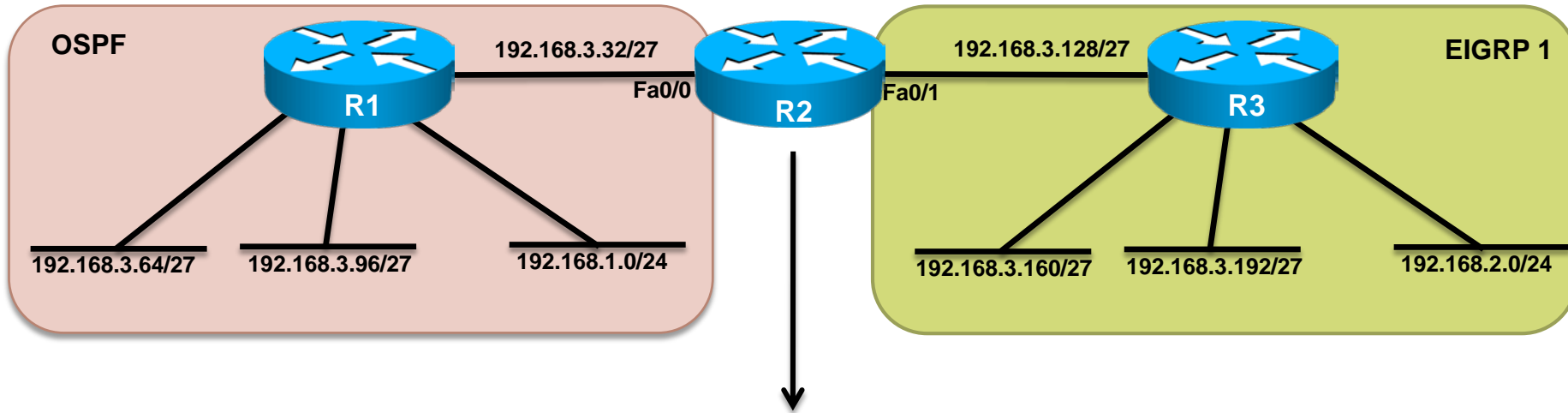
► Unde?

- În modul de configurare al protocolului de rutare în care se vor adăuga rețelele

► Ce?

- Primește ca parametru tipul de rețele care vor fi adăugate

Exemplu : OSPF – EIGRP



One-way:

```
router eigrp 1
  network 192.168.3.0
  redistribute ospf 1 metric 1000 100 1 255 1500
```

Two-way:

```
router ospf 1
  redistribute eigrp 1 metric 100 metric-type 1 subnets
  default-metric 50
  network 192.168.3.33 0.0.0.0 area 0
```

Metrică
injectată

Metrica de intrare (seed metric)



- ▶ Setarea metricii implicite
 - ▶ Pentru toate rutele introduse
 - ▶ comanda `(config-router)# default-metric`
 - ▶ Pentru rute specifice la redistribuția acestora
 - ▶ parametrul `metric` al comenzii `"redistribute"`

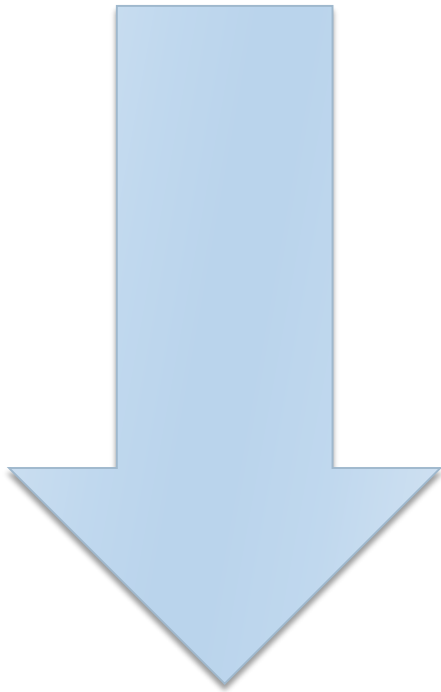
- ▶ Se recomandă modificarea metricii implicite pentru a satisface cerințele rețelei

- ▶ A nu se confunda cu parametrul `"metric-type"` al OSPF.

Metrici de intrare

Protocol sursă	Protocol destinație				
/	RIP	EIGRP	OSPF	ISIS	BGP (MED)
Connected	1	Metrica int	20 (E2)	0	0
Static	1	Metrica int	20 (E2)	0	0
RIP	-	infinite	20 (E2)	0	Metrică IGP
EIGRP	infinite	-	20 (E2)	0	Metrică IGP
OSPF	Infinite	Infinite	20 (E2)	0	Metrică IGP
ISIS	Infinite	Infinite	20 (E2)	0	Metrică IGP
BGP	infinite	infinite	1 (E2)	0	Metrică IGP

Probleme potențiale



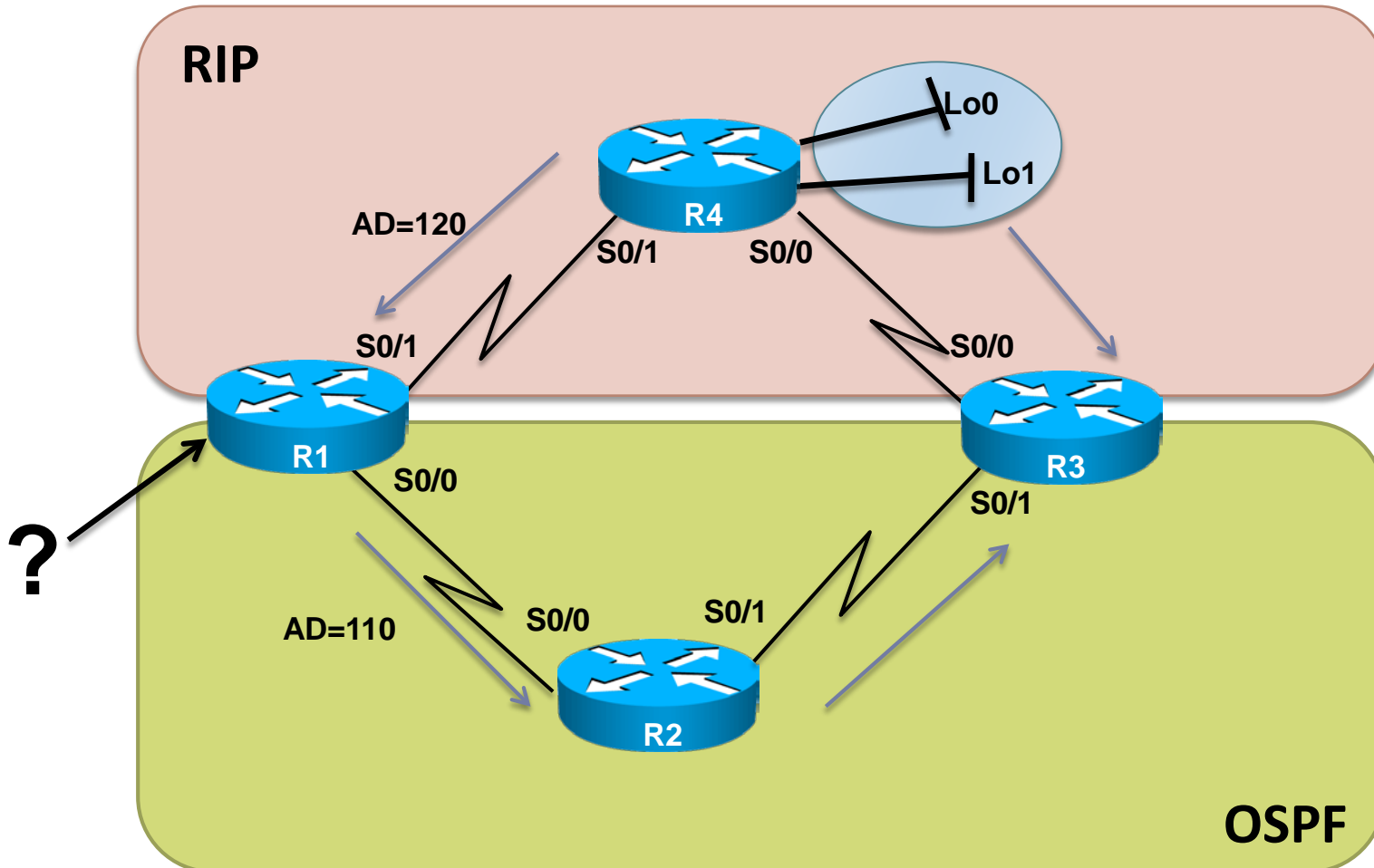
- ▶ Bucle de rutare
 - ▶ Se pot transmite informații de rutare despre un AS înapoi în acel AS
- ▶ Informații incompatibile
 - ▶ Nemodificarea metricilor
- ▶ Inconsistența timpilor de convergență
 - ▶ Timpuri diferiți de convergență pentru protocoalele între care se realizează redistribuirea



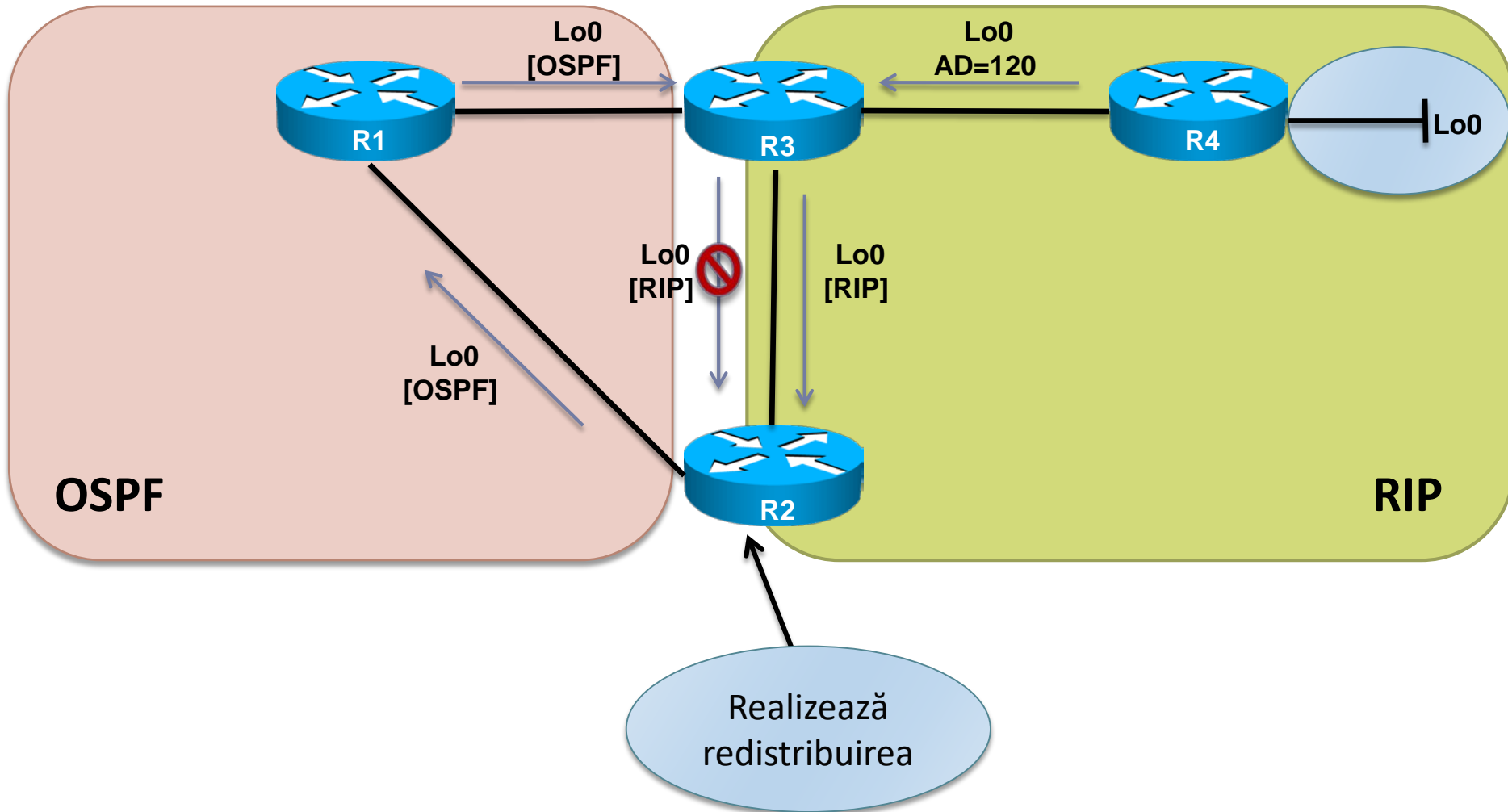
Manipularea rutelor



Exemplu 1 – PoC



Exemplu 2



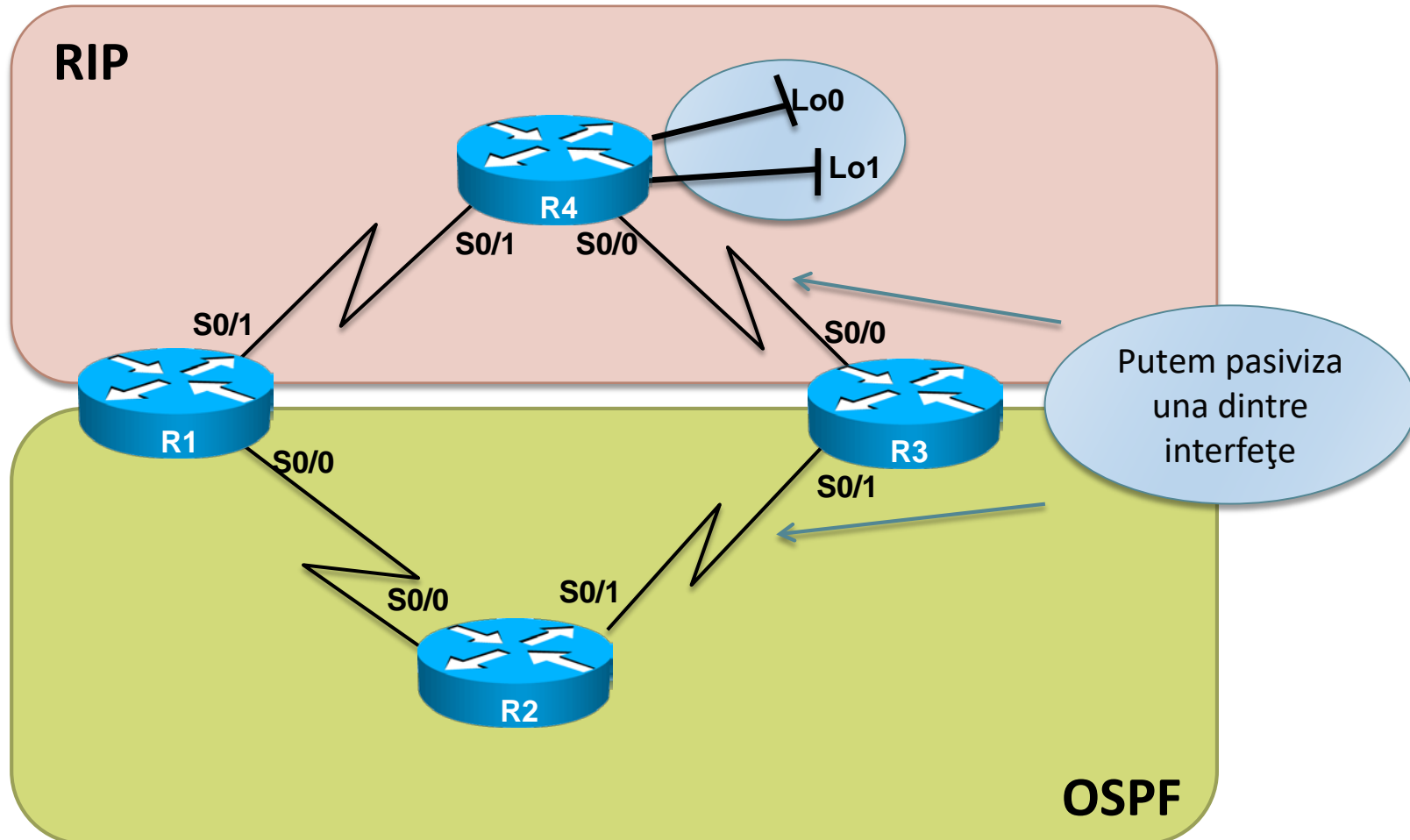
Pasivizarea interfețelor

- ▶ 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



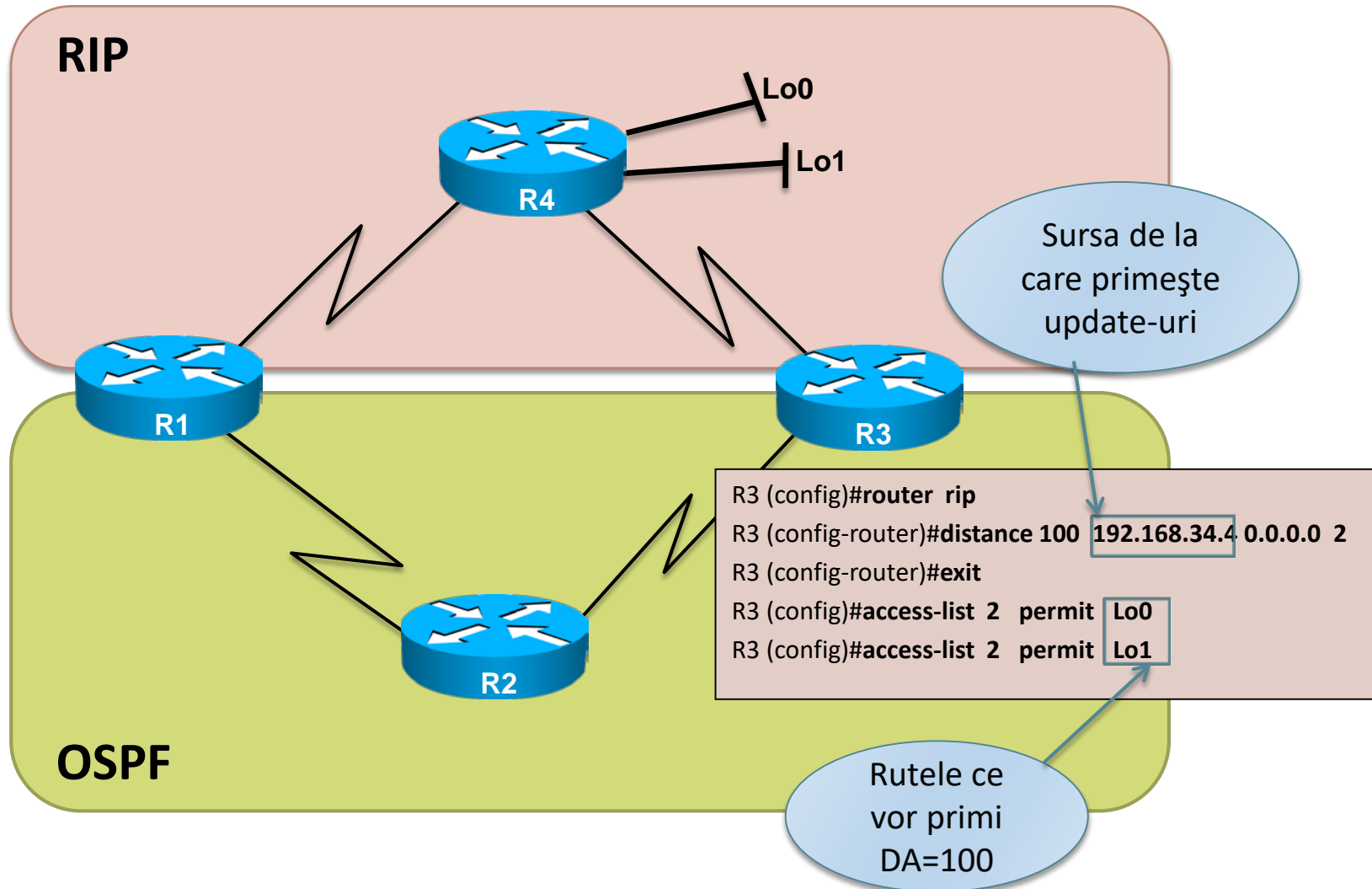
Distanțe administrative

- ▶ 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



Dezavantaje “distance”

- ▶ Greu de urmărit în configurații complexe
- ▶ Nu e o soluție scalabilă: se bazează pe intrări în ACL-uri
- ▶ IP-ul dat ca parametru în comandă reprezintă pentru OSPF/IS-IS RID-ul, iar pentru celelalte protocoale de rutare, IP-ul vecinului de la care s-a primit update-ul
- ▶ Modificarea este locală:
 - ▶ Distanța administrativă nouă nu este comunicată altor rutere

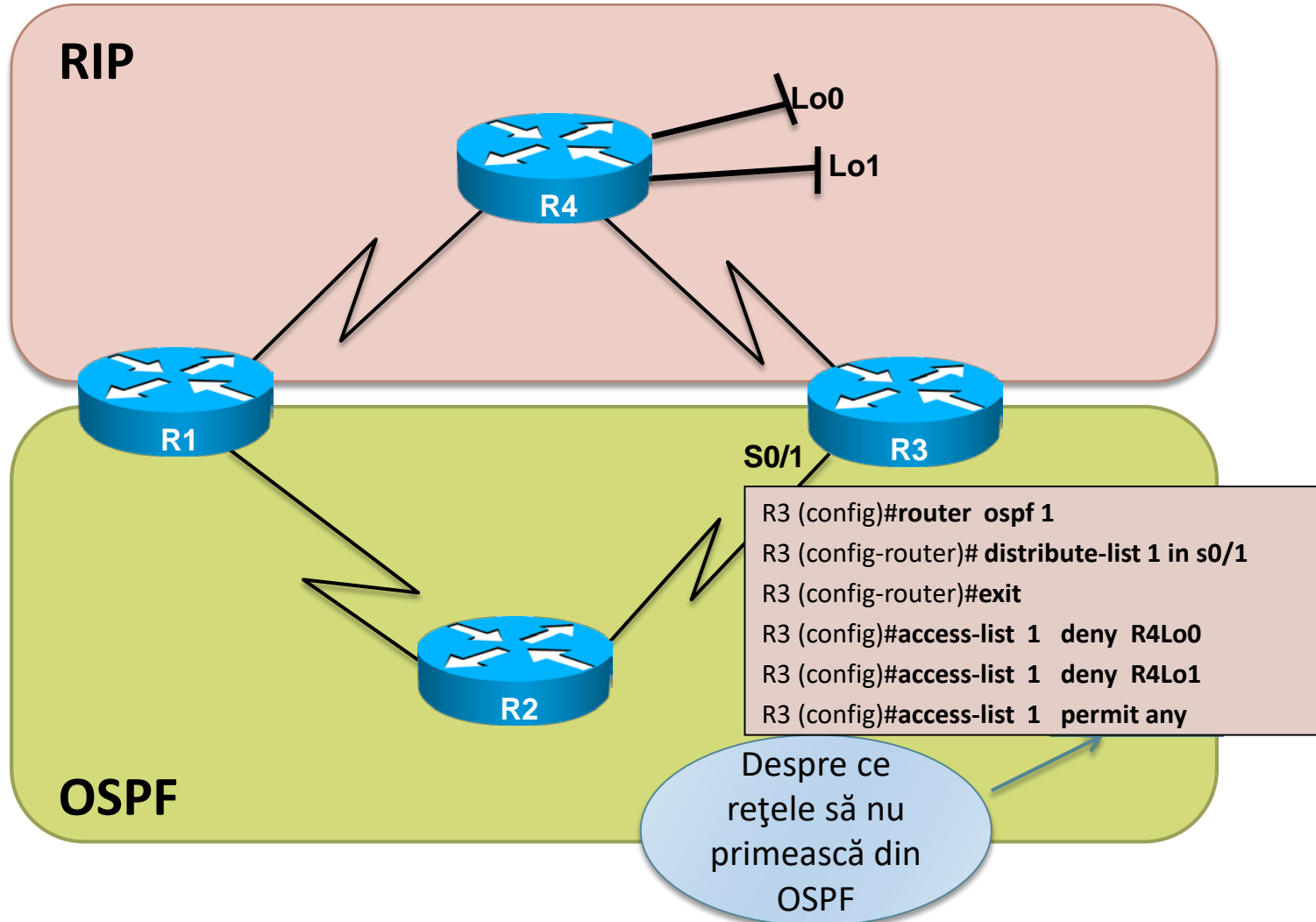
Liste distribuite

- ▶ 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)

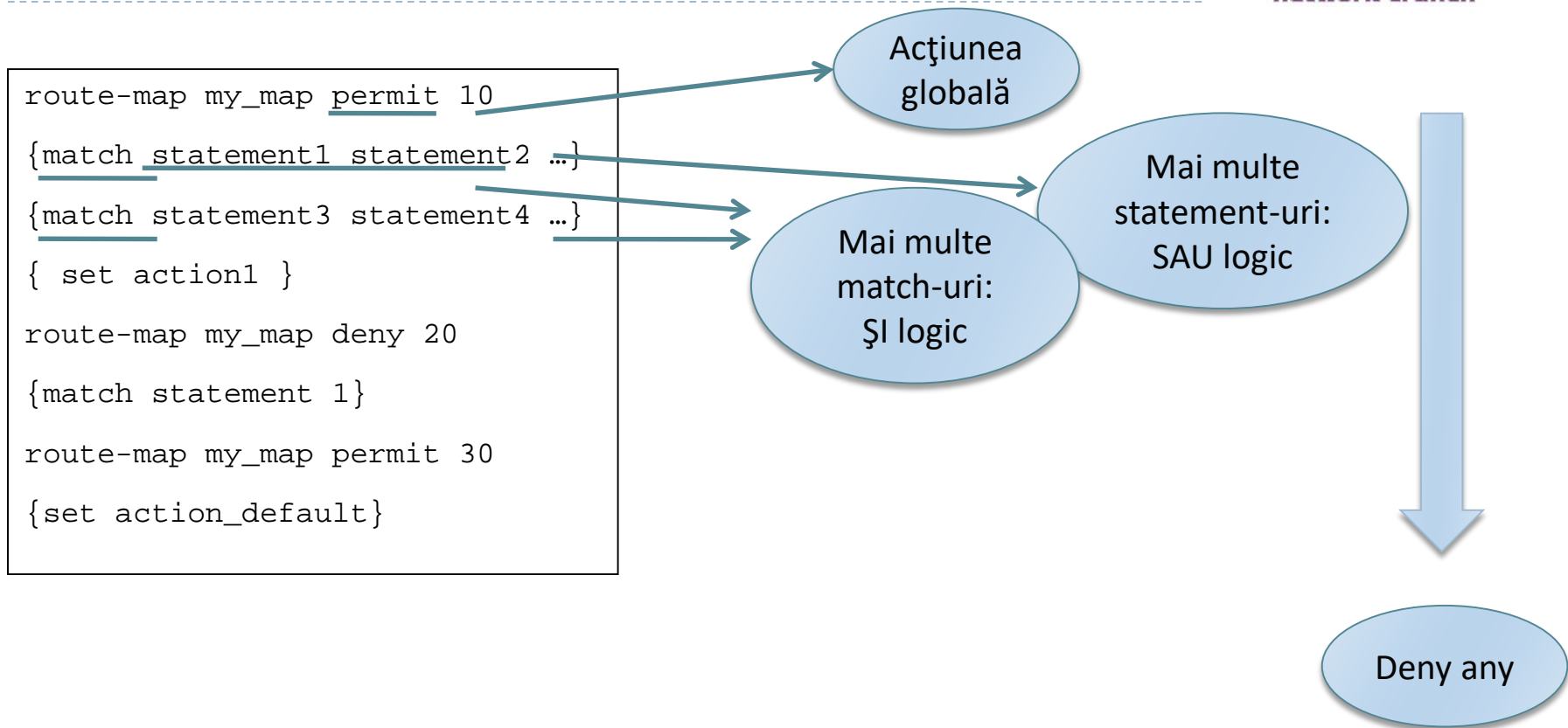
Route-maps



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

Pacurgerea unui route-map



► Lipsa unei clauze match == match any

Clauza 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

Clauza match – exemplu

- ▶ Dacă nu este specificată comanda match, se consideră valide toate pachetele
- ▶ O declarație match cu mai multe condiții – ACL1 sau ACL2

```
Router(config-route-map)# match ip 1 2
```

- ▶ Mai multe declarații match – ACL1 și ACL2

```
Router(config-route-map)# match ip 1  
Router(config-route-map)# match ip 2  
  
Router(config)# access-list 1 permit 10.0.1.0 0.0.0.255  
Router(config)# access-list 2 permit 10.0.2.0 0.0.0.255
```

Clauza 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:

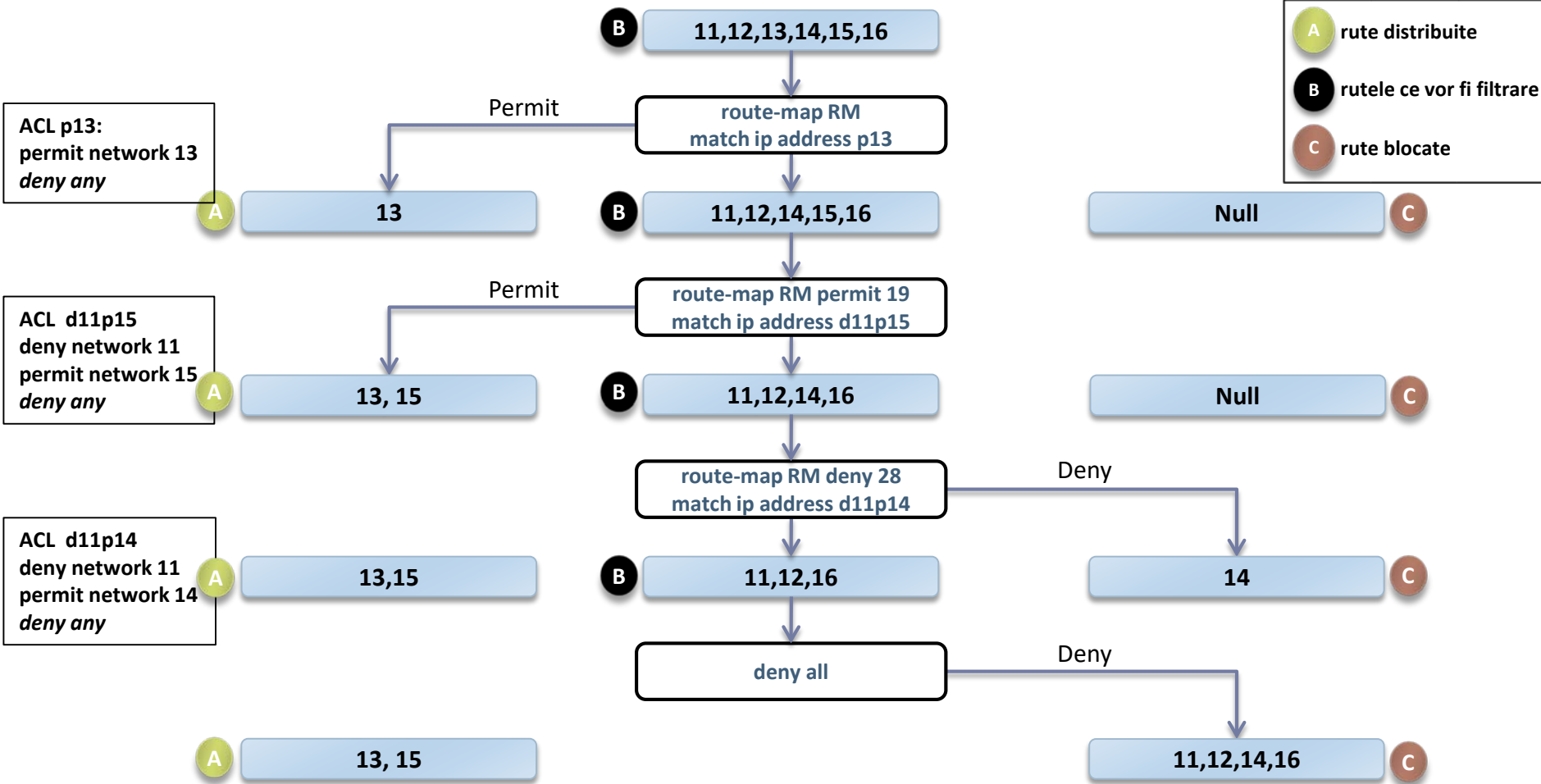
```
Router(config-route-map)#set ip default next-hop ip-address [...ip-address]
```

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

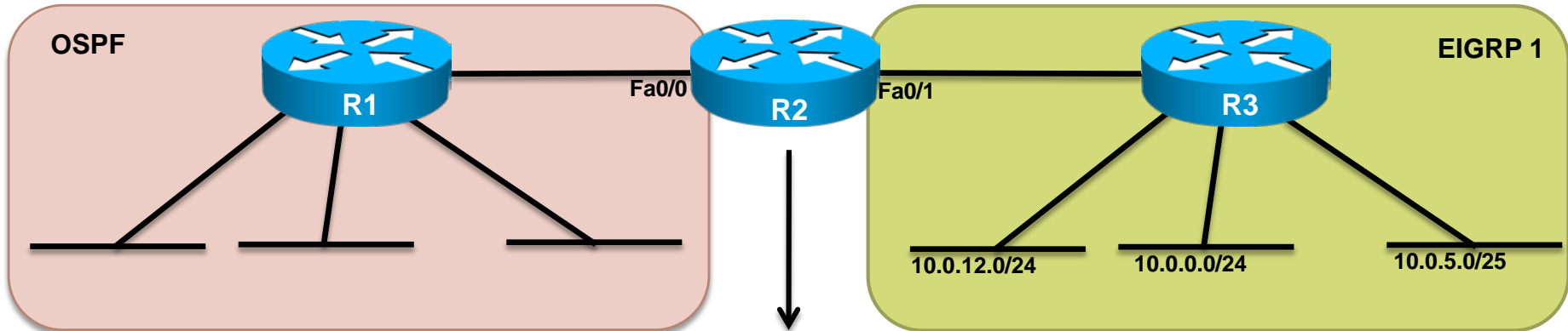
```
Router(config-route-map)#set default interface interface-type interface-number [... type ...number]
```

Logica unui route-map

● A rute distribuite
● B rutele ce vor fi filtrare
● C rute blocate



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).

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:

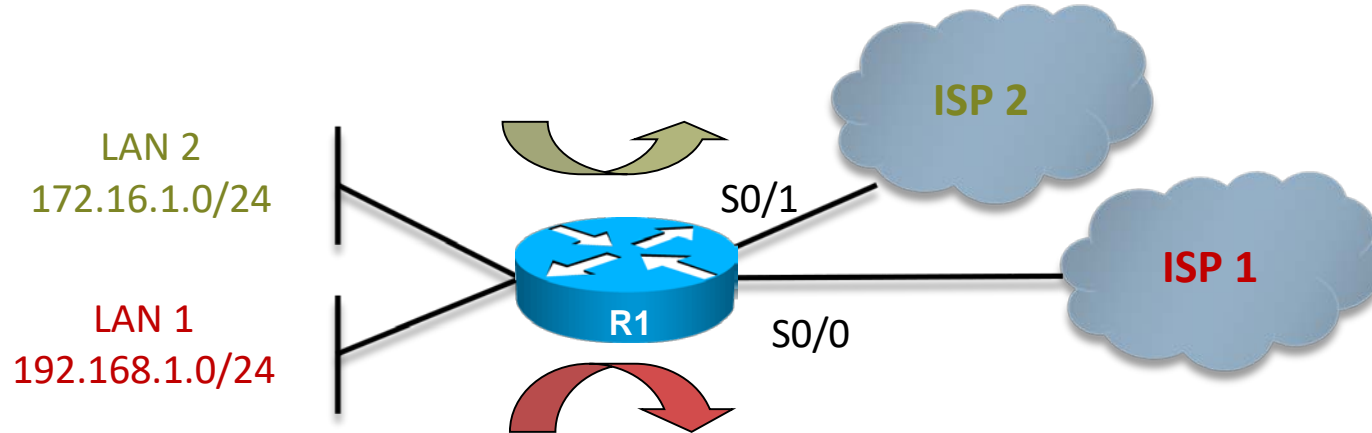
```
(config-if)# ip policy route-map <name>
```

- ▶ Pentru a afecta traficul generat de ruter:

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

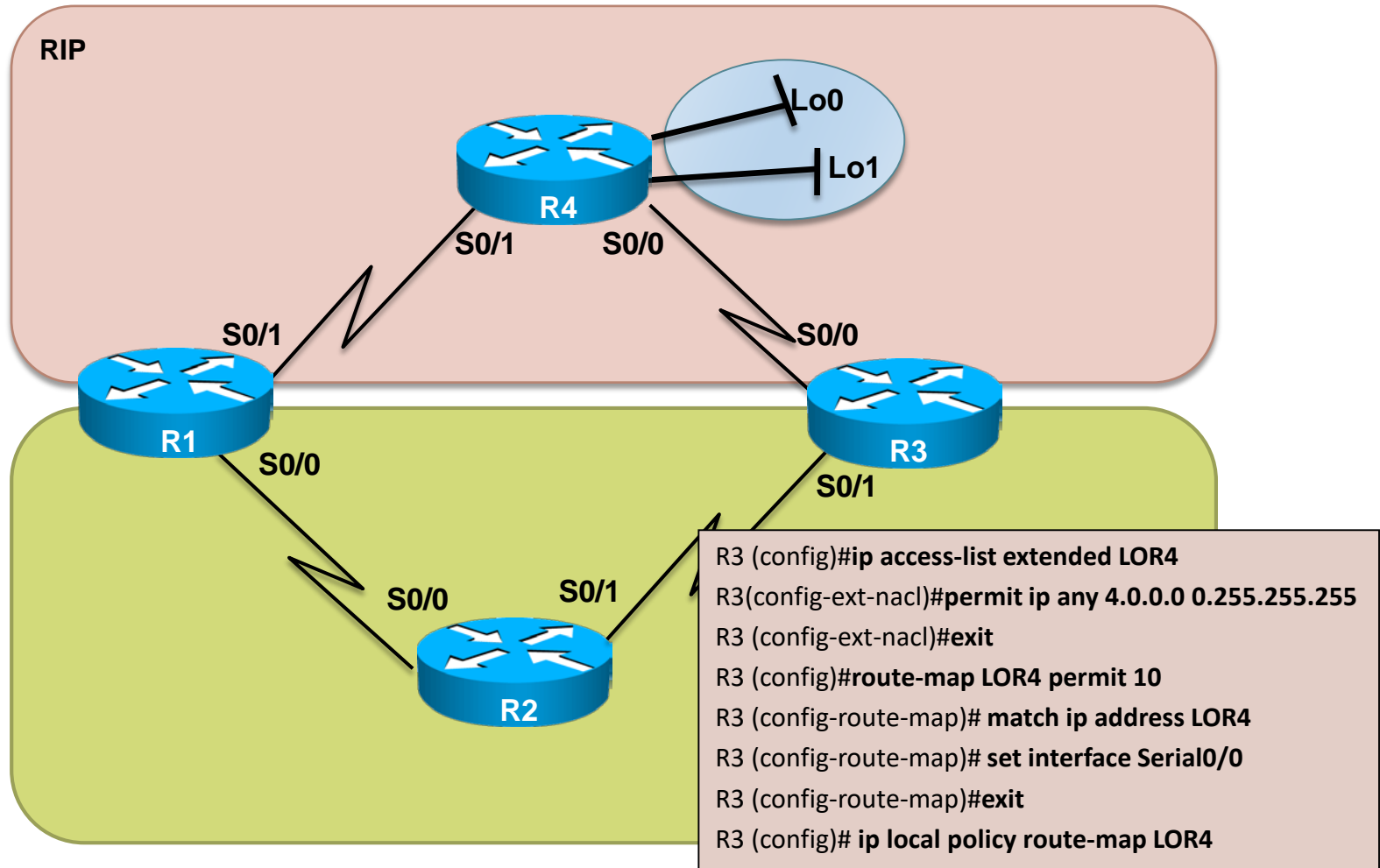
- ▶ Dacă un pachet nu face match pe nici o regulă de route-map, acesta este trimis în procesul de rutare normal

PBR – Exemplu



```
R1(config)#interface e0/0
R1(config-if)#ip policy route-map ISP1
R1(config)#interface e0/1
R1(config-if)#ip policy route-map ISP2
R1(config)#route-map ISP1 permit 10
R1(config-route-map)#match ip address 1
R1(config-route-map)#set interface s0/0
R1(config)#route-map ISP2 permit 10
R1(config-route-map)#match ip address 2
R1(config-route-map)#set interface s0/1
R1(config)#access-list 1 permit 192.168.1.0 0.0.0.255
R1(config)#access-list 2 permit 172.16.1.0 0.0.0.255
```

PBR – PoC



Sumar

- ▶ Redistribuția protocoalelor de rutare
- ▶ Manipularea rutelor
 - ▶ Passive-interface
 - ▶ Distanțe administrative
 - ▶ Distribute-list
- ▶ Policy Based Routing

