



# ACL și NAT

Proiectarea Rețelelor

# Cuprins

---

- ▶ Access Lists
  - ▶ Ce este un ACL
  - ▶ Funcționarea ACL-urilor
  - ▶ Tipuri de liste de acces
  - ▶ Exemple de configurare
- ▶ Network Address Translation
  - ▶ De ce este nevoie?
  - ▶ Concepte NAT
  - ▶ Configurare NAT
- ▶ Tunelare
  - ▶ Tunelare GRE



## *Access Lists*



# Ce este un Access List?

---

- ▶ Un **set de condiții** specificate de către administrator pentru **identificarea** unor anumite tipuri de trafic
- ▶ Traficul identificat poate fi
  - ▶ Filtrat
  - ▶ Alterat
  - ▶ Controlat
  - ▶ Asociat cu alte acțiuni
- ▶ În funcție de acțiunea dorită, traficul trebuie identificat după anumite criterii

# Utilități ale ACL-urilor

---

- ▶ Filtrarea și monitorizarea traficului
  - ▶ Cea mai des folosită aplicație a ACL-urilor
  - ▶ Remember **iptables –t filter**
  - ▶ Permitearea sau respingerea traficului
  - ▶ Inspecția mai avansată a traficului identificat

# Utilități ale ACL-urilor

---

- ▶ Marcarea și alterarea traficului
  - ▶ Remember **iptables -t mangle** and **-t nat**
  - ▶ QoS
    - ▶ Pasul 1: traffic tagging
    - ▶ Pasul 2: Traffic policing și traffic shaping
  - ▶ NAT
  - ▶ Criptare

# Utilități ale ACL-urilor

---

- ▶ Asocierea cu accesul la alte servicii
  - ▶ Accesul la terminale virtuale (ssh/telnet/http)
  - ▶ Controlul actualizărilor protoalelor de rutare
  - ▶ Policy baseded routing (vom vedea în curs 10)

- ▶ Adresă IP
  - ▶ Sursă
  - ▶ Destinație
- ▶ Protocol
  - ▶ IPv4, IPv6, IPX, AppleTalk
  - ▶ TCP, UDP
  - ▶ ICMP
- ▶ Port sau tip
  - ▶ Port sursă sau destinație la TCP sau UDP
  - ▶ Tip de mesaj ICMP



## *ACL-uri pentru filtrare*



# Dezavantaje?

- ▶ Timp de latență mai mare
- ▶ Încărcare suplimentară a echipamentului

## Router dedicat

- Principala funcție: **rutare**
- Permite implementarea funcțiilor de filtrare
- Nu oferă implicit criptare
- Folosește protocoale de nivel 3 și 4 pentru a lua decizii.

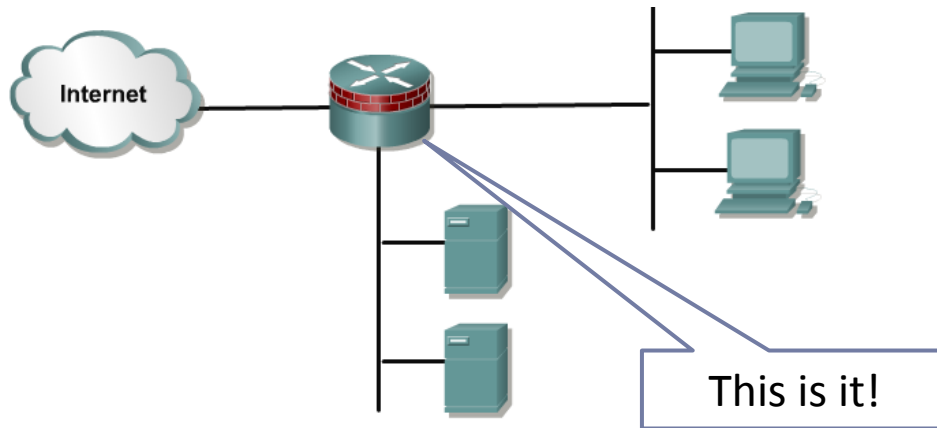
VS

## Firewall dedicat

- Principala funcție: **filtrare**
- Poate ruta, dar suportă mult mai puține facilități
- Oferă criptare HW la rate foarte mari
- Ia decizii pe baza protocoalelor de nivel 3-7
- Server ssh integrat

# Dar ce este un Firewall?

- ▶ Un firewall constă în una sau mai multe mașini care au ca scop prevenirea accesului neautorizat la o rețea.



- ▶ Acestea controlează accesul la servicii atât **din** cât și **în** rețeaua internă
- ▶ ACL-urile sunt folosite pentru a crea firewall-uri între rețeaua internă și cea externă
- ▶ **Demilitarized Zone (DMZ)** conține servicii disponibile din Internet
- ▶ Ruterele firewall trebuie plasate între rețeaua internă și lumea exterioară

# Definiția unui ACL

---

- ▶ O listă de acces conține intrări/reguli pentru controlul accesului
- ▶ Fiecare regulă
  - ▶ identifică diferite tipuri de trafic pe baza unor criterii
  - ▶ specifică acțiunea care trebuie luată în cazul în care criteriul a fost îndeplinit (există match)
    - ▶ Permite traficul : *permit*
    - ▶ Oprește traficul : *deny*

# Parcurgerea unui ACL

---

- ▶ Regulile sunt testate secvențial, linie cu linie, de sus în jos, până se găsește o regulă care să facă match, sau până la sfârșitul listei
  - ▶ La match, se aplică acțiunea, și restul ACL-ului nu se mai verifică
  - ▶ Dacă nu se găsește niciun match, se ajunge la finalul fiecărui ACL , unde există un implicit *deny any*

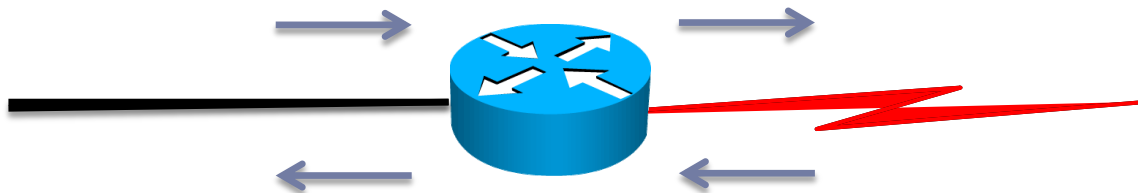
# Aplicarea unui ACL

---

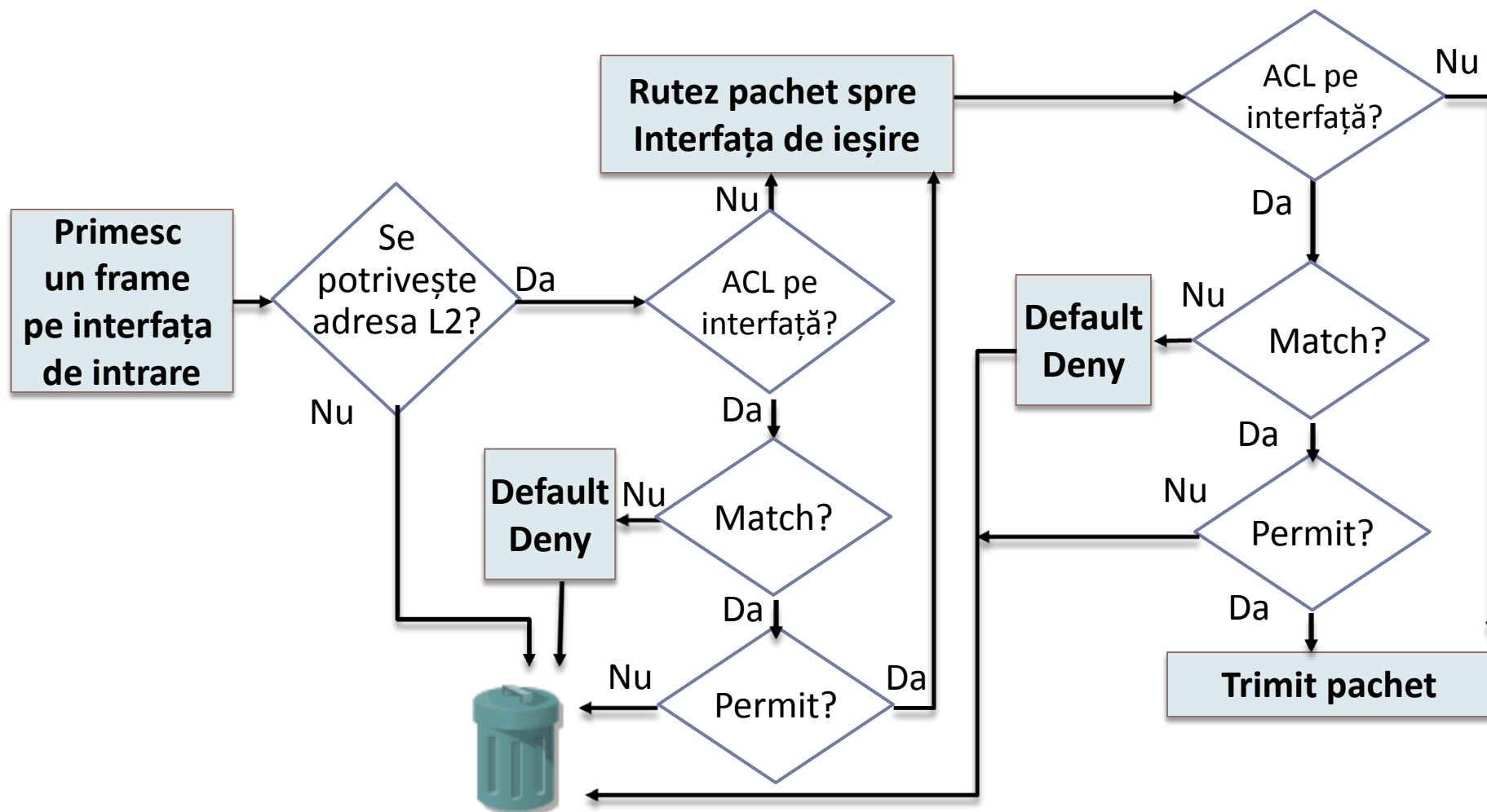
- ▶ **ACL-urile de filtrare pot aplica**
  - ▶ Pentru fiecare protocoale de layer 3 (IP, IPv6 etc.)
  - ▶ Pentru fiecare interfață
  - ▶ Pentru fiecare direcție
    - ▶ Inbound, pentru traficul ce intră
    - ▶ Outbound, pentru traficul ce iese

# Exercițiu: Aplicarea unui ACL

- ▶ Un ruter cu 2 interfețe rulează dual stack (IPv4, IPv6)
- ▶ Care este nr. maxim de ACL-uri de filtrare ce pot fi aplicate
  - ▶ 2 (interfețe) x 2 (protocoale rutate) x 2 (in și out)



# Funcționarea ACL-urilor





# Tipuri de liste de acces

---

- ▶ Liste de acces standard
- ▶ Liste de acces extinse
- ▶ Liste de acces cu nume
  - ▶ standard
  - ▶ extinse

# Tipuri de liste de acces

## ▶ Liste de acces standard

## ▶ Liste de acces extinse

## ▶ Liste de acces cu nume

- ▶ standard
- ▶ extinse

- Identificate printr-un număr între 1 și 99, sau 1300-1999 în IOS-urile mai recente
- Acceptă sau respinge o întreagă suită de protocoale
- Verifică doar **sursa pachetului**
- Trebuie plasat în rețea cât mai aproape de **destinație**.

# Tipuri de liste de acces

- ▶ Liste de acces standard
- ▶ **Liste de acces extinse**
- ▶ Liste de acces cu nume
  - ▶ standard
  - ▶ extinse

- Identificate printr-un număr între 100 și 199, sau 2000-2699 pentru IOS-urile recente
- Pot accepta sau respinge un protocol specific
- Verifică sursa pachetului, destinația, protocolul sau chiar portul
- Trebuie plasat în rețea cât mai aproape de **sursă**.

# Tipuri de liste de acces

- ▶ Liste de acces standard
- ▶ Liste de acces extinse
- ▶ **Liste de acces cu nume**
  - ▶ standard
  - ▶ extinse

- Identificate printr-un nume configurat de administrator
- Pot fi fie **standard**, fie **extinse**
- Oferă flexibilitate mai mare decât listele standard sau extinse
- Recomandate să fie folosite față de cele cu număr

# Wildcard mask

---

- ▶ O mască ce se suprapune peste o adresă IP
- ▶ Identifică partea comună a unor adrese IP
- ▶ Reprezintă un șir de 32 de biți de 1 și 0
  - ▶ **Bitul 0** – face match
  - ▶ **Bitul 1** – ignoră valoarea bitului din IP
- ▶ Poate fi privită ca și inversul măștii de rețea, însă poate fi folosită și pentru a identifica altfel

# Wildcard mask

---

- ▶ Se pot folosi 2 cuvinte cheie în ACL-uri:
  - ▶ **any** – înseamnă adresa IP 0.0.0.0 și WM 255.255.255.255, toate IP-urile vor face match
  - ▶ **host** – testează egalitatea cu o adresă de host, echivalent cu WM 0.0.0.0

# Wildcard mask - exemplu

```
Router(config)#access-list 10 permit 172.16.0.0 0.0.255.255
```

▶ În acest exemplu, ruterul va verifica doar primii 16 biți din adresele IP și îi va compara cu cei din adresa IP. Această declarație va permite traficul având ca sursă 172.16.\*.\*

- ▶ Biții de 0 – fac match
- ▶ Biții de 1 – sunt ignorați

172.16.0.0    10101100.00010000.00000000.00000000

0.0.255.255    00000000.00000000.11111111.11111111

# ACL-uri clasice

---

- ▶ Standarde sau Extinse
  - ▶ Tipul este dat de numărul (ID-ul) listei
- ▶ Grupate în funcție de numărul (ID) comun
- ▶ Adăugate linie cu linie dar întotdeauna la sfârșit
- ▶ Nu se poate șterge o singură linie din ACL



# ACL-uri clasice standard

- ▶ Filtrează pachetele doar în funcție de **sursă**
- ▶ Numărul asociat unui astfel de ACL trebuie să fie între 1 și 99, sau, în versiunile mai recente de IOS, între 1300 și 1999

Fără WM  
specificat,  
WM = 0.0.0.0

```
R(config)# access-list 50 deny 172.16.1.1  
R(config)# access-list 50 permit 172.16.0.0 0.0.255.255
```

Număr între 1 și 99,  
Sau între 1300 și 1999

Deny sau  
Permit

Wildcard  
Mask

# ACL-uri clasice extinse

- ▶ Filtrează pachetele în funcție și de **sursă** și de **destinație**. De asemenea, pot filtra pachete și în funcție de **protocol** și de **port**
- ▶ Numărul asociat unui astfel de ACL trebuie să fie între 100 și 199; în versiunile mai recente de IOS se pot folosi și numere între 2000 și 2699

Protocol

IP Sursă

IP Destinație

Port Destinație

```
access-list 101 permit tcp 172.16.6.0 0.0.0.255 any eq telnet
```

Permite Telnet-ul de la toate host-urile din rețeaua 172.16.6.0/24

# Aplicarea unui ACL

- ▶ Crearea listelor de acces este doar *jumătate din muncă*
- ▶ Cealaltă jumătate constă în **aplicarea ACL-urilor pe interfețe** (pentru filtrare)

```
R(config)# interface fastethernet 0/0
R(config-if)# ip access-group ?
<1-199>          IP access list (standard or extended)
<1300-2699>      IP expanded access list (standard or extended)
WORD             Access-list name
R(config-if)#ip access-group 10 ?
in      inbound packets
out     outbound packets
```

# Editarea unui ACL clasic

---

- ▶ Pentru a edita un ACL clasic standard sau extended:
  - ▶ Copiați ACL-ul într-un fișier text
  - ▶ Stergeți ACL-ul din fișierul de configurare al ruter-ului folosind 'no' și declarația ACL-ului
  - ▶ Faceți modificările necesare în fișierul text
  - ▶ Copiați pe ruter ACL-ul modificat, în global configuration mode

**sau...**

# Named ACLs

---

- ▶ Nu mai sunt folosite numere pentru a diferenția ACL-uri, ci nume
  - ▶ Numele sunt mai intuitive decât numerele
    - ▶ 254 vs „DMZ\_IN\_FILTER”
- ▶ Este posibilă numerotarea regulilor ce sunt adăugate, pentru ca apoi să se poată face modificări fără a șterge complet lista

# Named ACLs - Exemplu

```
R(config)#ip access-list extended FILTER_LAN_IN  
R(config-ext-nacl)#20 permit ip any any
```

Dacă am uitat 2 reguli ce trebuiau definite înainte..

```
R(config-ext-nacl)#5 permit icmp host 10.0.0.0 any  
R(config-ext-nacl)#10 deny icmp any any
```

Dacă am greșit regula de pe linia 5...

```
R config-ext-nacl)#no 5  
R config-ext-nacl)#5 permit icmp host 10.0.0.1 any
```

După definire, pot aplica ACL-ul pe interfață

```
R(config)#interface fastEthernet 0/1  
R(config-if)#ip access-group FILTER_LAN_IN in
```

# Un caz special

---

- ▶ ACL-urile standard pot fi și ele folosite pentru a gestiona traficul pentru conexiunile la distanță
- ▶ Soluția:

```
R(config)#line vty 0 4  
R(config-line)#access-class access-list-number {in | out}
```

# ACL remarks

---

- ▶ Permit trafic către rețeaua A și opresc trafic către host B

```
R(config)# access-list 50 remark permit traficul spre A
R(config)# access-list 50 permit 172.16.0.0 0.0.255.255
R(config)# access-list 50 remark opresc traficul spre B
R(config)# access-list 50 deny 192.168.10.15
```

- ▶ Un comentariu este limitat la 100 de caractere



# Log-uri

---

- ▶ Generează un mesaj ce cuprinde
  - ▶ nr. listei
  - ▶ dacă a fost acceptat/respins pachetul
  - ▶ sursa
  - ▶ nr. de pachete
- ▶ Mesajul este generat pentru primul pachet care corespunde unei reguli, iar apoi la intervale de 5 minute

```
R(config)# access-list 50 permit 172.16.0.0 0.0.255.255 log
```



# Verificarea ACL-urilor

- Comenzi de **show** pentru verificarea **conținutului** și pentru **poziționarea** ACL-urilor:

Comanda	Descriere
<code>show ip interface</code>	Informații privind numărul de ACL-uri de intrare și ieșire
<code>show access-list</code>	Afișează conținutul ACL-urilor configurate pe router
<code>show running-config</code>	Afișează, printre altele, poziționarea și conținutul ACL-urilor configurate

# Exemple de ACL-uri

- ▶ O listă de acces care să permită doar traficul de la stația 193.230.2.1

```
R(config)# access-list 1 permit host 193.230.2.1
```

sau

```
R(config)# access-list 2 permit 193.230.2.1 0.0.0.0
```

sau

```
R(config)# access-list 3 permit 193.230.2.1
```

- ▶ Soluție folosind ACL extins

```
R(config)# access-list 101 permit ip host 193.230.2.1 any
```

# Exemple de ACL-uri

- Construiți și aplicați pe interfața ethernet 1 o listă de acces ce va permite doar traficul inițiat de la adresele 11.2.2.90 și 11.2.2.91.

```
R(config)# acces-list 18 permit host 11.2.2.90
R(config)# acces-list 18 permit host 11.2.2.91
sau
R(config)# acces-list 18 permit 11.2.2.90 0.0.0.1

R(config)# interface ethernet 1
R(config-if)# ip acces-group 18 in
```

# Exemple de ACL-uri

---

- Care este efectul următoarelor linii?

```
R(config)# interface ethernet 4
R(config-if)# ip access-group 199 out

R(config)# access-list 199 permit ip any any
R(config)# access-list 199 deny ip 106.45.0.0 0.0.255.255 any
R(config)# access-list 199 deny tcp any 44.7.12.224 0.0.0.15 eq
ftp
R(config)# access-list 199 deny udp 23.145.64.0 0.0.0.255 host
1.2.3.4 eq rip
```

## *Network Address Translation*



# Problemă

---

- ▶ Creștere rapidă a Internetului



- ▶ Deficit de adrese IP disponibile

- ▶ Soluția:

- ▶ Adrese private +
- ▶ Network Address Translation


# NAT

- ▶ Un standard Internet care permite unui LAN să folosească un set de adrese IP pentru traficul intern, și un set diferit de adrese pentru traficul extern.
- ▶ Adresa IP privată a sursei unui pachet este translatată într-o adresă publică (rutabilă) de către gateway

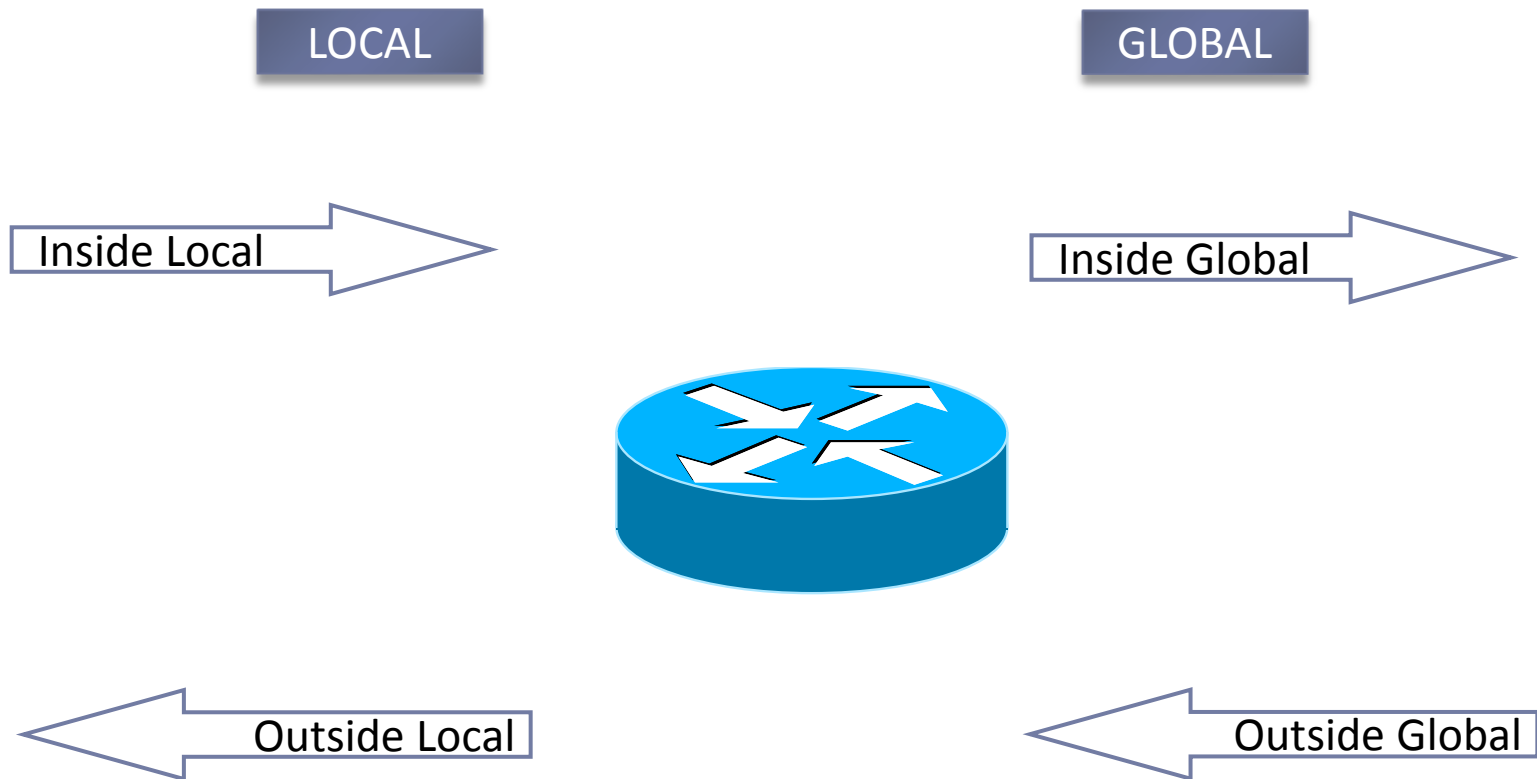


# Adrese IP private

Clasa	Intervalul de adrese	Prefix CIDR
A	10.0.0.0 – 10.255.255.255	10.0.0.0/8
B	172.16.0.0 – 172.31.255.255	172.16.0.0/12
C	192.168.0.0 – 192.168.255.255	192.168.0.0/16

- ▶  aka RFC1918
- ▶ Pot fi folosite de oricine, fără restricții
- ▶ De ce nu sunt rutabile în Internet?

# Terminologia NAT



- ▶ Statică

- ▶ mapare constantă 1 la 1
- ▶ utilă pentru servere web ce au nevoie de o adresă accesibilă oricând

- ▶ Dinamică

- ▶ oferă adresele dinamic, pe baza unui pool de adrese
- ▶ regula: primul venit, primul servit

- ▶ Port Address Translation
  - ▶ a.k.a. NAT overloading, NAPT, masquerading
- ▶ Permite asocierea unei adrese IP publice cu un grup de adrese private
- ▶ Se bazează pe schimbarea portului sursă
  - ▶ Modificări si la nivelul 3 și la nivelul 4

# Tipuri de translatare

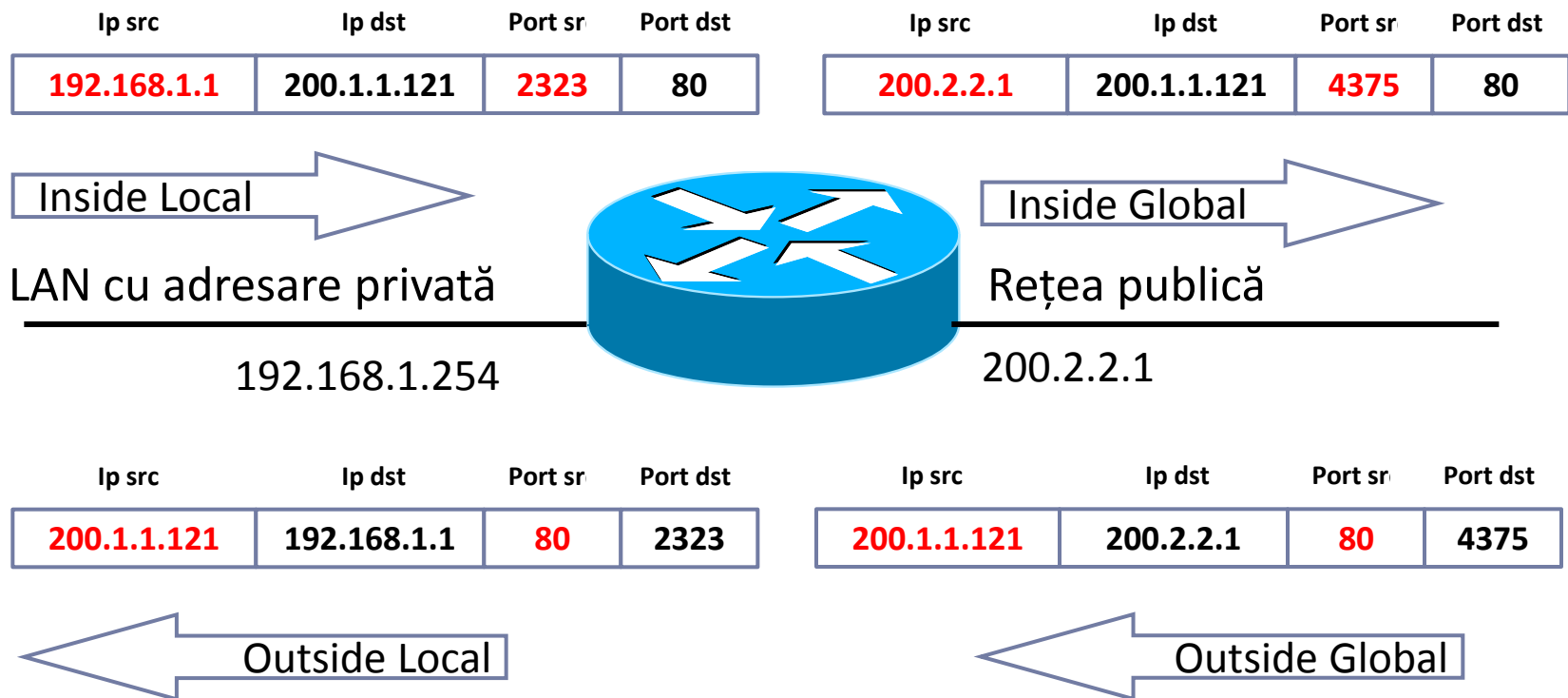
## NAT

- Network Address Translation
- gateway-ul are definită o listă de adrese IP publice
- mapează **o adresă privată pe o adresă publică**
- numărul de stații din rețeaua privată ce pot accesa Internetul limitat de dimensiunea listei de adrese publice

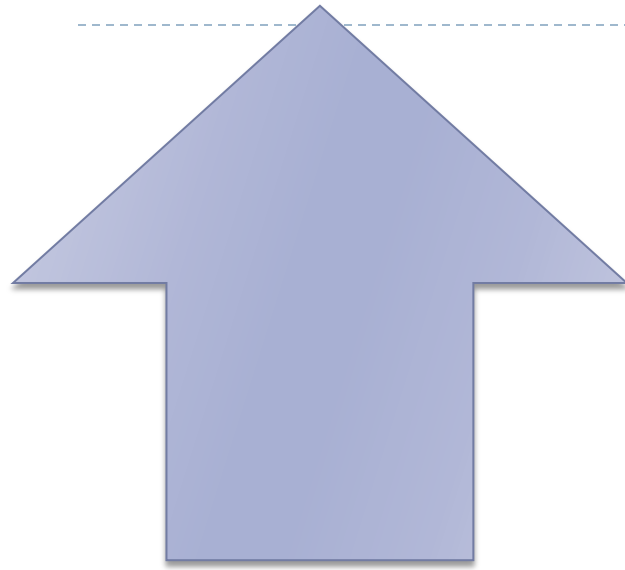
## PAT

- Port Address Translation
- gateway-ul va înlocui toate adresele private cu adresa sa publică
- **mai multe adrese private pe o singură adresă publică**
- folosește translatarea la nivel de port pentru a diferenția între diferitele translații
- Nu este făcut de toate ruterele în hardware

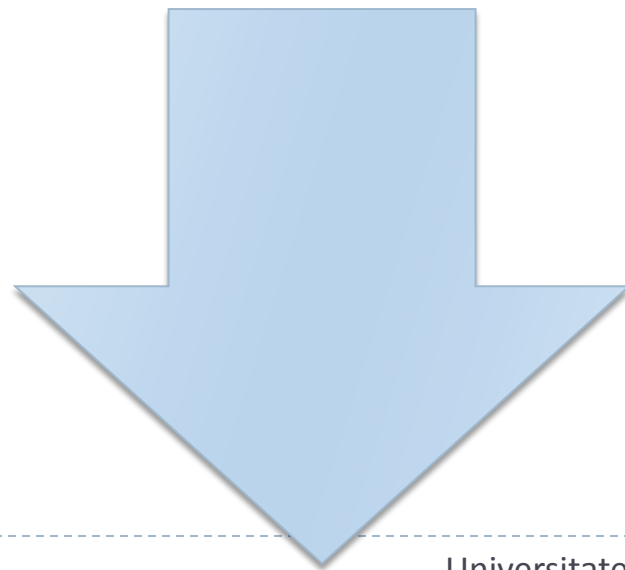
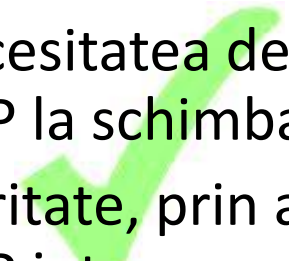
# Funcționare PAT



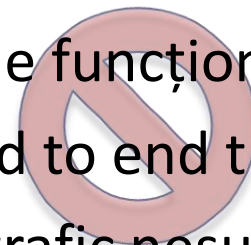
# PRO & CONTRA



- ▶ Conservarea adreselor IP disponibile
- ▶ Elimină necesitatea de schimbare a adreselor IP la schimbarea ISP-ului
- ▶ Oferă securitate, prin ascunderea adreselor IP interne



- ▶ Latență mărită
- ▶ Pierdere de funcționalitate
- ▶ Loss of end to end traceability
- ▶ Tipuri de trafic nesuportate: SNMP, update-uri de rutare



# Pași configurare NAT

---

- ▶ Ce translatez?
  - ▶ stabilire ce IP-uri private sunt translatate printr-un ACL
  
- ▶ În ce translatez?
  - ▶ pool de adrese (NAT)
  - ▶ IP intrerfață (PAT)
  
- ▶ În ce sens translatez?
  - ▶ Stabilire cine este inside și cine outside





# Configurare NAT

## ► Translatare statică

```
R(config)# ip nat inside source static <localIP> <globalIP>
```

## ► Translatare dinamică

```
R(config)# access-list 1 permit 10.0.0.0 0.255.255.255  
R(config)# ip nat pool <name> <start-ip> <end-ip>  
          {netmask <netmask> | prefix-length <prefix>}  
R(config)# ip nat inside source list 1 pool name
```

## ► PAT

```
R(config)# ip nat inside source list 1 interface intf  
                                                          overload
```

# Configurare NAT

---

- ▶ La nivel de interfață
  - ▶ pe interfața către rețeaua privată

```
R(config-if)# ip nat inside
```

- ▶ pe interfața către Internet

```
R(config-if)# ip nat outside
```

# Verificarea NAT

- Comenzi de **show** pentru verificarea **conținutului** și pentru **poziționarea** ACL-urilor:

Comanda	Descriere
<code>show ip translations</code>	Afișează tabela cu translațiile NAT
<code>show ip nat statistics</code>	Afișează informații statistice legate de translațiile NAT
<code>clear ip nat translations *</code>	Șterge tabela de translații dinamice
<code>debug ip nat</code>	Afișează informații pentru fiecare pachet translatat de ruter

## *Tunelare*



# Tunelare

---

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



# Tipuri de tunele

Application	HTTP
Transport	SSL VPN
Network	IPIP, 6to4, SIT, IPSec, <b>GRE</b>
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

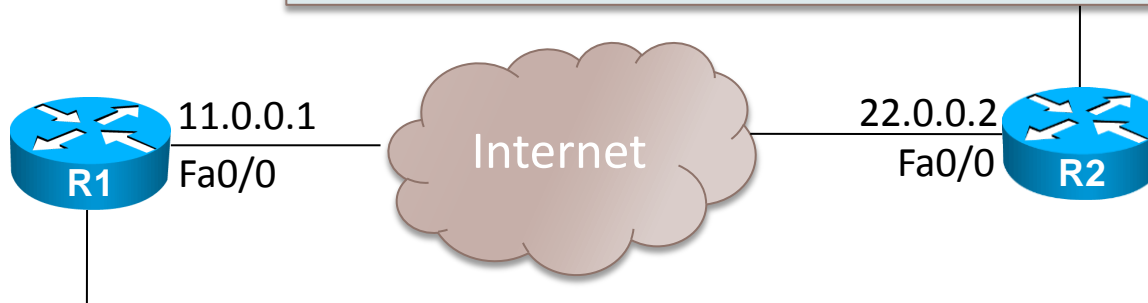
---

- ▶ Creerea 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
R1(config-if)# tunnel source FastEthernet0/0
R1(config-if)# tunnel destination 11.0.0.1
```

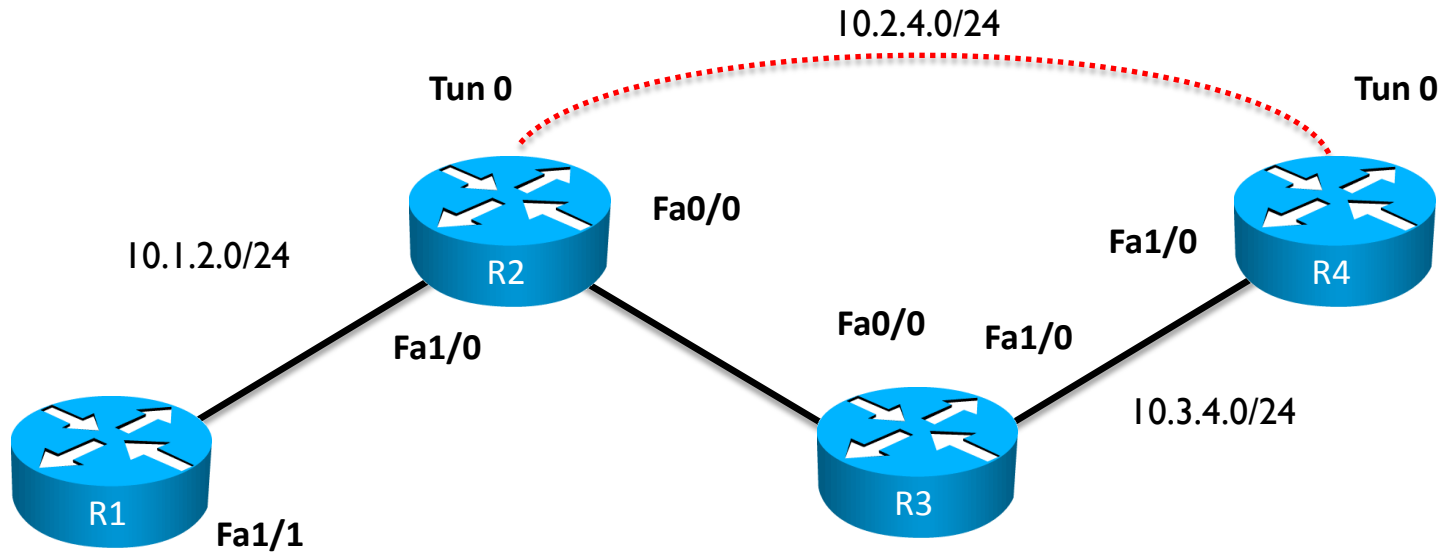


```
R1(config)# ip route 22.0.0.0 255.255.255.0 Fa0/0
R1(config)# interface Tunnel0
R1(config-if)# ip address 12.0.0.1 255.255.255.0
R1(config-if)# tunnel source FastEthernet0/0
R1(config-if)# tunnel destination 22.0.0.2
```

- ▶ Access Lists
  - ▶ Ce este un ACL
  - ▶ Funcționarea ACL-urilor
  - ▶ Tipuri de liste de acces
  - ▶ Exemple de configurare
- ▶ Network Address Translation
  - ▶ De ce este nevoie?
  - ▶ Concepte NAT
  - ▶ Configurare NAT
- ▶ Tunelare
  - ▶ Tunelare GRE



# POC: ACL, NAT, GRE



- ▶ Configurați astfel încât R4 să nu accepte pachete de la R1
- ▶ Configurați pe R3 astfel încât R2 să nu poată accesa serviciul HTTP de pe R4
- ▶ Configurați astfel încât doar R2 să se poată autentifica pe terminalul virtual al lui R4
- ▶ Reparați conectivitatea de la R2 spre serviciul HTTP de pe R4 fără a șterge ACL-ul de pe R3. Hint: Tunnel
- ▶ Configurați ca R1 să se poată autentifica pe terminalul virtual al lui R4, fără a șterge/modifica ACL-ul. Hint: NAT