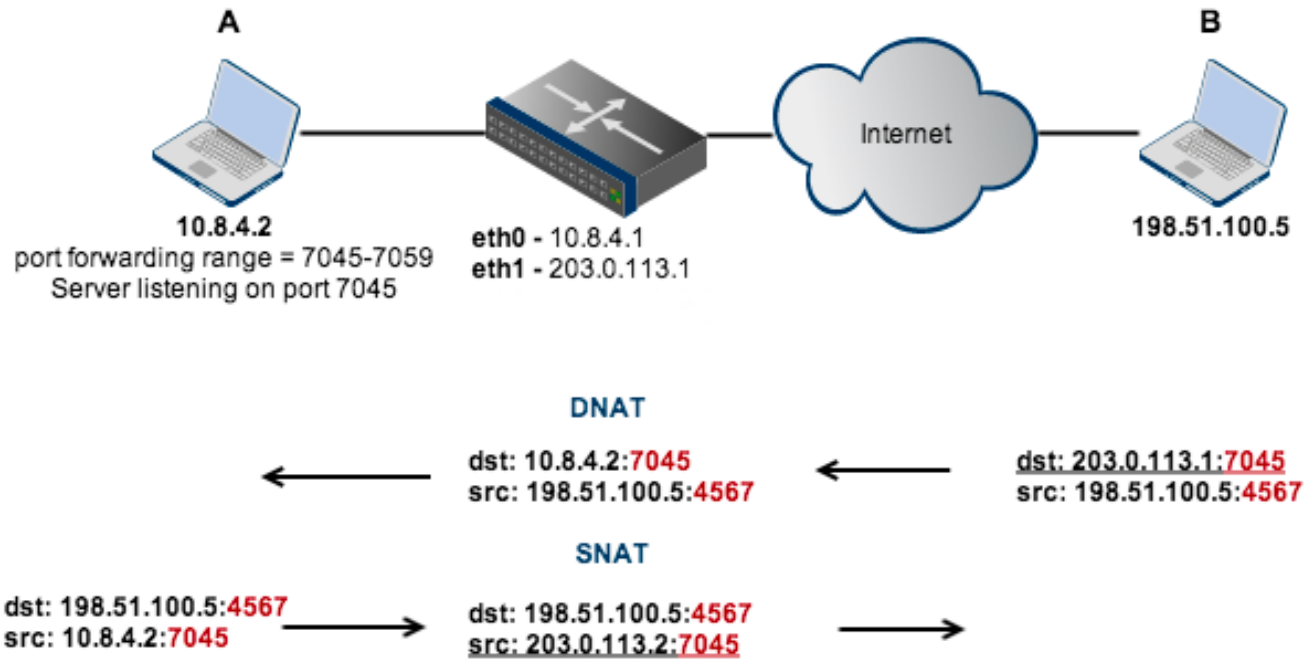
Sieci komputerowe

Tadeusz Kobus, Maciej Kokociński
Instytut Informatyki, Politechnika Poznańska

Translacja adresów w Linuksie

Network Address Translation (NAT)



Translacja źródłowa – Source NAT (SNAT)

Pozwala na dostęp do sieci z adresacją publiczną (np. Internet) urządzeniom z sieci z adresacją prywatną.

Maskarada:

- uproszczony tryb translacji źródłowej,
- adres źródłowy jest automatycznie ustawiany na adres interfejsu, którym pakiet opuszcza router,
- trzeba podać przynajmniej interfejs wyjściowy!

```
# iptables -t nat -A POSTROUTING -o eth1 -j MASQUERADE
```

Klasyczny SNAT:

- wymaga podania adresu (lub adresów) na jakie ma być zmieniane źródło

```
# iptables -t nat -A POSTROUTING -j SNAT --to-source 203.0.113.2
```

Translacja źródłowa – Source NAT (SNAT)

Problematyczny scenariusz:

- dwa komputery z sieci prywatnej A (10.8.4.2) i A' (10.8.4.3),
- A i A' chcą łączyć się z tym samym sewerem B (198.51.100.5),
- A i A' losują ten sam numer portu dla połączenia z B, np. 56000,
- po SNAT na routerze, nie można byłoby okreslić, do którego komputera (A czy A') należy skierować pakiety zwrotne od B.

Rozwiązanie:

- router dokonuje czasami (np. dla A') zarówno translacji IP jak i portu źródłowego,
- po SNAT na routerze, pakiety generowane przez A' będą wyglądać tak, jakby były nadane przez router z adresu 203.0.113.2 i portu np. 56007.

Translacja docelowa – Dest. NAT (DNAT)

Pozwala na dostęp świata do urządzeń znajdujących się w sieci prywatnej.

W szczególności pozwala na przekierowywanie portów między hostami.

```
# iptables -t nat -A PREROUTING -i eth1 -p tcp --dport 80 -j DNAT \
                                          --to-destination 10.8.4.2
# iptables -t nat -A PREROUTING -i eth1 -p tcp --dport 22 -j DNAT \
                                          --to-destination 10.8.4.2:8008
```

SNAT vs DNAT

SNAT i DNAT to zupełnie dwie niezależne rzeczy!

- zarówno SNAT i DNAT dokonują translacji adresów źródłowych jak i adresów docelowych, ale
- dla danego żądania i odpowiedzi:
  - SNAT: najpierw translacja adresów źródłowych w żądaniach, a później translacja adresów docelowych w odpowiedziach,
  - DNAT: najpierw translacja adresów docelowych w żądaniach, a później translacja adresów źródłowych w odpowiedziach,
- może być SNAT bez DNAT (patrz punkt 2 następnego Zadania),
- może być DNAT bez SNAT (patrz punkty 6-7 następnego Zadania).

Zadanie 1 (1)

Do rozwiązania w rzędach, min. 3 komputery PC1–PC3.

Konfiguracja podstawowa:

- dwie oddzielne sieci prywatne PC1–PC2 (na `p4p1`) i PC2–PC3 (na `p4p2`)
- PC1, PC3: dla pewności wyłączony interfejs `em1`,
- PC2: ustawione przekazywanie pakietów: `sysctl -w net.ipv4.conf.all.forwarding=1`,
- PC3: PC2 jako domyślna brama (adres interfejsu `p4p2`).

Zadanie 1 (2)

1. Ustaw na PC2 maskaradę adresów tak, żeby PC1 i PC3 mogły korzystać z Internetu (PC1 nie ma routingu domyślnego, więc w tym momencie Internet nie będzie działać). Uwaga na `/etc/resolv.conf`.
2. Sprawdź czy PC3 może łączyć się z PC1. Jeśli nie, napraw to używając SNAT.
3. Używając wiresharka zobacz jakie adresy ma ustawiony pakiet wysłany z PC1 do PC3 na każdym komputerze.
4. Zmień TTL pakietów przechodzących przez PC2 z PC3 do PC1 na 1, zaobserwuj (w wiresharku) co się dzieje.
5. Włącz na PC3 program nasłuchujący na wybranym porcie.
6. Skonfiguruj PC2 tak, by PC1 mógł połączyć się z programem działającym na PC3 używając adresu PC2.
7. Skonfiguruj PC2 tak, by PC1 mógł połączyć się z serwerem SSH działającym na PC3 używając adresu PC2 i portu innego niż 22.
8. Używając wiresharka zobacz jak modyfikowane są pakiety z powyższych poleceń.

Rozwiązanie – przykładowe komendy

Konfiguracja:

```
             PC1                  PC2                  PC3
 em1:                        150.254.32.66/26
p4p1:  192.168.1.1/24   --  192.168.1.2/24
p4p2:                       192.168.111.2/24   -- 192.168.111.1/24
```

PC2:

```
# iptables -t nat -A POSTROUTING -o em1 -j MASQUERADE
# iptables -t nat -A POSTROUTING -s 192.168.111.0/24 -d 192.168.1.0/24 \
                                 -j SNAT --to-source 192.168.1.2
# iptables -t mangle -A POSTROUTING -s 192.168.111.0/24 -d 192.168.1.0/24 \
                                 -j TTL --ttl-set 1
# iptables -t nat -A PREROUTING -p tcp --dport 74 -j DNAT \
                                 --to-destination 192.168.111.1:74
# iptables -t nat -A PREROUTING -p tcp --dport 1234 -j DNAT \
                                 --to-destination 192.168.111.1:22
```

Rozwiązanie – aktualny stan tablicy `nat`

```
# iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target      prot opt source               destination
DNAT        tcp  --  anywhere             anywhere   tcp dpt:74 \
                                                    to 192.168.111.1:74
DNAT        tcp  --  anywhere             anywhere   tcp dpt:1234 \
                                                    to 192.168.111.1:22
Chain INPUT (policy ACCEPT)
target      prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target      prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target      prot opt source               destination
MASQUERADE  all  --  anywhere             anywhere
SNAT        all  --  192.168.111.0/24     192.168.1.0./24 to:192.168.1.2
# iptables -t mangle -L
...
Chain POSTROUTING (policy ACCEPT)
target      prot opt source               destination
TTL         all  --  192.168.111.0./24    192.168.1.0./24 TTL set to 1
```