Bezpieczeństwo systemów informatycznych

ĆWICZENIE Zapory sieciowe

1. Filtracja pakietów

Filtry pakietów sprawdzają nagłówki pakietów przepływających przez stos protokołów. Decydują o ich losie, **akceptując** (ang. *accept*) lub **odrzucając** (ang. *drop*) poszczególne pakiety. Taką funkcjonalność posiadają np. routery filtrujące.

1.1 Filtracja pakietów w systemie Linux (netfilter/iptables)

W systemie Linux filtracja pakietów IP jest wbudowana w jądro systemu operacyjnego. Odpowiedzialny jest za to moduł netfilter oferujący filtrację pakietów przychodzących, wychodzących oraz routowanych (w tzw. tablicy filter). Narzędzie iptables zarządza listami reguł filtracji, nazywanymi **łańcuchami**. Podstawowe łańcuchy w tablicy filter noszą nazwy INPUT (dla strumienia pakietów przychodzących, docelowych), OUTPUT (strumień pakietów wychodzących) i FORWARD (routing). Reguły filtracji utrzymywane są w pamięci jądra, a ich dostępność pomiędzy restartami systemu można zapewnić przechowując wybrane konfiguracje w plikach (wykorzystując skrypty iptables-save i iptables-restore).

Działanie narzędzia iptables zobrazujemy na następujących przykładach poleceń:

- wyświetlenie reguł filtracji dla łańcucha FORWARD:
```
iptables -L FORWARD
```
`-L` = List rules

- ustawienie domyślnej polityki obsługi pakietów dla łańcucha FORWARD:
```
iptables -P FORWARD DROP
```
`-P` = Policy
co będzie zgodne z regułą domyślnej odmowy dostępu.

- zezwolenie na przepuszczanie pakietów protokołu TCP:
```
iptables -A FORWARD -p tcp -j ACCEPT
```
`-A` = Add rule
`-p` = protocol
`-j` = job to do

- usunięcie w/w reguły z łańcucha FORWARD:
```
iptables -D FORWARD -p tcp -j ACCEPT
```
`-D` = Delete rule

- wstawienie reguły na 1-sze miejsce w łańcuchu FORWARD:
```
iptables -I FORWARD 1 -p tcp -j ACCEPT
```
`-I` = Insert rule

- usunięcie reguły nr 3 z łańcucha FORWARD:
```
iptables -D FORWARD 3
```

- usunięcie wszystkich reguł łańcucha FORWARD:
```
iptables -F FORWARD
```
`-F` = Flush chain

- zezwolenie na nadawanie pakietów protokołu ICMP w pętli zwrotnej:
```
iptables -A OUTPUT -p icmp -s 127.0.0.1 -j ACCEPT
```
`-s` = source address

- odrzucanie pakietów protokołu ICMP przychodzących z sieci 199.1.2.0:

```
iptables -A INPUT -p icmp -s 199.1.2.0/255.255.255.0 -j DROP
```

lub:

```
iptables -A INPUT -p icmp -s 199.1.2.0/24 -j DROP
```

- odrzucanie pakietów innych niż TCP przychodzących na interfejs eth0 skierowanych do innej sieci niż 199.1.2.0:

```
iptables -A INPUT -i eth0 -p ! tcp -d ! 199.1.2.0/24 -j DROP
```

`-d` = destination address
`-i` = input interface

- odrzucanie routowanych pakietów TCP przychodzących na interfejs eth0 skierowanych do innej sieci niż 199.1.2.0:

```
iptables -A FORWARD -i eth0 -p tcp -d ! 199.1.2.0/24 -j DROP
```

- odrzucanie wychodzących pakietów TCP z ustawionymi tylko flagami SYN,ACK (przy weryfikacji wszystkich flag nagłówka):

```
iptables -A OUTPUT -p tcp --tcp-flags ALL SYN,ACK -j DROP
```

1.2 NAT i maskarada

Narzędzie iptables umożliwia również konfigurację funkcji translacji (ukrywania) adresów – NAT. Funkcje NAT realizuje w Linuksie moduł jądra o nazwie iptable_nat (tablica o nazwie nat)

Na ogół rozróżnia się translację adresów źródłowych (SNAT) i docelowych (DNAT). Translacja SNAT (maskarada) na routerze pozwala ukryć rzeczywiste adresy sieci wewnętrznej, np. w celu utajnienia ich przed światem zewnętrznym lub przy możliwości stosowania tylko ograniczonego zakresu adresów źródłowych przydzielonych w sieci publicznej. Z kolei translacja DNAT może być użyteczna przy stosowaniu transparentnych usług *proxy*.

Oto przykłady konfiguracji dla:

- pojedynczego adresu wyjściowego:

```
iptables -t nat -A POSTROUTING -o eth0 \
             -j SNAT --to 199.1.1.1
```

- predefiniowanej puli portów TCP:

```
iptables -t nat -A POSTROUTING -o eth0 -p tcp \
             -j SNAT --to 199.1.1.1:8000-9000
```

- oraz przedziału adresów wyjściowych:

```
iptables -t nat -A POSTROUTING -o eth0 \
             -j SNAT --to 199.1.1.1-199.1.1.99
```

Przy wykorzystaniu DHCP można konfigurację uprościć, np.:

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

1.3 Rozszerzenia iptables

Moduły iptables rozszerzają możliwości zapory firewall. Poniżej krótko przedstawimy większość z najciekawszych modułów. Większość z modułów ładowana jest opcją `-m`, jednak część jest używana automatycznie przy użyciu w regule wybranego protokołu, np. `-p tcp`.

1.3.1 Przykładowe moduły

1. Określające typ pakietu:
   - icmp – dopasowuje się do pakietów protokołu ICMP,
   - tcp – dotyczy pakietów protokołu TCP,
   - udp – dotyczy pakietów protokołu UDP,
   - unclean – dotyczy pakietów zniszczonych i niepoprawnych.
2. Analizujące pola nagłówka pakietu:
   - ttl – sprawdza pole TTL (*Time To Live*),
   - tos – sprawdza pole ToS (*Type of Service*),
   - dscp – dopasowuje się do 6 bitów DSCP znajdujących się w polu ToS,
   - ecn – sprawdza bit ECN w nagłówku,
   - ipv4options – sprawdza różne opcje nagłówka pakietu, np. *source routing*, *record route*,
   - tcpmss – sprawdza pole MSS (*Maximum Segment Size*).
3. Rozszerzone sprawdzanie pakietów:
   - length – sprawdzające wielkość pakietu,
   - string – dopasowujące się do zawartości pakietu w polu DATA,
   - ipp2p – dopasowujące się do ruchu generowanego przez aplikacje typu P2P.
4. Określające limity:
   - connlimit – umożliwia określenie maksymalnej liczby asocjacji (połączeń),
   - connrate – umożliwia określenie maksymalnego/minimalnego transferu,
   - hashlimit – umożliwia określenie transferu z podziałem na poszczególne asocjacji.
5. Nadzorujące połączenia (stanowość):
   - state i conntrack – umożliwiają zidentyfikowanie istniejących połączeń oraz nowych,
   - helper – pozwala określić “pomocnika” do przekazywania połączenia, takim pomocnikiem jest np. moduł pozwalający przekazywać połączenia ftp-data w trybie aktywnym.
6. Znakujące pakiety:
   - mark – umożliwia znalezienie oznakowanego pakietu,
   - connmark – umożliwia znalezienie każdego pakietu związanego z oznakowanym połączeniem.
7. Ułatwiające tworzenie reguł:
   - iprange – umożliwia wpisanie zakresu adresów IP,
   - mac – umożliwia sprawdzenie adresu sprzętowego karty sieciowej MAC,
   - multiport – umożliwia wpisanie zakresu portów,
   - owner – dla lokalnych połączeń umożliwia określenie który użytkownik wysłał dany pakiet,
   - pkttype – określa typ pakietu (unicast, multicast, broadcast)
   - set – wykorzystanie stworzonych zbiorów adresów (polecenie ipset),
   - time – określenie daty i/lub czasu,
   - comment – umożliwia dopisanie komentarza do każdej reguły.
8. Określające częstość:
   - limit – określa jak często reguła będzie dopasowana, np. 3 razy na sekundę,
   - nth – określa co jaką liczbę pakietów będzie dopasowana reguła, np. co 5 pakiet,
   - random – określa dopasowanie reguły do pakietu z zadanym prawdopodobieństwem, np. 50% pakietów.
9. Monitorujące:
   - recent – tworzy statystyki pakietów pasujących do reguły w katalogu `/proc/net/xt_recent/`,
   - quota – określa ograniczenie ilości (*quota*) określonych pakietów.

10. Dotyczące IPsec:
    - ah – dopasowują się do pola SPI w nagłówku AH pakietu,
    - esp – dopasowują się do pola SPI w nagłówku ESP pakietu,
    - policy – dopasowują się do polityki bezpieczeństwa dotyczącej danego pakietu.
11. Inne:
    - condition – pozwala warunkować stosowanie reguły
    - osf – odczytuje pasywnie "odcisk palca" (*fingerprint*) konkretnego adresu IP i zapisuje w pliku `/proc/sys/net/ipv[46]/osf`
    - psd – pozwala wykryć skanowanie portów TCP i UDP.

1.4 Cel (TARGET)

Cel określa co zostanie zrobione z pakietem pasującym do reguły. Podstawowymi celami są akceptacja (ACCEPT) i odrzucenie (DROP) pakietu, lista dostępnych celów jest pokaźna.

Lista predefiniowanych celów została podzielona na następujące kategorie:

1. Zmiana adresów IP (→ tablica nat):
    - SNAT – Source NAT,
    - MASQUERADE – ukrywanie źródłowych adresów IP,
    - DNAT – Destination NAT,
    - REDIRECT – przekierowanie ruchu na lokalny komputer,
    - BALANCE – podobne do DNAT, ale równomiernie rozkłada obciążenie na wiele adresów IP,
    - SAME – podobne do DNAT/SNAT, ale zawsze przydziela te same adresy IP konkretnym klientom,
    - NETMAP – statyczna zamiana adresów całych podsieci.
2. Modyfikacje nagłówka pakietu (→ tablica mangle):
    - TTL – ustawienie pola TTL,
    - TCPMSS – ustawienie pola MSS,
    - TOS – ustawienie pola ToS,
    - DSCP – umożliwia zmianę 6 bitów DSCP pola ToS,
    - ECN – umożliwia usunięcie bitu ECN,
    - IPV4OPTSSTRIP – usunięcie wszystkich opcji IPv4 z nagłówka pakietu.
3. Znakowanie pakietów:
    - MARK – znakuje pakiety,
    - CONNMARK – znakuje połączenia,
    - IPMARK – znakowanie pakietów na podstawie adresu IP.
4. Śledzenie pakietów:
    - TRACE – włącza śledzenie połączenia,
    - NOTRACK – wyłącza śledzenie połączenia,
    - TARPIT – przechwytuje połączenia i zawiesza je, w celu zabezpieczenie się przez skanowaniem.
5. Logowanie:
    - LOG – logowanie pakietów (systemowe),
    - ULOG – logowanie w przestrzeni użytkownika.
6. Inne:
    - REJECT – odrzuca pakiety wysyłając określony kod błędu do nadawcy,
    - ROUTE – pozwala nadpisać wpisy tablicy tras,
    - SET – dodaje/usuwa adresy z zakresów (polecenie ipset),
    - XOR – pozwala zastosować proste zabezpieczenie danych pakietu, poprzez wykonanie funkcji XOR na danych i określonym kluczu.

1.5 Dodatkowe informacje

Szczegółowe informacje odnoście każdego z modułów i celów można uzyskać z plików pomocy.

Uzyskanie pomocy o module:

```
iptables -m <modul> -h
```

Uzyskanie pomocy o celu:

```
iptables -j <cel> -h
```

1.6 Przykłady

Blokada pakietów o rozmiarze mniejszym niż 16 bajtów:

```
iptables -A FORWARD -m length --length 0:16 -j DROP
```

Utworzenie statystyk ruchu z portem 22:

```
iptables -A FORWARD -p tcp --dport 22 -m recent --name ssh --set
```

Akceptacja 3 pakietów na minutę:

```
iptables -A FORWARD -m limit --limit-burst 3 --limit 3/min -j ACCEPT
```

Blokada więcej niż 2 jednoczesnych połączeń na port 22 ze zwrotnym komunikatem ICMP:

```
iptables -A FORWARD -p tcp --syn --dport 22 -m connlimit \
--connlimit-above 2 -j REJECT --reject-with icmp-host-unreachable
```

Umożliwienie tylko jednego na minutę nowego połączenia z portem 22, dla każdego z poszczególnych adresów źródłowych:

```
iptables -A FORWARD -p tcp --dport 22 -m hashlimit --hashlimit 1/min \
--hashlimit-mode srcip --hashlimit-name ssh -m state --state NEW -j ACCEPT
```

Blokowanie przechodzących przez router pakietów z adresami prywatnymi:

```
ipset --create priv nethash
ipset --add priv 10.0.0.0/8
ipset --add priv 172.16.0.0/12
ipset --add priv 192.168.0.0/16
ipset --add priv 169.254.0.0/16
iptables -A FORWARD -m set --set priv src -j DROP
```

🕮 Literatura dodatkowa:

https://www.netfilter.org
https://wiki.archlinux.org/title/Iptables

🗒 Problemy do analizy:

- Czy iptables oferuje jakąkolwiek formę szyfrowania komunikacji?