**Network Exercise, Part 2**

Start en node/Express server på OS og efterlad det kørende under alle opgaverne. Åbn en terminal og start Wireshark.

OPGAVE 1

**Nmap and aggressive - what does that mean?**

Brug nmap til at scanne din IP-adresse lokalt på OS.

-v = verbose – meget information

-p 0-5552 = port

1. Start Wireshark og start capture.
2. Åbn en terminal og skriv: nmap -v -p 0-5552 -A IP\_TO\_YOUR\_OWN\_OS

(Add -Pn if requested)

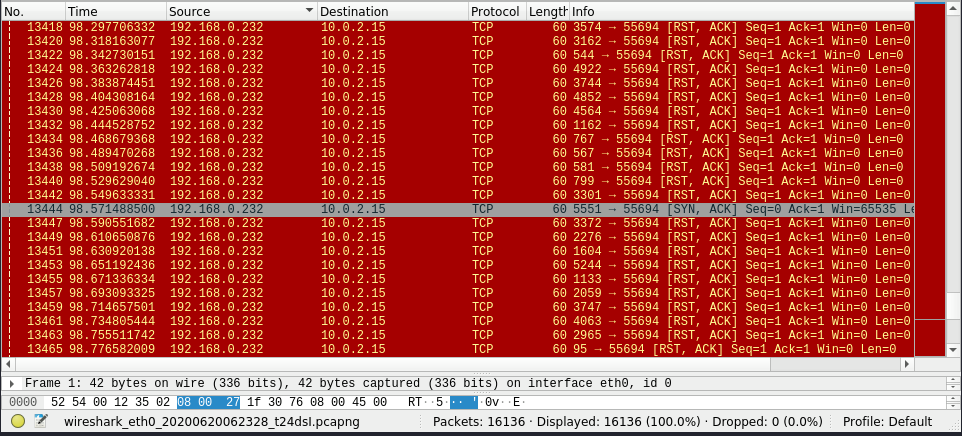
Dette kan tage lidt tid, da den skal scanne 5552 porte. Har man mere tid, kan man ændre til at scanne 65525 porte.

1. **How many packets were sent/received by this scan (see the bottom of the Wireshark Window)?**

16136 pakker er sendt/modtaget.

1. **Do you think something like this, will pass on unnoticed by a server that knows about security?**

Forsøger man at scanne en anden server, med kendskab til sikkerhed, vil det opdages straks, da mængden af data (antallet af pakker) er så stor.



Her ses response på de request sendt i forbindelse med scanning. De røde linjer indikerer et response på de lukkede porte, mens den grå port indikerer et response fra en åben port.

OPGAVE 2

**TCP SYN Flood Attack & Detection**

I denne opgave får vi kendskab til følgende:

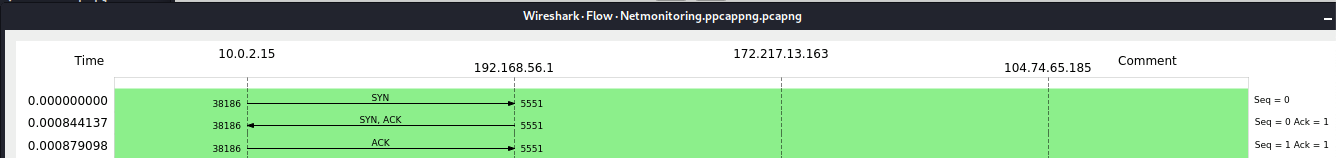
* SYN FLOOT attack mod TCP’s Three-way handshake.
* At man ikke kan stole på kilde IP-adressen, for indgående netværks forbindelser.
* *Hping3*, som er et netværks penetration Testing værktøj.

**Slow TCP SYN Attack, without a spoofed IP**

1. Start en Wireshark capture.
2. Åbn terminal og skriv: sudo hping3 -S -V  IP\_TO\_YOUR\_SERVER -p 5551
3. Efter få sekunder stop hping3 (CTRL+C) og stop capture.
4. Display filter: **tcp**

**Sammenlign output fra Wireshark, med det gemte Three-way handshake fra part-1 og forklar resultatet**

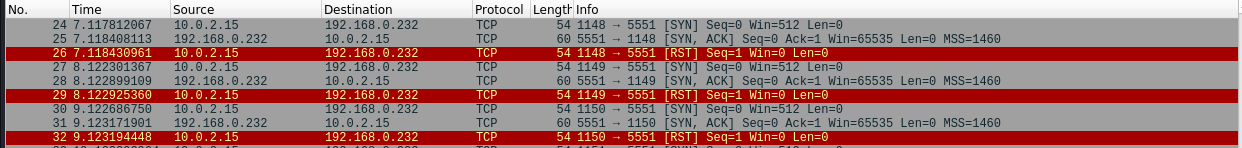
**Three-way handshake fra part 1**



**Three-way handshake fra part 2**

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse



I et Three-way handshake i part 1, skabes der en forbindelse mellem Kali Linux og OS, da der sendes en ACK fra Kali til serveren. I part 2 scannes den åbne port, for at se om der kan skabes forbindelse til serveren gennem porten. Dette bekræftes ved at der sendes en SYN ACK tilbage fra serveren til Kali. I stedet for at Kali sender en ACK retur, lukker den i stedet forbindelsen ved at sende et RST i stedet. Dette skyldes at vi ikke ønsker at oprette forbindelsen, men blot for at teste om det er muligt.

**Slow TCP SYN Attack, with random spoofed IP’s**

1. Start en Wireshark capture
2. Åbn terminal og skriv: hping3 -S -V  IP\_TO\_YOUR\_SERVER -p 5551 --rand-source
3. Efter få sekunder stop hping3 (CTRL+C) og stop capture.
4. Display filter: **tcp**

**Forklar resultatet, specielt kilde IP-adressen der bruges.**

På nedenstående billede, ses det at kilde IP-adressen ændrer sig ved hver SYN der sendes til serveren. Efterfølgende sender serveren en SYN ACK til den pågældende spoofed IP-adresse der sender SYN og derfor registrere Kali ikke det SYN ACK der sendes som response.

Et billede, der indeholder computer

Automatisk genereret beskrivelse

**TCP SYN FLOOD Attack, with random spoofed IP’s**

1. Start en Wireshark capture
2. Åbn terminal og skriv: hping3 -S --flood  IP\_TO\_YOUR\_SERVER -p 5551 --rand-source
3. Efter få sekunder stop hping3 (CTRL+C) og stop capture.
4. Display filter: **tcp && ip.src==IP\_TO\_KALI**

**Se hvor mange TCP-pakker (”displayed”) der var sendt fra din klient. Forklar resultatet af dine observationer.**

Serveren (host) og port er utilgængelig. ICMP står for at sende error beskeder og information om hvorvidt kommunikation med en anden IP-adresse var en succes eller ikke. SYN FLOOD-angrebet har været en succes, da serveren ikke længere er tilgængelig.

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

OPGAVE 3

**Performing a UDP Flood DOS attack**

UDP is a protocol which does not create a session between two devices. In other words, no handshake process required.  
  
A UDP flood does not exploit any vulnerability. The aim of UDP floods is simply creating and sending a large amount of UDP datagrams from spoofed IP’s to the target server. When a server receives this type of traffic, it is unable to process every request and it consumes its bandwidth by sending ICMP “destination unreachable” packets.

In this demo, we will set up a simple UDP-time server, a legal user client and a hacker as sketched below.

**Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse**

Getting Started

Udskift koden den simple node/Express server, med [this](https://github.com/SecurityDatFall2018/Week15NetworksAndSecurity/blob/master/app.js) server. Den virker som før, men nu inkludere den en simple UDP-time server, som vi vil forsøge at angribe.

1. Start serveren fra din OS.
2. For at teste UDP-serveren, skal vi bruge en klient (Kali).
3. På Kali skrives i terminalen: netcat -u SERVER\_IP 5555

Tryk ENTER et par gange, for at se resultat. Ved at trykke ENTER sendes UDP-pakker til serveren, som vil sende response tilbage med en pakke der indeholder en server-time. Lad den søde klient køre, for at være sikker på at serveren er oppe eller nede/langsom i forhold til DoS-angreb.

**Performing the UDP Flood DOS attack**

Serveren der skal hackes og den søde klient er klar. Klargør den onde hacker:

1. Åbn en ny terminal, og skriv følgende: hping3 --flood --udp -p 5555 SERVER\_IP
2. Start en Wireshark capture og kør for 5-10 sekunder (NOT MORE)
3. Mens hping3 stadig kører, tryk ENTER et antal gange i den søde klient. Får du et response?

Et billede, der indeholder sidder, sort, bord, skærm

Automatisk genereret beskrivelse

Nej. Serveren er under UDP flood fra hackeren,

så den søde klient kan ikke komme igennem.

1. Stop hping3

**Investigate the sample**

1. **In Wireshark, you should see a “billion” UPD packets, all with the UDP-server as the destination. How many packets were sent in total (see the status bar at the bottom)?**

hping3 siger der er 1.785.214 pakker.

Et billede, der indeholder sort, foto, orange, mørk

Automatisk genereret beskrivelse

Wireshark har opfanget 2466 pakker.

1. **We would expect, for each UPD-packet sent, to get a packet back with the server-time. Enter this value in the display-filter box:** **udp && ip.dst==IP\_FOR\_KALI - How many packets did you receive (compared to what was sent)?**

Kali har modtaget/registeret 1 UDP-pakke, med en server-time.

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

1. **Change the filter to this:** **ICMP && ip.dst==IP\_FOR\_KALI** - **Did you receive any ICMP packets (see the intro to this demo)?**

Kali har modtaget/registeret 1 ICMP-pakke. Denne pakke er sendt, da serveren ikke kan håndtere det store antal request der sendes.

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

1. **Have you in effect demonstrated a DoS-attack?**

Eftersom at den søde klient ikke modtager en server-time, samt der er registeret en ICMP med ”destination unreachable” kan vi konkludere at DoS-attack har været en succes.

In this demo, we did not spoof the sender when we flooded the server since we would like to see the returned (if any) UDP packages.

For at lave et rigtigt angreb, kan man spoof sin IP, ved at skrive: hping3 --flood --rand-source --udp -p 5555 SERVER\_IP

Et billede, der indeholder tekst

Automatisk genereret beskrivelse

OPGAVE 4

**DNS Attack Example**

Dette eksempel bruger en falsk ”man-in-the-middle” server. For at ”forgite” offerets lokale DNS-cache, gøres det manuelt, ved at gøre følgende, i Kali terminal.

1. Åbn hosts-file i terminalen: **sudo nano /etc/hosts**
2. Tilføj denne til bunden af filen: **68.183.71.148 danskebank.dk**
3. Åbn et inkognito vindue i browseren og skriv: <http://danskebank.dk>
4. Bekræfter dette (proof-of-concept) demonstrationen af man-in-the-middle angreb? Hvis ja, forklar hvorfor og tegn arkitekturen.

For at fungere som man-in-the-middle, skal en hacker lave om i klientens DNS-cache, for på den måde at omdirigere et offer til en falsk hjemmeside, som er en kopi af den side, som vedkommende forsøger at komme ind på. I dette her tilfælde Danske Bank. Forsøger vedkommende at logge ind på sin Danske Bank konto, igennem den falske hjemmeside, kan hackeren få fat på klientens login oplysninger. Vedkommende bliver videreført til den rigtige hjemmeside, og opfatter ikke at en ”man-in-the-middle”, nu har loginoplysningerne.

Et billede, der indeholder skærmbillede, monitor

Automatisk genereret beskrivelse

*Please note that it does not necessarily have to be the local DNS cache that gets poisoned. Any poisoned DNS-server, in between you and the server that provides the IP, could do the same.*

OPGAVE 5

**ARP Spoofing and more “advanced” attacks**

Before you start, make sure you have watched this [video](https://www.youtube.com/watch?v=RTXAUJ2yqCg&t=83s) (7 min.) since it provides the background and some theory for this exercise. The exercise below, however, goes a bit further.

*What we will simulate is the situation sketched below. A hacker has in some way gained access and is connected to your internal network (evil below). You (happy below)  would like to access a server (dat-security.dk) hosting some very very important secrets ;-) so authentication is required.*

*Obviously, access to this remote server requires access to the “big internet” which must happen via your default gateway.*

*What you hope/think is, that your packets will follow the green path sketched below. What you don’t know, is that a hacker has placed himself as a man-in-the-middle, and as so has access to all packages sent from your computer (red path).*

Et billede, der indeholder computer, enhed

Automatisk genereret beskrivelse

**Local network Denial of Service Attack**

1. Sæt Netværk til *Bridged-Adapter*.
2. Find IP-adressen på din computer 192.168.0.230
3. Find gateway for Kali (skulle gerne være den samme som på din computer) (Kali → ip route)
4. Skriv i terminalen på Kali: **arp** (Kun IP for gateway er synlig).
5. I terminalen skriv: **ping + IP\_FOR\_YOUR\_LAPTOP**
6. Stop ping og skriv igen: **arp** (IP for gateway og IP for laptop er synlig).
7. Brug built-in-tool ”arp spoofing”, skriv følgende - benyt root-bruger (sudo -s):

**arpspoof -i eth0 -t IP\_FOR\_YOUR\_LAPTOP -r IP\_FOR\_YOUR\_GATEWAY**

1. På din computer, prøv at tilgå en hvilken som helst hjemmeside. Dette er ikke muligt, eftersom at **arpspoof** har fået din computer til at tro at Kali er Gateway og eftersom at Kali ikke bruger/gør noget med disse pakker, er de ”bare tabt”.

Indtil videre har vi lavet et lokalt DoS-angreb, som fungere fint til at se arp-spoofing i aktion, men fik ikke det resultat vi ønskede. Vi ønsker at Kali er ”man-in-the-middle”, så næste step er at videresende disse pakker til den rigtige Gateway.

**ARP-spoofing and man in the middle**

* 1. Sluk arpspoof og tænd for ip-forward I Kali, ved at skrive følgende:

**echo 1 > /proc/sys/net/ipv4/ip\_forward** (repeat with 0, when you want to turn it off)

* 1. Start arpspoof igen:

**arpspoof -i eth0 -t IP\_FOR\_YOUR\_LAPTOP -r IP\_FOR\_YOUR\_GATEWAY**

* 1. Bekræft at du igen kan få adgang til internettet fra din computer, eftersom at evil-Kali videregiver pakkerne.

**Sniffing the traffic (non-http)**

1. Åbn Wireshark og start capture.
2. Fra din computer, åbn browser og skriv: <http://dat-security.dk/xss/AddUser>
3. Login med: brugernavn: **sveske**, adgangskode: **test**
4. Stop capture.
5. Display filter: **http**
6. Find POST request der er sendt til serveren.
7. Kan du aflæse brugernavn og adgangskode, der er sendt med dette request?
8. Kan du finde og læse Token der er returneret?

Et billede, der indeholder skærmbillede

Automatisk genereret beskrivelse

Hvis begge svar er JA, burde du være overbevist om reglen: **ALWAYS HTTPS**

**Sniffing the traffic (https)**

Gentag alt fra forrige opgave, og brug URL [https ://dat-security.dk](http://dat-security.dk)

1. Kan du aflæse brugernavn og adgangskode, der er sendt med dette request?
2. Kan du finde og læse Token der er returneret?

Hvis begge svar er NEJ, burde du være overbevist om reglen: **ALWAYS HTTPS**

**Problems with the attacks**

This will target the gateway, so you may not do this at the school premises or on any public networkEt billede, der indeholder tegning, skilt, mad

Automatisk genereret beskrivelse

The attack done above was actually only a **half** man-in-the-middle, since we have not fooled the gateway to think that we are the happy-client, but only fooled the happy-client to think that we (Kali) is the gateway.

In one terminal do:

**arpspoof -i eth0 -t IP\_FOR\_YOUR\_LAPTOP -r IP\_FOR\_YOUR\_GATEWAY**

In another terminal do:

**arpspoof -i eth0 -t IP\_FOR\_YOUR\_GATEWAY IP\_FOR\_YOUR\_LAPTOP**

*Note: I could not get this to work on either of my two gateways out of the house, for reasons which I cannot yet explain*