Securitatea Informatiei Tema 3 - Atacul Spoofed SYN Flooding

Smau Adrian-Constantin

January 2022

1 Ce este atacul Spoofed SYN Flooding si cum functioneaza?

Atacul SYN Flood (atac 'half-open') este un tip un atac de tip 'Denial-of-service' care consta in initierea rapida de catre un atacator a unor cereri de conexiune catre un server (pachete SYN) fara a le finaliza. Astfel, victima este suprasolicitata si resursele sale sunt consumate, aceasta nefiinf capabila sa mai raspunda in timp util la cereri legitime de conexiune. In circumstante normale, procedura de conexiune TCP consta in 3 pasi pentru a crea o conexiune:

1. Clientul trimite un pachet SYN catre serverul cu care se doreste initierea conexiunii
2. Serverul raspunde pachetului initial cu un pachet de tip SYN/ACK, care confirma initierea comunicararii
3. In final, clientul returneaza pachetul ACK (de acknowledgement) pentru a confirma, la randul sau, primirea mesajului de la server. In acest moment, conexiunea TCP este deschisa si este posibil schimbul de date

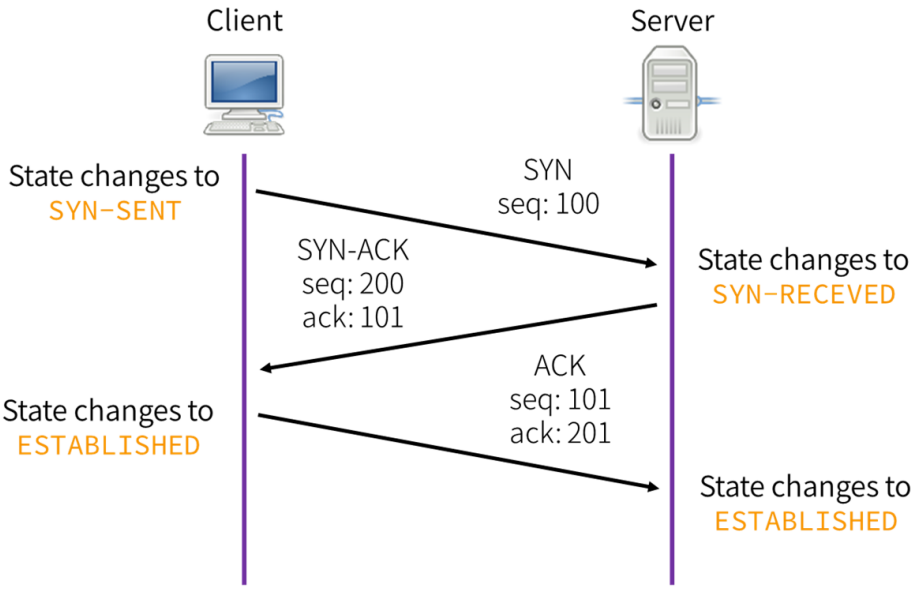

Figure 1: Modelul 3-way handshake, pe care atacul SYN Flooding il exploateaza

Pentru a bloca serviciul (a crea un 'denial-of-service'), un atacator exploateaza faptul ca, dupa ce pachetul SYN initial a fost primit de catre server (in urma pasului 1 de mai sus), serverul va raspunde cu unul sau mai multe pachete de tip SYN/ACK si va astepta pasul final pentru a deschide conexiunea TCP. Exploatarea se va face in felul urmator:

1. Atacatorul trimite un numar mare de pachete SYN catre victima(un server), cu o adresa IP 'spoofed', falsificand originea pachetelor pentru a-si masca adevarata identitate

2. Serverul raspunde la fiecare dintre cererile de conexiune si lasa un port deschis pentru a primi raspunsul din partea clientului (pachetul final ACK)

3. Cat timp serverul asteapta pachetul ACK final, atacatorul continua in a solicita serverul cu mai multe pachete SYN. Astfel, resursele serverului sunt epuizate din cauza faptului ca acesta pastreaza un port deschis pentru o durata nespecificata de timp. Odata ce toate porturile serverului au fost utilizare, acesta nu va mai putea raspunde corect la cererile legitime de conectare

Din cauza faptului ca serverul lasa o conexiune deschisa, insa clientul nu interactioneaza cu aceasta, conexiunea se numeste 'half-open'. Natura atacului face ca serverul sa lase multiple conexiuni deschise si sa astepte ca acestea sa devina 'timed-out' inainte ca porturile respective sa devina din nou valabile. De aceea, SYN Flooding este un atac de tip 'half-open'.

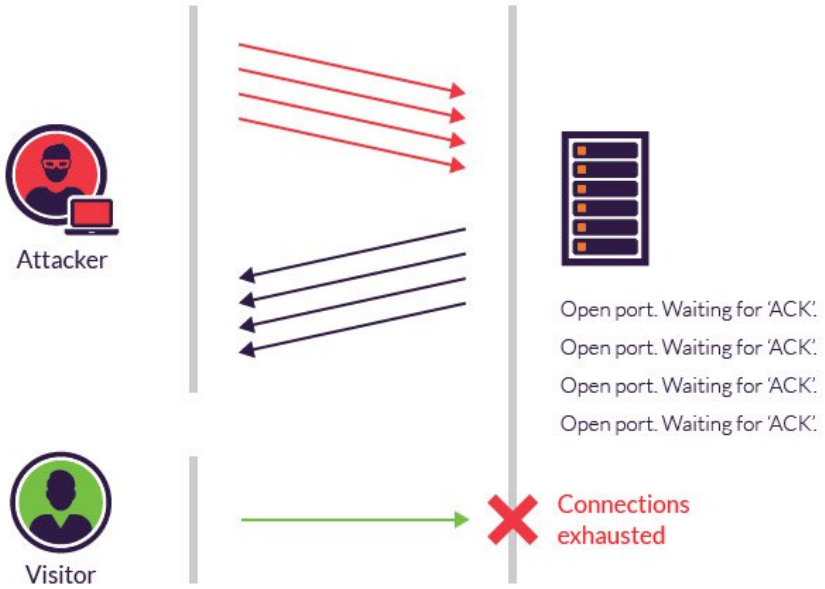

Figure 2: Modelul SYN Flooding, care suprasolicita serverul prin trimiterea unui numar mare de cereri de initiere ale conexiunii fara a le finaliza

2 Atacul Spoofed SYN Flood

2.1 Configurarea atacului

2.1.1 Tool-uri folosite

In urma configurarii setup-ului urmand pasii de la "https://profs.info.uaic.ro/ liliana.cojocaru/Lab11.pdf", am folosit masina virtuala C1 drept atacator, C2 drept observator, iar pe MV Router am configurat un server FTP cu ajutorul tool-ului 'vsftpd', apoi activand firewall-ul cu ajutorul comenzii "sudo ufw enable".

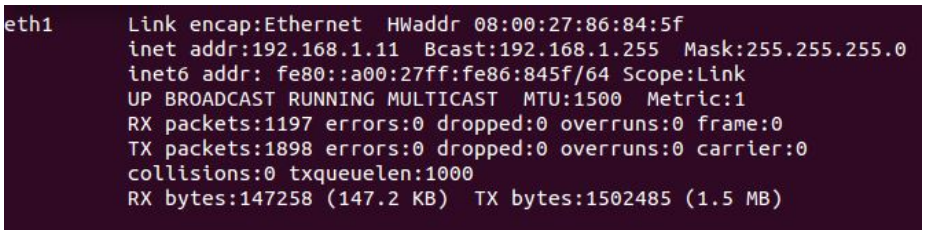

Figure 3: Rularea comenzii ifconfig pe Router, care are adresa statica 192.168.1.11

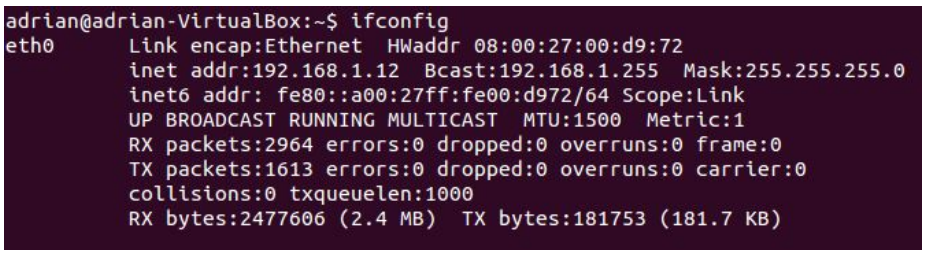

Figure 4: Rularea comenzii ifconfig pe atacatorul C1, care are adresa statica 192.168.1.12

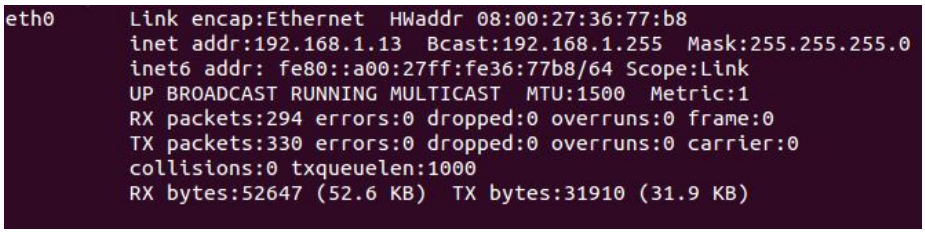

Figure 5: Rularea comenzii ifconfig pe observatorul C2, care are adresa statica 192.168.1.13

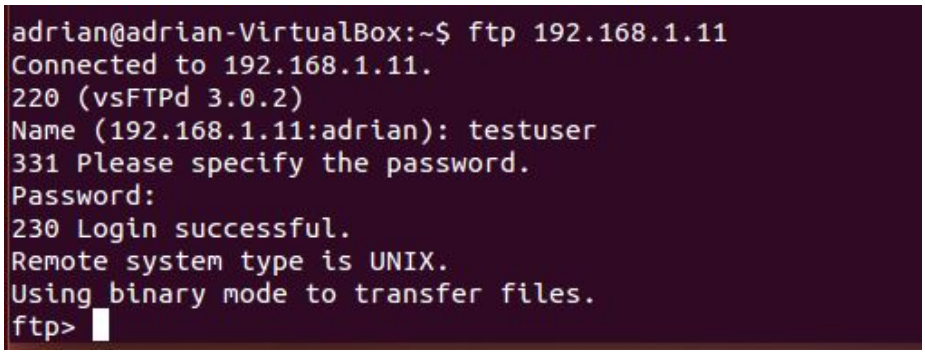

Figure 6: Conectarea observatorului C2 la serverul FTP de pe Router inainte de atac

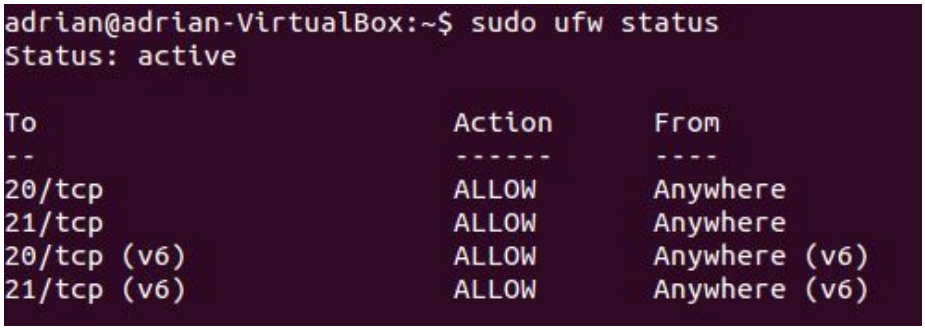

Figure 7: Activarea Firewall-ului pe serverul FTP de pe Router

Pentru monitorizarea traficului si detectarea atacului, am folosit tool-ul Wireshark, iar pentru trimiterea pachetelor SYN catre server, mascarea originii si setarea portului 21 (specific serverelor FTP) vom folosi tool-ul Netwox 76 - SynFlood ("https://web.ecs.syr.edu/ wedu/Teaching/cis758/netw522/netwox-doc_html/tools/76.html") - care, insa, se va dovedi ineficient. Asadar, vom opta pentru tool-ul hping3, care ne va ajuta sa setam numarul pachetelor trimise, dimensiunea acestore, tipul acestora (SYN), de a realiza spoofing-ul cu succes randomizand originea pachetelor si de a seta dimensiunea ferestrei TCP. Capturi de ecran cu utilizarea acestora vor putea fi vizualizate in sectiunile ce urmeaza.

2.1.2 Pasi pentru a spori eficienta atacului

Pentru a evita denial-of-service, sistemele de operare au implementat un parametru numit "backlog" care seteaza un numar maxim de conexiuni simultare care pot fi in starea "SYN-RECEIVED". Observam dimensiunea cozii de pe masina Router prin executarea comenzii "sysctl -q net.ipv4.tcp_max_syn_backlog". Putem mod-

ifica acest parametru din fisierul "/etc/sysctl.conf", sau prin comanda "sudo sysctl -w net.ipv4.tcp_max_syn_backlog=32", pe care am rulat-o pentru a micsora dimensiunea "backlog-ului" de la 128 la 32.

```
adrian@adrian-VirtualBox:~$ sysctl -q net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 128
```

Figure 8: Dimensiunea default a backlog-ului

```
adrian@adrian-VirtualBox:~$ sudo sysctl -w net.ipv4.tcp_max_syn_backlog=32
net.ipv4.tcp_max_syn_backlog = 32
adrian@adrian-VirtualBox:~$ sysctl -q net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 32
```

Figure 9: Dimensiunea backlog-ului dupa micsorarea acesteia

Similar, sistemele de operare implementeaza un parametru numit tcp_syncookies, care are scopul de a renunta la conexiuni odata ce backlog-ul este plin. Astfel, serverul raspunde la fiecare cerere de conexiune cu un pachet SYN-ACK, dar apoi distruge cererea SYN din backlog, stergand request-ul din memorie si lasand port-ul deschis pentru a face o noua conexiune. Daca aceasta conexiune este legitima, un pachet final ACK este trimis de la client catre server si serverul va putea reconstrui entry-ul din backlog. Desi aceasta mitigare poate pierde unele informatii despre conexiunea TCP, este o optiune preferabila in defavoarea unui potential atac denial-of-service. Putem vedea daca aceste cookie-uri sunt enabled prin comanda "sysctl -q net.ipv4.tcp_syncookies", si pe putem da disable fie prin manipularea fisierului "/etc/sysctl.conf", fie prin comanda "sudo sysctl -w net.ipv4.tcp_syncookies=0".

```
adrian@adrian-VirtualBox:~$ sysctl -q net.ipv4.tcp_syncookies
net.ipv4.tcp_syncookies = 1
```

Figure 10: Setarea default a tcpcookies din Router

```
adrian@adrian-VirtualBox:~$ sudo sysctl -w net.ipv4.tcp_syncookies=0
net.ipv4.tcp_syncookies = 0
adrian@adrian-VirtualBox:~$ sysctl -q net.ipv4.tcp_syncookies
net.ipv4.tcp_syncookies = 0
```

Figure 11: Dezactivarea tcpcookies din Router

2.1.3 Observatii prelimiare

Prin rularea comenzii "netstat –listening –all –tcp", putem verifica statusul cozii, filtrand ascultarea doar a conexiunilor de tip TCP.

Figure 12: Rularea comenzii netstat inaintea atacului

2.2 [INEFICIENT] Rularea atacului - Netwox 76

Pentru a rula atacul, am incercat sa folosesc tool-ul Netwox 76, setand portul 21 (pentru serverul FTP), trimitand pachetele la adresa 192.168.1.11 (adresa Router-ului) - spoofing-ul fiind activat by default, astfel mascand originea pachetelor (sudo netwox -i 192.168.1.11 -p 21). Insa, tool-ul nu reuseste sa suprasolicite server-ul. In Wireshark, vom filtra pachetele syn fara cele de acknowledgement cu ajutorul filtrului "tcp.flags.syn == 1 and tcp.flags.ack == 0" din Wireshark.

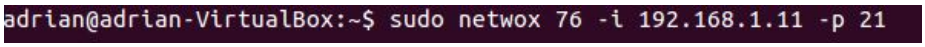

Figure 13: Rularea comenzii netwox pe atacatorul C2

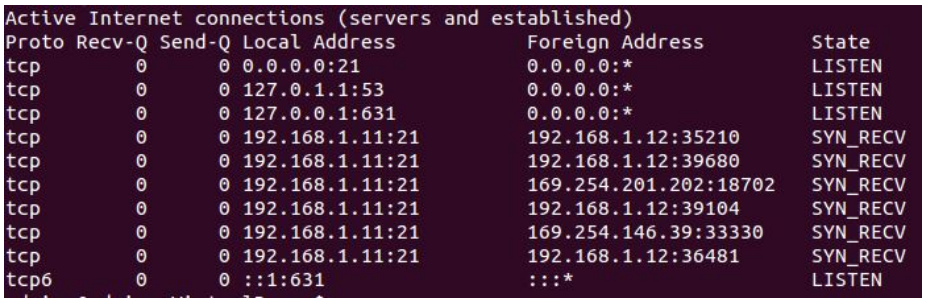

Figure 14: Rularea comenzii netstat in timpul atacului netwox

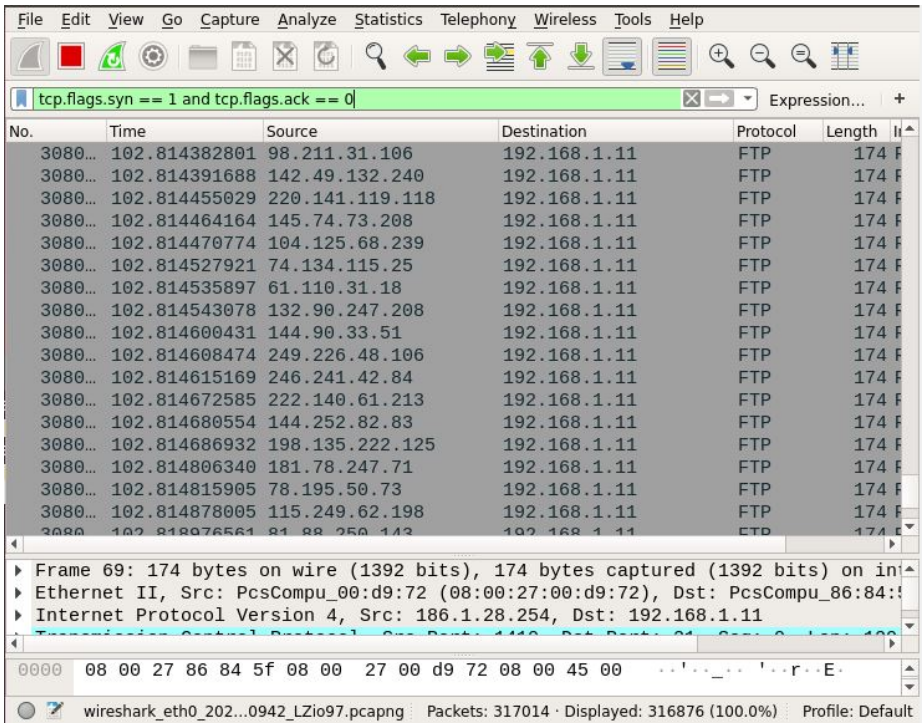

Figure 15: Interceptarea traficului cu ajutorul Wireshark atunci cand rulam o comanda cu spoofing

2.3 [EFICIENT] Rularea atacului - Hping3 fara spoofing

Asadar, vom folosi tool-ul hping3, mai intai fara spoofing. Cu ajutorul acestui tool fara a integra spoofing-ul, reusim sa epuizam resursele server-ului FTP, astfel incat observatorul C2 nu reuseste sa se conecteze la server, conexiunea sa fiind timed-out, deci atacul a reusit.

```
adrian@adrian-VirtualBox:~$ sudo hping3 -p 21 -S --flood 192.168.1.11
HPING 192.168.1.11 (eth0 192.168.1.11): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

Figure 16: Rularea comenzii hping fara spoofing pe atacatorul C2

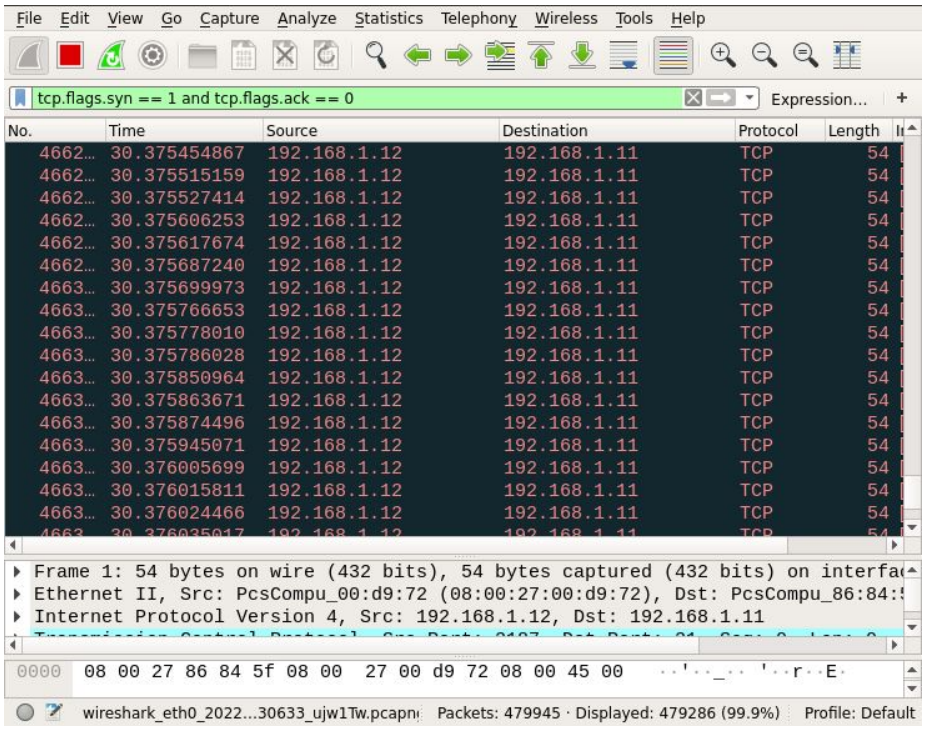

Figure 17: Interceptarea traficului cu ajutorul Wireshark atunci cand rulam o comanda fara spoofing

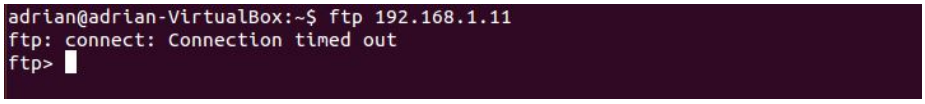

Figure 18: Observatorul C2 nu reuseste sa se conecteze la serverul FTP, deci atacul a reusit

2.4 [INEFICIENT] Rularea atacului - Hping3 cu spoofing

Atunci cand rulam tool-ul hping3 cu spoofing (integram parametrul –rand-source in comanda), observam ca nici acum nu se reuseste suprasolicitarea serverului FTP din Router, comanda netstat aratandu-ne faptul ca resursele sunt inva valabile si putand sa ne conectam de pe observator fara probleme. Fiind o metoda cu spoofing, Wireshark va arata similar cu printscreen-ul din Figura 15.

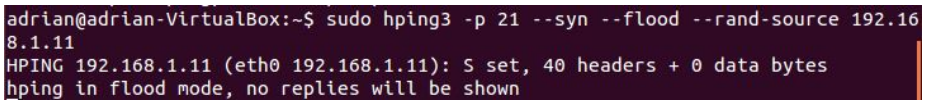

Figure 19: Rularea comenzii hping cu spoofing pe atacatorul C2

Figure 20: Rularea comenzii netstat de pe Router, fapt ce indica ineficienta atacului cu spoofing

3 Bibliografie

1. https://profs.info.uaic.ro/ liliana.cojocaru/Lab11.pdf
2. https://web.ecs.syr.edu/ wedu/Teaching/cis758/netw522/netwox-doc_html/tools/76.html
3. https://phoenixnap.com/kb/install-ftp-server-on-ubuntu-vsftpd
4. https://en.wikipedia.org/wiki/SYN_flood
5. https://youtu.be/p70YziFk7WU
6. https://youtu.be/lFpDnPGXNwk
7. https://www.firewall.cx/general-topics-reviews/network-protocol-analyzers/1224-performing-tcp-syn-flood-attack-and-detecting-it-with-wireshark.html