LAB10B: Scapy vs Hping3

[Simon X. Camilo. Cybersecurity Student]

[BHCC CIT-275-WB Professor Philip Kazanjian]

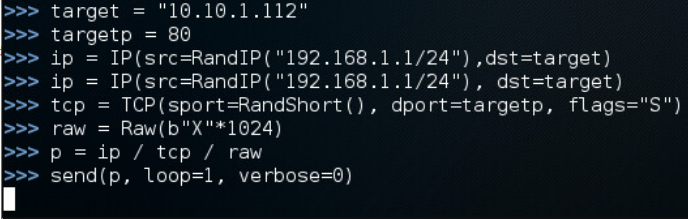
Introduction

This document shows an attempt at creating a DoS attack that will bypass the firewall in the network using Scapy. Scapy is a powerful packet manipulation program that performs better compared to many other tools, like hping3. This tool can be used to perform a DoS attack on a server, and also slip by the firewall rules set in the server.

(Scapy.net, 9/5/2018)

# Analysis

There are many ways to create a DoS script for Scapy. The script I used is show in image 1

  
**Image 1**

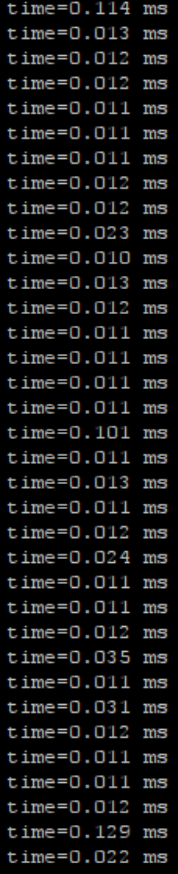
*Scapy DoS script*

This script sends many syn ack 1kb packets filled with “X’s” to the target’s (“10.10.1.112”) port 80, while at the same time making the packet show a different source ip address for each packet, making it harder for the attack to detect and also to be stopped by the firewall anti-DoS systems. Using scapy is very simple, and the script is also very simple to do, the script is going to be deconstructed below to show how an attacker might try to attack the systems

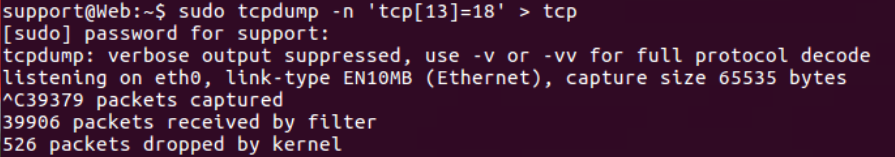
* target - The ip address of the target for DoS
* Targetp - The port that the packets are going to use to enter in the target’s machine
* ip = IP(src=RandIP(“192.168.1.1/24”), dst=target) – creates ip packet
  + RandIP - this packet will say where the packet comes from, which will be a random ip address that will hide the identity of the attacker and make these packets harder to ignore.
  + Dst=target - This packet also has the ip of the target
* tcp = TCP(sport=RandShort(), dport=targetp, flags=”SA”) – Creates tcp syn packet
  + sport =RandShort() – sets the source port for the tcp packet, RandShort will make it choose random ports so the attacker is harder to detect and stop
  + dport=targetp – sets the port of the target
  + flags=”S” – It sets the flag of the tcp packet, which in this case is S
* raw = Raw(b”X”\*(1024) – Data of the packet
* p = ip / tcp/ raw – puts all the layers in one variable. This creates the payload
* send(p, loop=1, verbose=0) – Executes the attack, loop will make the packet not stop sending until the attacker makes it stop.

(Rockiz, 9/8/2020) (Pierre, 8/4/2017

As shown in image 2, the site is receiving spikes, with some packets not being sent as quickly as they used to before the attack. This won’t make the site go down since more machines are needed to make the site crash.

  
**Image 2**

*urbank.com(10.10.1.112) is experiencing slow downs*

  
**Image 3**

*urbank.com(10.10.1.112) is experiencing slow downs*

Image3 shows that most of the packets were received and haven’t been dropped. This shows that the script was successful and it has the potential to do real damage to the website.

# Conclusion

Sending packets with Scapy was a lot more advanced than Hping3. Scapy gives attackers a lot more control over the packets, and it’s packets were not dropped when the firewall was on. Scapy requires some knowledge and tedious code building unlike Hping3, which is simple to do and requires a few commands. If Scapy had the option to be a lot more interactive for people who have little to no knowledge about sending packets, then it would be much easier to understand and use.

References

Philippe Biondi and the Scapy community. (2018, May 9). Scapy. Retrieved March 03, 2021, from https://scapy.net/

Philippe Biondi and the Scapy community. (2016, August 3). Usage¶. Retrieved March 03, 2021, from https://scapy.readthedocs.io/en/latest/usage.html

Pierre. (2017, August 04). Scapy - persistent randip. Retrieved March 03, 2021, from https://stackoverflow.com/questions/45509493/scapy-persistent-randip

Rockikz, A. (2020, August 09). How to make a syn flooding attack in python. Retrieved March 03, 2021, from <https://www.thepythoncode.com/article/syn-flooding-attack-using-scapy-in-python>