**SNORT**

Setting it up:

1. kali doesn’t allow us to automatically download this program.
2. First, you have to edit the sources.list file. This is located at */etc/apt/sources.list*
3. to the end of the file, add the line ‘deb http://deb.debian.org/debian buster main’. You will most likely need sudo to do this.
4. Next, you will need to update apt with ‘sudo apt update -y’
5. And finally, install the program with ‘sudo apt update -y’
6. This may take sometime to fully download, so be patient.
7. Next in ‘*/etc/snort*’ create a directory calls ‘logs’ and a file called ‘fullstack.rules’. This will give the program a location to place its log files (logs) and to pull the rules from (fullstack.rules)
8. Next we need to configure it.
9. Go to /etc/snort and use a text editor on snort.conf
10. Three areas need to be changed.
11. First, ‘var HOME\_NET’ is default set to any but it needs to be set to the host IP.
12. next, ‘var EXTERNAL\_NET’ is default set to any but should be set to !HOME\_NET because we want to be any except the home IP.
13. Finally we need to check that ‘var RULE\_PATH /etc/snort/rules’ is the correct path.
14. If all are correct, it can be saved and exited.

Writing rules:

1. There are multiple parts of a rule that you should know.
2. First part is the action. These include alert, log, pass, activate, or dynamic. Alert will log the packet and alerts the security admin. (This is the most common)
3. Next is the protocol. You can pick which ever protocol you need to create the rule to alert you about a certain action.
4. Next is the source IP and Port. Unless we know the exact IP and Port we may be attacked from, we simply type ‘any’ for both.
5. Next is the flow or direction. while this usually goes from source to destination, it can also reverse and go both ways.
6. Next is the destination IP and Port. The IP will most likely be the IP of the machine you are on unless you want any machine, then you put ‘any’. for port unless you know the specific port, put ‘any’.
7. the final section is options. While there are many options that can be researched, the ones that will be discussed here are message and SID.
   1. message= ‘msg:’ is a message that explains what is going on in plain text. helps identify what a specific rule is designed to do.
   2. SID= ‘sid:’ is a random ID number specifically only for this rule. This number should be larger than 1,000,000.
8. A final rule will look like ‘***alert ICMP any any -> 192.168.56.100 any (msg:'YOU ARE BEING PING'D!!!';sid:1000001;)***’

Running SNORT:

1. For this example we will capture ping’s between one VM and a kali vm using wireshark.
2. Once the ping’s are captured, they should be saved into ‘*/etc/snort’* and saved with the .pcap extension.
3. Next execute the command ‘**snort -k none -l ./logs/ -c ./fullstack.rules -r ./captured.pcap’**
4. running that will start up snort and allow it to process the captured packets.
5. Because we were pinging and looking at protocol ICMP, it should give us alerts.

Scan Behavior:

1. There are multiple ways to run nmap to find out info about the target. normal, TCP, XMAS, FIN, NULL, UDP)
2. Here are example rules for each :
   1. (Normal nmap scan) alert icmp any any -> 192.168.56.110 any (msg: "NMAP ping sweep Scan"; dsize:0;sid:10000002; rev: 1;)
   2. (TCP nmap scan) alert tcp any any -> 192.168.56.110 any (msg: "NMAP TCP Scan";sid:10000003; rev:2; )
   3. (XMAS nmap scan) alert tcp any any -> 192.168.56.110 any (msg:"Nmap XMAS Tree Scan"; flags:FPU; sid:1000004; rev:1; )
   4. (FIN nmap scan) alert tcp any any -> 192.168.56.110 any (msg:"Nmap FIN Scan"; flags:F; sid:1000005; rev:1;)
   5. (NULL nmap scan) alert tcp any any -> 192.168.56.110 any (msg:"Nmap NULL Scan"; flags:0; sid:1000006; rev:1; )
   6. (UDP nmap scan) alert udp any any -> 192.168.56.110 any ( msg:"Nmap UDP Scan"; sid:1000010; rev:1; )
3. After running the nmaps for each of there types and capturing the files with wireshark, the files can then be read with snort and the alerts can be viewed in ‘/logs/alert’.