LAB02 How to respond against scanning

Class student ID Name

Simulation Environment

|  |  |  |
| --- | --- | --- |
|  | Attacker | Target |
| OS | Ubuntu  Window |  |
| IP | Test-bed IP |  |
| Responding type | Check the port in use  Block specified port  Check the process in running | |
| Responding program | Linux command, Window command | |
| Process | [Module 1] [Module 2] [Module 3] | |

**[Module 1] Exercise 10 processes command and explain each step [essentional]**

## Check the TCP port currently in use

## Check process currently using the port

## Confirm portnumber

## Check the port in Mac

## Block specified port

## Kill 8080 port

## Free up the Port

## Find what service or process is running on port 8080 on Windows

## Kill the process on Windows

[Module 2] Exercise 8 Linux commands and test as far as you can. Describe each test result including error status [essentional]

### Create ipset liststest

* 1. $ ipset create port\_scanners hash:ip family inet hashsize 32768 maxelem 65536 timeout 600
  2. $ ipset create scanned\_ports hash: ip,port family inet hashsize 32768 maxelem 65536 timeout 60

1. Create iptables rules
   1. $ iptables –A INPUT –m state --state INVALID –j DROP
   2. $ iptabels –A INPUT –m state --state NEW –m set ! --match-set scanned\_ports,dst –m hashlimit --hashlimit-above 1/hour --hashlimit-burst 5 --hashlimit-mode srcip --hashlimit-name portscan --hashlimit-htable-expire 10000 -j SET --add-set port\_scanners src –exist
   3. $ iptables -A INPUT -m state --state NEW -m set --match-set port\_scanners src -j DROP
   4. $ iptables -A INPUT -m state --state NEW -j SET --add-set scanned\_ports src,dst
2. Create a whitelisted list
3. Change drop rule

[Module 3] Port knocking commands exercise [optional]

1. Explain what is Port knocking, Port proof ?
2. Exercise following commands & explain each step [optional]

Step 1: Install knockd

$ sudo apt install knockd

$ sudo apt-get install iptables-persisitent

$ sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

$ sudo iptables -A INPUT -p tcp --dport 22 -j REJECT

$ sudo systemctl start netfilter-persistent

$ sudo netfilter-persistent save

$ sudo netfilter-persistend reload

Step 2: Configuring knockd

$ sudo gedit /etc/knockd.conf

Step 3: The knockd Control File Edits

$ ip addr

Step 4: Testing

$ sudo systemctl start knockd

$ tail –f /var/log/syslog

$sudo apt-get install knockd

$sudo apt-get install iptables-persistent

$sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

$sudo iptables -A INPUT -p tcp --dport 22 -j