

# Chinese PlugX Malware Hidden in Your USB Devices?

By Mike Harbison, Jen Miller-Osborn

Published: 2023-01-26 · Archived: 2026-04-05 16:11:50 UTC

## Executive Summary

Recently, our Unit 42 incident response team was engaged in a Black Basta breach response that uncovered several tools and malware samples on the victim's machines, including [GootLoader](#) malware, [Brute Ratel C4](#) red-teaming tool and an older [PlugX](#) malware sample. The PlugX malware stood out to us as this variant infects any attached removable USB media devices such as floppy, thumb or flash drives and any additional systems the USB is later plugged into.

This PlugX malware also hides actor files in a USB device using a novel technique that works even on the most recent Windows operating systems (OS) at the time of writing this post. This means the malicious files can only be viewed on a Unix-like (\*nix) OS or by mounting the USB device in a forensic tool.

We also discovered a similar variant of PlugX in VirusTotal that infects USB devices and copies all Adobe PDF and Microsoft Word files from the host. It places these copies in a hidden folder on the USB device that is created by the malware.

PlugX is a second-stage implant used not only by multiple groups with a Chinese nexus but also by several cybercrime groups. It has been around for over a decade and has been observed in some high-profile cyberattacks, including the [U.S. Government Office of Personnel Management \(OPM\)](#) breach in 2015. It is a modular malware framework, supporting an evolving set of capabilities throughout the years.

Palo Alto Networks customers receive protections against the types of threats discussed in this blog by products including [Cortex XDR](#) and [WildFire](#).

## Introduction

It's not uncommon for multiple malware samples to be discovered during an investigation, as occurred in this situation with GootLoader, Brute Ratel C4 and PlugX. Numerous threat actors compromise targets and can coexist simultaneously on the affected machine.

Because we can't conclusively say whether these malware samples were left by one group or several, we can't attribute these tools to the Black Basta ransomware group. However, the version of Brute Ratel C4 used in this case is the same one reported by [Trend Micro](#), which also involved the Black Basta ransomware group.

## PlugX Malware Infection

Historically, a PlugX infection begins by hijacking a known and trusted, digitally signed software application to load an actor-created encrypted payload. This technique has been used since 2010 and is listed in the MITRE

ATT&CK techniques as [Hijack execution flow DLL-Side loading](#) ID: T1574.002 Sub-technique T1574.

In this case, the threat actors decided to hijack a popular and free open source debugging tool for Windows called [x64dbg](#), which is used by the malware analysis/reverse engineering community. X64dbg applications are digitally signed by "Open Source Developer Duncan Ogilvie."

The developers of this tool offer two types of debugger applications: x64 for 64-bit applications and x32 for 32-bit applications. In this case, the actors used x32dbg.exe, which is the 32-bit debugger of x64dbg.

Upon execution of x32dbg.exe, Microsoft Windows will attempt to resolve any dependency files necessary to run the application. That search starts locally (i.e., in the current working directory). If found, the necessary files are loaded and executed.

X32bridge.dll is a Windows Dynamic Link Library (DLL) dependency file of x32dbg.exe. A legitimate x32bridge.dll also carries the same digital signature. In this case, the file is not signed.

Once loaded, the malware searches locally for an actor-created encrypted payload file: x32bridge.dat (SHA256: e72e49dc1d95efabc2c12c46df373173f2e20dab715caf58b1be9ca41ec0e172).

X32bridge.dat was first submitted to VirusTotal on Jan. 22, 2021. As of Dec. 15, 2022, it has a detection score of eight out of 61 engines. None of the engines identify the file as PlugX malware.

Once loaded and decrypted in memory, the malware infects the host and any removable USB devices attached with the PlugX malware. Figure 1 below illustrates PlugX DLL side loading using x64dbg DLL hijacking.

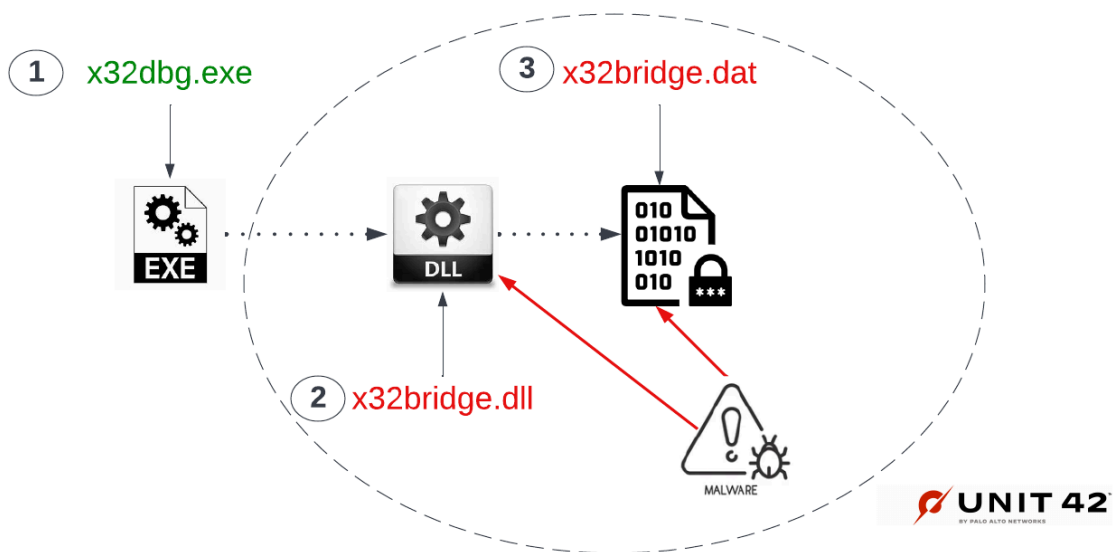


Figure 1. PlugX DLL side loading using x64dbg.

Both the hijacking of x64dbg and the association of this behavior with the PlugX malware were reported by Sophos back in [November 2020](#). Their blog refers to this malware as *KillSomeOne*, based on a Program Database (PDB) string found in one of the binaries.

Sophos performed an excellent analysis of the samples and touched on the USB infection. We confirmed that our sample matched the behaviors described in their report. From there, we wanted to expand our research by focusing

on the USB infection, other USB variants in the wild and links to the PlugX malware.

## PlugX Malware USB Overview

The technique used by the PlugX malware to hide files in a USB device involves using a certain Unicode character. This hinders Windows Explorer and the command shell (cmd.exe) from displaying the USB directory structure and any files, concealing them from the victim.

The Unicode character used by this PlugX malware for the directories is 00A0 (a whitespace character called a no-break space). The whitespace character prevents the Windows Operating System from rendering the directory name, concealing it rather than leaving a nameless folder in Explorer.

To achieve code execution of the malware from the hidden directory, a Windows shortcut (.lnk) file is created on the root folder of the USB device. The shortcut path to the malware contains the Unicode whitespace character, which is a space that does not cause a line break, but this is not visible when viewed via Windows Explorer, as shown below in Figure 2.

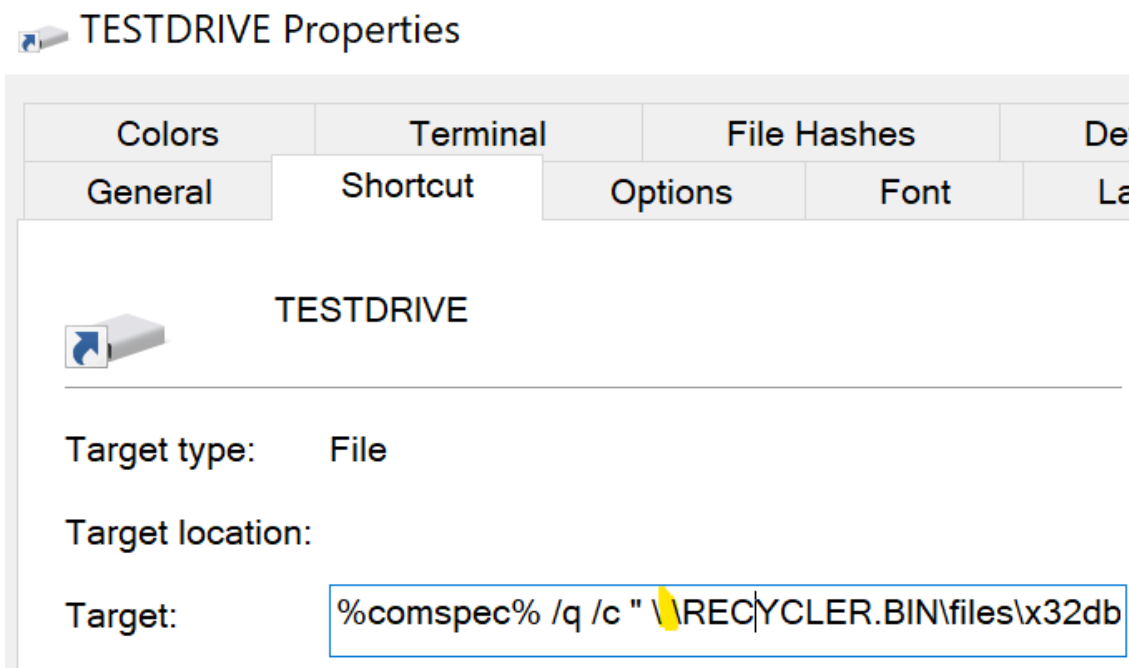


Figure 2. Windows shortcut properties for accessing the hidden directory.

When the shortcut file is viewed in a hex editor, the hex representation of these characters is shown as highlighted below in Figure 3.

```

00000000  43 3A 5C 57 69 6E 64 6F 77 73 5C 53 79 73 74 65  C:\Windows\Syste
00000010  6D 33 32 5C 63 6D 64 2E 65 78 65 00 00 29 00 2F  m32\cmd.exe..)/
00000020  00 71 00 20 00 2F 00 63 00 20 00 22 00 A0 00 5C  .q. ./ .c. ." . \
00000030  00 A0 00 5C 00 52 00 45 00 43 00 59 00 43 00 4C  . \.R.E.C.Y.C.L
00000040  00 45 00 52 00 2E 00 42 00 49 00 4E 00 5C 00 66  .E.R...B.I.N.\f
00000050  00 69 00 6C 00 65 00 73 00 5C 00 78 00 33 00 32  .i.l.e.s.\.x.3.2
00000060  00 64 00 62 00 67 00 2E 00 65 00 78 00 65 00 22  .d.b.g...e.x.e."
    
```

Figure 3. Windows shortcut properties shown in a hex editor display.

The PlugX malware uses the Component Object Model (COM) interface to create the .lnk files and includes the Unicode character 00A0. It does this by creating an instance of a shell desktop to create the associated Windows shortcut files. The ShellLink SetArguments method is used to set the command line arguments, which include the white space no break Unicode character, as shown in Figure 4 below.

```
73delle5 ff512c call dword ptr [ecx+2Ch] ds:002b:74df2a40=(windows storage) ShellLink::SetArguments (753323f0)
0:000:x86> db edi
0019e138 2f 00 71 00 20 00 2f 00-63 00 20 00 22 00 a0 00 /.q. ./c. ."...
0019e148 5c 00 a0 00 5c 00 52 00-45 00 43 00 59 00 43 00 \...\R.E.C.Y.C.
0019e158 4c 00 45 00 52 00 2e 00-42 00 49 00 4e 00 5c 00 L.E.R...B.I.N.\
0019e168 66 00 69 00 6c 00 65 00-73 00 5c 00 78 00 33 00 f.i.l.e.s.\x.3.
0019e178 32 00 64 00 62 00 67 00-2e 00 65 00 78 00 65 00 2.d.b.g...e.x.e.
0019e188 22 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ".....
0019e198 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
0019e1a8 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
```

Figure 4. COM ShellLink::SetArguments.

Finally, the *ShellLink::Save* method saves the shortcut file with all the changes.

### PlugX Malware USB Infection

The PlugX malware x32bridge.dll loads x32bridge.dat, which is responsible for implanting the host with malware and infecting any attached removable media USB devices such as floppy, thumb or flash drives. If a removable media device is found, the following steps are performed:

1. It creates the following directory structure:  
<usb volume>:\u00A0\u00A0\RECYCLER.BIN\files. Example: F:\ \RECYCLER.BIN\files.
2. It creates a hidden file named desktop.ini in each folder, which specifies the icon for the folder. This file contains the following data:

```
[.ShellClassInfo]
IconResource=%systemroot%\system32\SHELL32.dll,7
```

The Windows OS uses a single file to retrieve icon images that are displayed on the desktop as shortcuts or from Windows Explorer files and folders. The Shell32.dll file contains a list of icons and a unique number. In this case, the drive icon and the number 7 are used, as shown below in Figure 5.



Figure 5. Drive icon number 7.

The use of this drive icon makes the directories appear as drives within Windows Explorer when viewed with hidden files enabled. If deleted, the directories appear as folders.

3. In the second directory, it creates a subfolder named RECYCLER.BIN. This directory acts as a recycle bin. In that directory is a subdirectory named files and a hidden desktop.ini file. This desktop.ini file contains the following data:

[.ShellClassInfo]

CLSID = {645FF040-5081-101B-9F08-00AA002F954E}

This CLSID instructs Windows Explorer to display the created RECYCLER.BIN folder as a recycle bin directory by giving it the recycle bin icon. This is shown in Figure 11 as an example.

4. The files subfolder contains copies of the PlugX malware x32dbg.exe, any encrypted actor .dat files and the actor’s malicious DLL. An example of this is shown in Figure 6 below.

Name	Ext	Size	Created	Modified	Accessed	Attr	1st sector
x32bridge.dat	dat	125 KB	11/12/2022 13:39:44	07/03/2019 22:15:40	11/12/2022 22:15:40	A	8296
x32bridge.dll	dll	71.0 KB	11/12/2022 13:39:44	07/03/2019 23:10:46	11/12/2022 23:10:46	A	8552
x32dbg.exe	exe	52.8 KB	11/12/2022 13:39:44	11/26/2019 05:35:26	11/12/2022 05:35:26	A	8696

Figure 6. Infected UBS removable device displayed via WinHex.

It should be noted that pre-infection of the USB device, existing files or directories on the root of the removable device are moved to the second hidden folder on the USB device, as shown below in Figure 7.

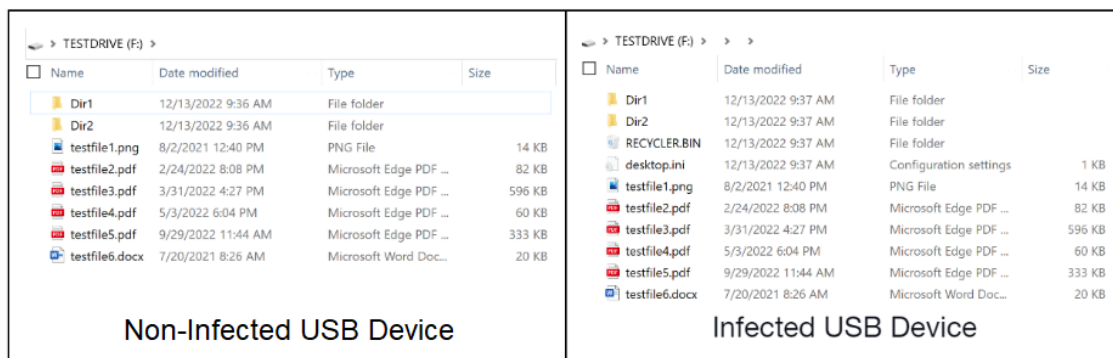


Figure 7. Side-by-side comparison of the root directories of a non-infected versus infected USB device.

5. Whenever the shortcut file from the infected USB device is clicked, the PlugX malware launches Windows Explorer and passes the directory path as a parameter. This then displays the files on the USB device from within the hidden directories and also infects the host with the PlugX malware. The victim sees their files and assumes all is working as expected.

Figure 8 below illustrates how an infected USB thumb drive would appear to a victim in Windows Explorer using default settings (i.e., not displaying hidden files).

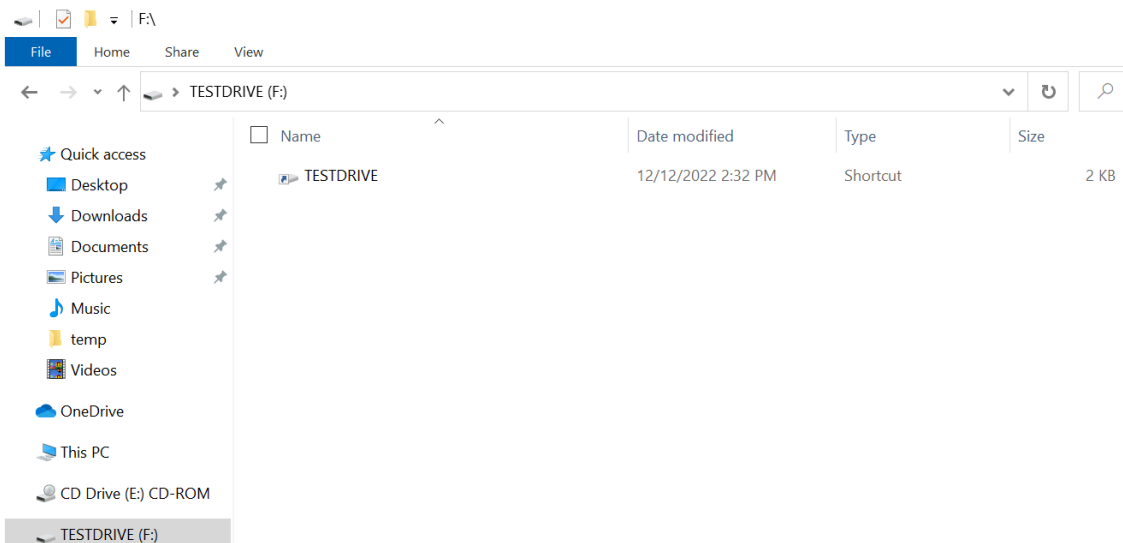


Figure 8. Windows Explorer showing infected USB Removable Device. (Show hidden files not enabled.)

An infected PlugX USB device has no files or directories in the root folder and contains only a .lnk file.

Figure 9 shows that the shortcut file is TESTDRIVE, which matches the USB device name. The shortcut file is responsible for infecting the host and navigating to the hidden directory on the USB device each time it is clicked. For example, the shortcut file referenced in Figure 9 contains the following data:

```
%comspec% /q /c " \RECYCLER.BIN\files\x32dbg.exe"
```

Once the target clicks the shortcut, x32dbg.exe is launched via cmd.exe from the hidden files directory on the USB device. The host the USB is attached to is now infected with the PlugX malware.

When viewing the contents of the infected USB device with Windows Explorer and hidden items enabled, a victim would see the following images:

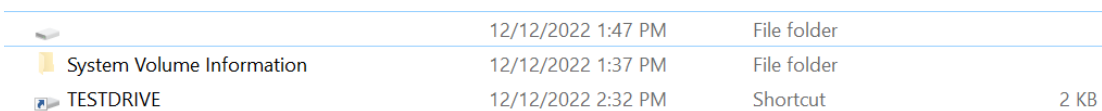


Figure 9. Windows Explorer showing infected USB Removable Device (root folder).

They would then see the following, as shown in Figure 10, within the first hidden directory:

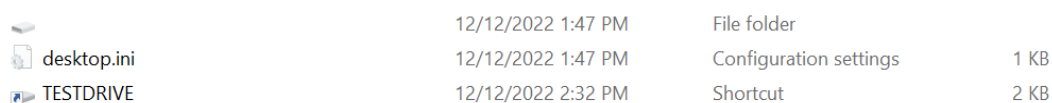


Figure 10. Windows Explorer showing an infected USB removable device (hidden directory number one).

As shown in Figure 11, Windows Explorer displays another hidden directory along with a shortcut file. The shortcut file has the same properties to launch x32dbg.exe and will infect the host with the USB device attached.

When viewing the contents of the next hidden folder, Windows Explorer shows the following directory structure:

Name	Date modified	Type	Size
RECYLER.BIN	12/12/2022 1:47 PM	File folder	
desktop.ini	12/12/2022 1:47 PM	Configuration settings	1 KB

Figure 11. Windows Explorer showing infected USB Removable Device (hidden directory number two).

This directory contains a RECYLER.BIN folder that masquerades as a Windows recycle bin, along with any files or directories that existed in the root folder of the USB device at the time of infection. When viewing the contents of the RECYLER.BIN with Windows Explorer, a victim will see what’s shown in Figure 12:

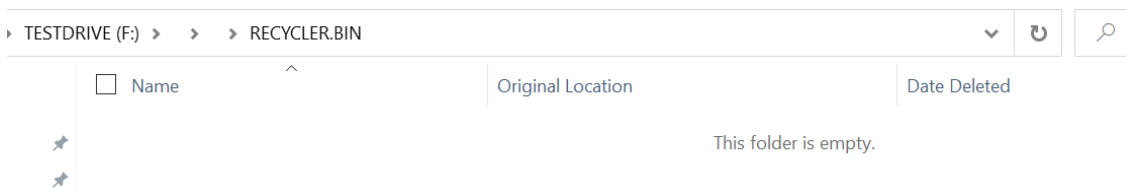


Figure 12. Windows Explorer showing infected USB Removable Device (RECYLER.BIN folder).

Since the default setting for Windows Explorer is not to show hidden items, the only item visible to the victim is the shortcut file, as shown in Figure 8. Even with hidden files enabled, Windows Explorer or cmd.exe cannot show the malware files that reside in the files subdirectory.

The malware files can only be viewed on a \*nix OS or by mounting the USB device in a forensic tool. Figure 13 shows how the USB device looks once it’s mounted in Ubuntu, browsing to it via File Explorer.

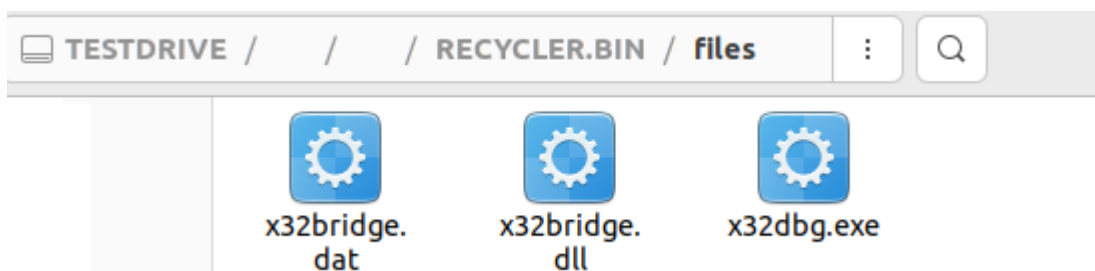


Figure 13. Ubuntu File Explorer viewing an infected USB device.

## PlugX Malware Post USB Infection

When a host is infected with this variant of the PlugX malware, the malware continuously monitors for USB removable devices. Once a USB device is discovered and infected, any new files written to the USB device root folder post-infection are moved to the hidden folder within the USB device. Since the Windows shortcut file resembles that of a USB device and the malware displays the victim's files, they unwittingly continue to spread the PlugX malware.

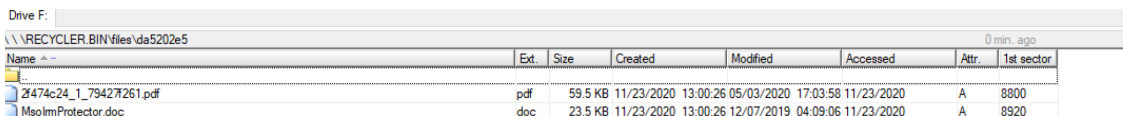
## PlugX Malware USB Variant Two

Armed with this knowledge, we wondered if other similar PlugX USB infection malware existed in the wild. File SHA256: 5b496972a86cea66aeecaac6e3f67a92e22f35cd5d2a98d54a2f1218fcd5dfc5 in VirusTotal matches the

behaviors of x32bridge.dat, specifically the creation of the Windows Scheduled Task and PDB string referenced in the Sophos blog.

This file is a Windows x86 PE file (DLL), the in-memory equivalent of x32bridge.dat. Our analysis shows the runtime behavior is identical to the x32bridge.dat USB infection detailed earlier, but it has an added capability. It copies documents from the host machine to a new hidden USB subfolder named da520e5.

The malware specifically copies all Adobe PDF and Microsoft Word documents from the host to this directory. Figure 14 below shows an example of the directory and files copied from our host.



Name	Ext	Size	Created	Modified	Accessed	Atr	1st sector
2f474c24_1_79427f261.pdf	pdf	59.5 KB	11/23/2020 13:00:26	05/03/2020 17:03:58	11/23/2020	A	8800
MeoImProtector.doc	doc	23.5 KB	11/23/2020 13:00:26	12/07/2019 04:09:06	11/23/2020	A	8920

Figure 14. PlugX malware variant with two exfil files.

This PlugX USB malware variant was designed to exfiltrate specific files from its target that could be retrieved later, as this directory and files are not used by the malware or displayed to the victim.

## Association With PlugX Malware

In addition to USB infection, both x32bridge.dat and the second PlugX malware variant discovered in VirusTotal check the target for specific running processes. If found, it terminates them and deletes the directories from which they were executed.

X32bridge.dat seeks out process names starting with AAM and, if found, terminates the process and deletes specific directories associated with the process. The Sophos blog speculated that this behavior likely removed older PlugX malware variants. Previous variants attributed to PlugX started with AAM file names (i.e., AAM Updates.exe).

The second variant we discovered in VirusTotal also performs this check, including one for the specific process file name AAM Updates.exe. Additionally, if AAM Updates.exe is found, it deletes the directory AAM UpdatesikB, which is a folder that has also been associated with the PlugX malware family.

Since both x32bridge.dat and the second PlugX malware variant discovered in VirusTotal share the same USB infection method along with other runtime behaviors, we can conclusively state that it is indeed associated with PlugX.

## Conclusion

PlugX malware has been used for over a decade and was historically extensively associated with Chinese nation-state APT groups. Over the years, it has been adopted and used by other threat groups, from nation-states to ransomware actors.

The typical tradecraft of PlugX uses benign files to achieve code execution, also known as DLL side loading, that many security vendors now detect and prevent. This might be part of why the actors added the capability to infect

any attached removable media USB devices such as floppy, thumb or flash drives as well as any additional systems the USB is later plugged into.

Any host infected with this variant of the PlugX malware will continuously monitor for new USB removable devices to infect. This PlugX malware also hides attacker files in a USB device with a novel technique, which makes the malicious files only viewable on a \*nix OS or by mounting the USB device in a forensic tool. Because of this ability to evade detection, the PlugX malware can continue to spread and potentially jump to air-gapped networks.

Additionally, we discovered a similar variant of PlugX in VirusTotal with the added capability of copying all Adobe PDF and Microsoft Word documents from the infected host to the USB device's hidden folder created by the PlugX malware. The discovery of these samples indicates PlugX development is still alive and well among at least some technically skilled attackers, and it remains an active threat.

For Palo Alto Networks customers, our products and services provide the following coverage associated with this campaign:

- [Cortex XDR](#) customers receive protection at the endpoints from the malware techniques described in this blog.
- [WildFire](#) cloud-based threat analysis service accurately identifies the malware described in this blog as malicious.

If you think you may have been compromised or have an urgent matter, contact the [Unit 42 Incident Response team](#) or call:

- North America Toll-Free: 866.486.4842 (866.4.UNIT42)
- EMEA: +31.20.299.3130
- APAC: +65.6983.8730
- Japan: +81.50.1790.0200

Palo Alto Networks has shared these findings, including file samples and indicators of compromise, with our fellow Cyber Threat Alliance members. CTA members use this intelligence to rapidly deploy protections to their customers and to systematically disrupt malicious cyber actors. Learn more about the [Cyber Threat Alliance](#).

## Unit 42 Managed Threat Hunting Queries

```
// Detecting potential DLL side loading

config case_sensitive = false timeframe = 30d

| dataset = xdr_data

| filter event_type = ENUM.LOAD_IMAGE and action_module_signature_status = 3 and
((action_module_path contains "mpsvc.dll" and actor_process_image_name = "aug.exe") or
(action_module_path contains "dismcore.dll" and actor_process_image_name = "aug.exe" ) or
```

```
(action_module_path contains "\hex.dll" and actor_process_image_name = "SafeGuard.exe") or  
(action_module_path contains "x32bridge.dll" and actor_process_image_name = "x32dbg.exe") or  
(action_module_path contains "x32bridge.dll" and actor_process_image_name = "Mediae.exe"))  
  
| comp count() as counter by actor_process_image_path , action_module_path , action_module_sha256
```

```
// Adding registry persistence  
  
config case_sensitive = false timeframe = 30d  
  
| dataset = xdr_data  
  
| filter event_type = ENUM.REGISTRY and event_sub_type = ENUM.REGISTRY_SET_VALUE and  
action_registry_key_name contains "Microsoft\Windows\CurrentVersion\Run" and (action_registry_data  
contains "x32dbg.exe" or action_registry_data contains "aug.exe" or action_registry_data contains  
"SafeGuard.exe")  
  
| fields action_registry_data , action_registry_file_path , action_registry_key_name ,  
actor_process_command_line , agent_hostname
```

```
// Payload execution with rundll32.exe  
  
config case_sensitive = false timeframe = 30d  
  
| dataset = xdr_data  
  
| filter event_type = ENUM.PROCESS and action_process_image_path contains "rundll32" and  
(action_process_image_command_line contains "\akm.dat" or action_process_image_command_line  
contains "\precious.dat")  
  
| fields action_process_image_command_line , actor_process_image_command_line , agent_hostname
```

## Indicators of Compromise

### Known PlugX Samples:

```
8ec37dac2beaa494dcefec62f0bf4ae30a6ce44b27a588169d8f0476bbc94115  
e72e49dc1d95efabc2c12c46df373173f2e20dab715caf58b1be9ca41ec0e172  
0e9071714a4af0be1f96cffc3b0e58520b827d9e58297cb0e02d97551eca3799  
39280139735145ba6f0918b684ab664a3de7f93b1e3ebcdd071a5300486b8d20  
41a0407371124bcad7cab56227078ccd635ba6e6b4374b973754af96b7f58119  
02aa5b52137410de7cc26747f26e07b65c936d019ee2e1afae268a00e78a1f7f
```

2a07877cb53404888e1b6f81bb07a35bc804daa1439317bccde9c498a521644c  
5d98d1193fcb2479668a24697023829fc9dc1f7d31833c3c42b8380ef859ff1

### **Known File Directories**

C:\ProgramData\UsersDate\Windows\_NT\Windows\user\Desktop\  
C:\Users\Public\Public Mediae\  
<usb volume>:\u00A0\u00A0\RECYCLER.BIN\files  
<usb volume>:\u00A0\u00A0\RECYCLER.BIN\files\da520e5

### **Known Windows Mutex Names**

LKU\_Test\_0.1  
LKU\_Test\_0.2  
TCP\_0.1

### **Known PlugX Encrypted Payload File Names**

akm.dat  
precious.dat  
x32bridge.dat  
Groza\_1.dat

### **Known Windows Scheduled Task Names**

LKUFORYOU\_1  
PRECIOUS\_0.1

### **Known Windows Process Names (Observed Abused Benign Files)**

x32dbg.exe  
x32dbge.exe  
Mediae.exe  
Aug.exe  
Precious.exe  
SafeGuard.exe  
Dism.exe

### **MITRE ATT&CK Techniques**

[ATOM PlugX Malware](#)

### **Additional Resources**

- [THOR: Previously Unseen PlugX Variant Deployed During Microsoft Exchange Server Attacks by PKPLUG Group](#)

- [Brute Ratel C4 Red Teaming Tool Being Abused by Malicious Actors](#)
- [NJCCIC Alert Gootloader Malware Platform Uses Sophisticated Techniques to Deliver Malware](#)
- [A new APT uses DLL side-loads to “KillSomeOne” – Sophos News](#)
- [Black Basta Ransomware Gang Infiltrates Networks via QAKBOT, Brute Ratel, and Cobalt Strike \(trendmicro.com\)](#)
- [Inside the OPM Hack, The Cyberattack that Shocked the US Government | WIRED](#)
- [ASCII Codes](#)
- [ASCII Table](#)

---

Source: <https://unit42.paloaltonetworks.com/plugx-variants-in-usbs/>