blog-en.itochuci.co.jp /entry/2025/03/12/140000

Hack The Sandbox: Unveiling the Truth Behind Disappearing Artifacts

ici-blog ⋮⋮ 3/12/2025

This post is also available in: 日本語

Introduction

The National Police Agency (NPA) and the National center of Incident readiness and Strategy for Cybersecurity (NISC) released a security advisory on January 8, 2025, regarding an APT attack campaign targeting organizations in Japan by "MirrorFace." The advisory highlights that MirrorFace exploited Windows Sandbox and Visual Studio Code, providing guidance on how to identify and detect traces of these activities. This article focuses on Windows Sandbox, one of the attack techniques used in this campaign. It provides detailed verification results, forensic artifacts, and key points useful for monitoring and investigation.
This article is based on the presentation "Hack The Sandbox: Unveiling the Truth Behind Disappearing Artifacts," at JSAC2025 on January 22, 2025.

Reference

Hack The Sandbox: Unveiling the Truth Behind Disappearing Artifacts

LilimRAT has been observed being used by the APT group “MirrorFace” (which is a subgroup of APT10 umbrella). LilimRAT is a customized version of the open-source Lilith RAT. It includes a function to check for the existence of the WDAGUtilityAccount user folder, and if this folder is not present, it will terminate.

```
29
30  FileAttributesA = GetFileAttributesA("C:\\Users\\WDAGUtilityAccount");
31  if ( FileAttributesA != -1 && (FileAttributesA & 0x10) != 0 )
32  {
33    c_GetModuleFileNameA();
34    c_WSAStartup();
35    v29 = 1;
36    Src = 0;
37    v13 = 0;
38    v14 = 0;
39    v24 = 0;
40    v25 = 15;
41    LOBYTE(v23[0]) = 0;
42    c_memmove(v23, &unk_440C40, 1u);
43    LOBYTE(v29) = 2;
44    v27 = 0;
```

Figure 1. LilimRAT code to check WDAGUtilityAccount user folder

Since WDAGUtilityAccount is used as the default user within Windows Sandbox, it is likely that LilimRAT was specifically designed to run only within Windows Sandbox.

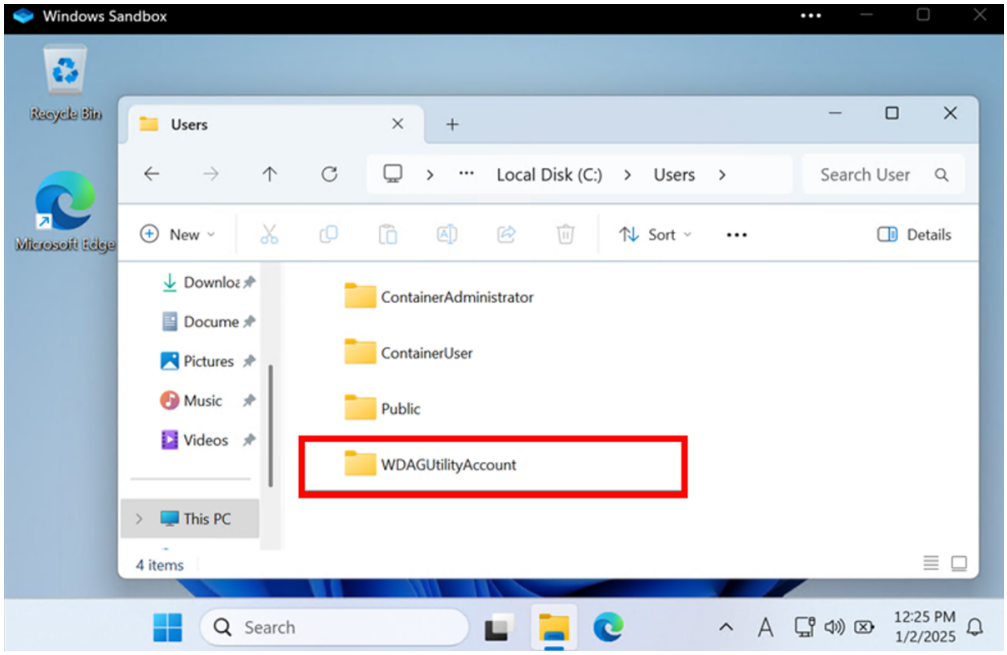

Figure 2. WDAGUtilityAccount user profile in Windows Sandbox

By default, Windows Sandbox is disabled. Therefore, after compromising a target machine, the attacker enables the Windows Sandbox feature. Since Windows Sandbox only becomes active after the host machine is restarted, the attacker creates a WSB (Windows Sandbox configuration) file on the system and then reboots the host machine. After the reboot, Windows Sandbox is available, and the malware (in this case, LilimRAT) is executed within the sandbox according to the WSB file's configuration, establishing communication with the C2 server.

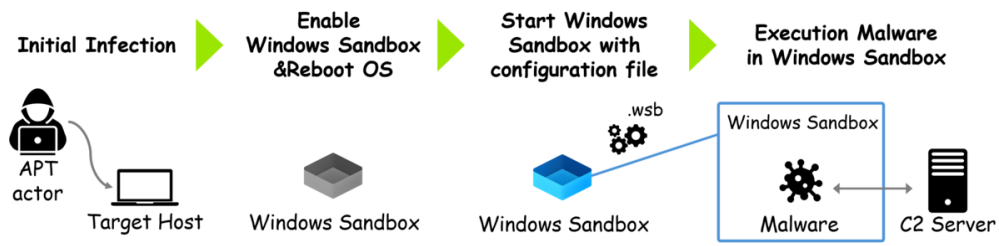

Figure 3. The process of abusing Windows Sandbox

Given the fact that Windows Sandbox is abused using this technique, we believe it's essential to conduct thorough technical verification to understand its specifications, uncover abusing methods, and establish investigation and countermeasure techniques.

About Windows Sandbox

Windows Sandbox is a virtual environment isolated from the host system, allowing users to safely test files and applications. It's like an additional OS running as software within the host OS. This feature is available on Windows

10 (Build 18342 and later) and Windows 11. Below are some key specifications regarding Windows Sandbox.

Windows Enable Windows Sandbox

By default, Windows Sandbox is disabled and must be enabled via the GUI or CLI before use.

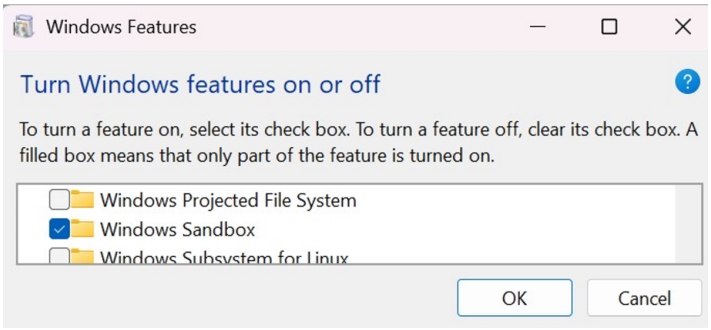

Figure 4. Enabling Windows Sandbox settings on optional features (GUI)

Once Windows Sandbox is enabled, the sandbox environment becomes available for use as shown below.

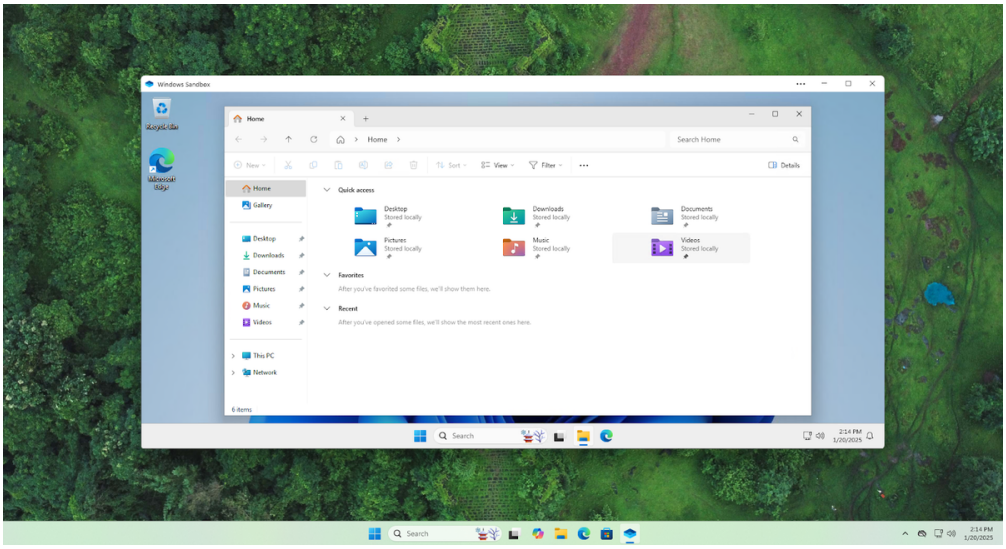

Figure 5. Windows Sandbox

Default user

Windows Sandbox runs with WDAGUtilityAccount user. This user belongs to the Administrators group.

```
C:\Users\WDAGUtilityAccount>net user WDAGUtilityAccount
User name                    WDAGUtilityAccount
Full Name
Comment                      Windows Defender Application Guard
User's comment
~ Redacted ~
Local Group Memberships      *Administrators *Remote Desktop Users *Users
Global Group memberships     *None
```

Windows Defender settings

Windows Defender is disabled by default within Windows Sandbox and cannot be enabled via either the GUI or PowerShell commands.

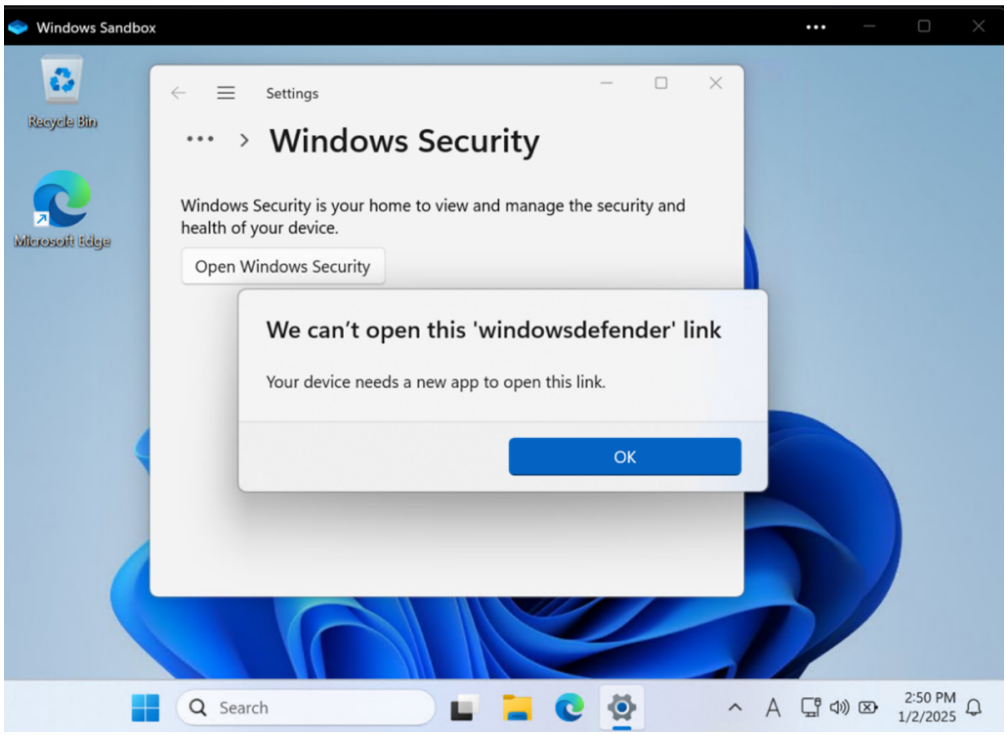

Figure 6. Windows Defender settings

Configuration file (.wsb)

A WSB file is an XML-based configuration file that defines the settings for Windows Sandbox. Below is an example of a WSB file.

```
<Configuration>
 <Networking>Enable</Networking>
  <MappedFolders>
    <MappedFolder>
      <HostFolder>C:\Users\Public\Downloads</HostFolder>
      <SandboxFolder>C:\Users\WDAGUtilityAccount\Downloads</SandboxFolder>
    <ReadOnly>false</ReadOnly>
    </MappedFolder>
  </MappedFolders>
  <LogonCommand>
    <Command>explorer.exe</Command>
  </LogonCommand>
  <MemoryInMB>1024</MemoryInMB>
</Configuration>
```

In this example, the following settings are configured:

- Enable network access
- Share the host machine's C:\Users\Public\Downloads folder with the sandbox's C:\Users\WDAGUtilityAccount\Downloads folder
- Open File Explorer upon startup
- Allocate 1 GB of memory to the sandbox

A WSB file offers many other customizable settings for Windows Sandbox. The sandbox starts with the configurations defined in this file.

Configuration item	Description
vGPU	Enables or disables GPU sharing.
Networking	Enables or disables networking in the sandbox.
Mapped folders	An array of folders, each representing a location on the host machine that is shared with the sandbox at the specified path.
Logon command	Specifies a single command that will be invoked automatically after the sandbox logs on.

Configuration item	Description
Audio input	Shares the host's microphone input into the sandbox.
Video input	Shares the host's webcam input into the sandbox.
Protected client	Adds a new layer of security boundary by running Sandbox inside an AppContainer Isolation execution environment.
Printer redirection	Enables or disables printer sharing from the host into the sandbox.
Clipboard redirection	Enables or disables sharing of the host clipboard with the sandbox.
Memory in MB	Specifies the amount of memory that the sandbox can use in MB.

Virtual Hard Disk (VHDX)

Windows Sandbox is built using VHDX (Virtual Hard Disk) files and employs a differential backup mechanism. When the sandbox starts, a VHDX-related folder is created under `C:\ProgramData\Microsoft\Windows\Containers`. This folder contains both the parent virtual disk and differential virtual disks, along with multiple subfolders. When the sandbox is closed, the differential backups are deleted.

```
C:\ProgramData\Microsoft\Windows\Containers\
                                                    * An example chain in Windows 11
<Parent>
    .\ContainerStorages\fbd7ba93-6b62-44cb-a59e-0cc2c59b697a\sandbox.vhdx
                                  ↑ Parent Path
<Differencing>
    .\ContainerStorages\2a3c6d32-561c-4096-b099-73945cff9665\sandbox.vhdx
                                  ↑ Parent Path
<Differencing>
    .\ContainerStorages\a73d9514-1027-4ead-a3cb-6009ac2e8f3e\sandbox.vhdx
```

Figure 7. VHDX chain on `C:\ProgramData\Microsoft\Windows\Containers` folder

The attack methods

Given the previously explained Windows Sandbox specifications, we will now detail the attack flow carried out by MirrorFace after Windows Sandbox has been enabled. The attacker places three files—a BAT file, an archiver, and an archive file—in any folder on the compromised host machine. Then, they enable Windows Sandbox, restart the system, and execute the WSB file to initiate the attack.

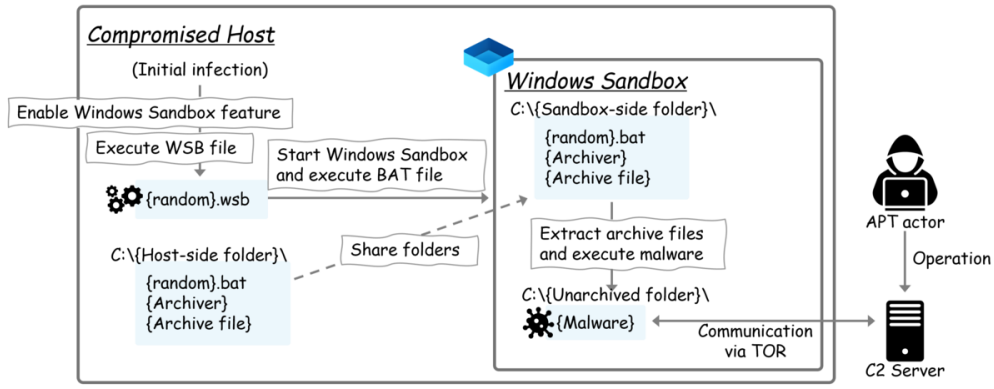

Figure 8. The attack flow using Windows Sandbox

Below is an example of the WSB file used in the attack.

```
<Configuration>
 <Networking>Enable</Networking>
  <MappedFolders>
    <MappedFolder>
    <HostFolder>C:\{Host-side folder}</HostFolder>
    <SandboxFolder>C:\{Sandbox-side folder}</SandboxFolder>
    <ReadOnly>false</ReadOnly>
    </MappedFolder>
  </MappedFolders>
  <LogonCommand>
    <Command>C:\{Sandbox-side folder}\{random}.bat</Command>
```

```
  </LogonCommand>
  <MemoryInMB>1024</MemoryInMB>
</Configuration>
```

This WSB file contains the following configurations:

- Folder sharing (shares a folder of the host machine with the Windows Sandbox)
- Enables network connection
- Executes a BAT file upon startup
- Allocates 1 GB of RAM

When the WSB file is executed, Windows Sandbox starts, and the BAT file runs automatically. The script written in the BAT file is then executed within the sandbox. The BAT file contains commands to extract the archive file and execute the extracted malware. If successful, the attacker gains control over the sandbox environment on the compromised host via a C2 server. Below is an example of a BAT file used by the attacker. In this example, 7-Zip is used as the archiver, but this is just one of the tools observed—other archiving tools have also been used in similar attacks.

```
@echo off
C:\{Sandbox-side folder}\7z.exe x C:\{Sandbox-side folder}\{Archive file} -oC:\
{Unarchived folder}\ -p{Password} -y
schtasks /create /tn {taskname} /tr "C:\{Unarchived folder}\{Malware}" /sc hourly /st
08:30 /ru system /f
schtasks /run /tn {taskname}
```

When Windows Sandbox is executed, a window is displayed. However, if it is launched via Task Scheduler under a different user account (e.g., SYSTEM privileges), it runs in the background without a window, making it extremely difficult to detect its activities. Additionally, malware executed within the sandbox communicates with the C2 server via the Tor network. The use of the Tor network is likely intended to encrypt communications and conceal the C2 server. Since the malware in Windows Sandbox operates according to the WSB file's configuration, it can access files on the host machine. However, because the files are accessed from the sandbox, any activity doesn't get logged by monitoring tools running on the host system. Furthermore, various attack tools used within the sandbox remain undetected, as Windows Defender is not active in this environment. This allows attackers to operate in an environment free from security products.

Emerging threats

During our investigation into Windows Sandbox features and attacks abusing it, we identified significant functional updates. While Microsoft's official documentation only mentions this new feature as a preview, it does not provide detailed information about the update. However, as of the time of writing, we have confirmed that a version including this feature has already been released.

> Windows Sandbox Client Preview] New! This update adds the Windows Sandbox Client Preview. It includes:
>
> - Runtime clipboard redirection
> - Audio and video input control
> - The sharing of folders with the host at runtime
>
> To access these, select the ellipses (…) at the upper right on the app. This preview also includes a version of command-line support. (The commands might change over time). To learn more, use the wsb.exe–help command. You can find new updates for this app in the Microsoft Store. This might not be available to all users because it will roll out gradually.
>
> October 24, 2024—KB5044384 (OS Build 26100.2161) Preview

The changes to Windows Sandbox after the Windows 11 update are as follows: Addition of the wsb.exe command, enabling sandbox execution via the command line Background execution of the sandbox Ability to modify certain settings via the GUI

Figure 9. Windows Sandbox configuration menu on the updated Windows 11

As shown in the example below, wsb.exe allows for starting Windows Sandbox, enumerating running instances, and connecting to a sandbox session.

```
> wsb.exe start
Windows Sandbox environment started successfully:
Id:  7f1397ca-3b46-416a-827a-a4a5b76e880e
> wsb.exe list
7f1397ca-3b46-416a-827a-a4a5b76e880e
> wsb.exe connect --id 7f1397ca-3b46-416a-827a-a4a5b76e880e
```

`wsb.exe` command options are as follows:

Commands, alias	Options	Description
StartSandbox, start	--id -c, --config	Starts an instance of Windows Sandbox.
ListRunningSandboxes, list	-	Lists the IDs of all running Windows Sandbox environments.
Execute, Exec	--id (REQUIRED) -c, --command (REQUIRED) -d, --working-directory <working-directory> -r, --run-as <ExistingLogin|System> (REQUIRED)	Executes a command in the running Windows Sandbox environments.
ShareFolder, share	--id (REQUIRED) -f, --host-path <host-path> (REQUIRED) -s, --sandbox-path <sandbox-path> -w, --allow-write	Shares a folder from the host to the Windows Sandbox session.
StopSandbox, stop	-	Terminates a running Windows Sandbox.
ConnectToSandbox	--id	Starts a remote session for a Windows Sandbox environment.
GetIpAddress, ip	--id (REQUIRED)	Gets the IP address of the Windows Sandbox environment.

Below is an example for the execution result of the `wsb.exe start` command.

```
> wsb.exe start -c "<Configuration> <Networking>Enable</Networking><MappedFolders>
<MappedFolder><HostFolder>C:\Users\Public\Downloads</HostFolder>
<SandboxFolder>C:\Users\WDAGUtilityAccount\Desktop</SandboxFolder>
<ReadOnly>false</ReadOnly></MappedFolder></MappedFolders><LogonCommand>
<Command>C:\Users\WDAGUtilityAccount\Desktop\a.bat</Command></LogonCommand>
```

```
<MemoryInMB>1024</MemoryInMB></Configuration>"
Windows Sandbox environment started successfully:
Id: c2d290db-5986-4c06-bd7b-05f35f091fa4
```

These recent feature updates may make it more difficult to detect attacks leveraging Windows Sandbox. The key reasons for this are as follows:

1. Background execution of Windows Sandbox Previously, in Windows 10 and early versions of Windows 11, Windows Sandbox always ran as a foreground GUI application. However, with the new wsb.exe start command, it can now run in the background. As a result, the sandbox can be launched without user awareness, and its window remains hidden until the wsb.exe connect command is executed.

2. Sandbox execution without a WSB file The updated wsb.exe command allows sandbox configurations to be set via command-line arguments. Previously, WSB files were an important forensic artifact during investigations, but this change increases the risk of leaving no trace of sandbox usage.

3. Persistent data inside the sandbox In earlier versions, closing the Windows Sandbox window would terminate the process and delete all data within the environment. However, after the update, closing the window does not stop the sandbox, and its data remains intact. To delete data, the sandbox must be explicitly stopped using the wsb.exe stop command or terminated by shutting down the host machine. This change significantly increases the potential for long-term attacker operations within the sandbox. Given these updates, security researchers must carefully verify whether such feature changes improve convenience for attackers and implement appropriate countermeasures when new functionalities are introduced.

Monitoring and Investigation for Windows Sandbox

We have discussed the features of Windows Sandbox and the attack techniques that leverage it. Based on our verification, we will now explain effective countermeasures against attacks that abuse Windows Sandbox.

Monitoring

Monitoring for host machine and network

Monitoring the host machine from the following perspectives can be useful for detecting compromises:

- Monitor client operation logs, sandbox processes, and memory
- Tracking activities related to WSB files
- Observing event logs related to the sandbox

Additionally, Windows Sandbox utilizes the host machine's network adapter. As a result, if malware inside the sandbox communicates with a C2 server, the host machine's IP address will be used as the source IP address. This means standard network monitoring can still be effective. However, if Tor is used, implementing a detection mechanism for Tor network traffic will be necessary.

Monitoring Windows Sandbox execution

The following are processes related to Windows Sandbox. By monitoring the execution of these processes on the host machine, it’s possible to detect the startup of Windows Sandbox.

Process Names	Paths
WindowsSandbox.exe	C:\Windows\System32\WindowsSandbox.exe
WindowsSandboxClient.exe	C:\Windows\system32\WindowsSandboxClient.exe
cmproxyd.exe	C:\Windows\system32\cmproxyd.exe

Process Names	Paths
WindowsSandboxServer.exe	C:\Program Files\WindowsApps\MicrosoftWindows.WindowsSandbox_0.3.1.0_x64__cw5n1h2txyew
WindowsSandboxRemoteSession.exe	C:\Program Files\WindowsApps\MicrosoftWindows.WindowsSandbox_0.3.1.0_x64__cw5n1h2txyew
wsb.exe	C:\Users{USERNAME}\AppData\Local\Microsoft\WindowsApps\wsb.exe

Verification of processes in the sandbox that is deployed in the host machine's memory

When we ran Mimikatz inside the sandbox, we observed that, as shown in the figure below. The vmmemWindowsSandbox process on the host machine contained Mimikatz strings. Additionally, scanning the dumped vmmemWindowsSandbox process using Yara also detected the same Mimikatz strings. From this, we can conclude that sandbox processes are executed within the vmmemWindowsSandbox process. This suggests that memory scanning on the host machine's vmmemWindowsSandbox process can be used to detect malware or tools running inside the sandbox.

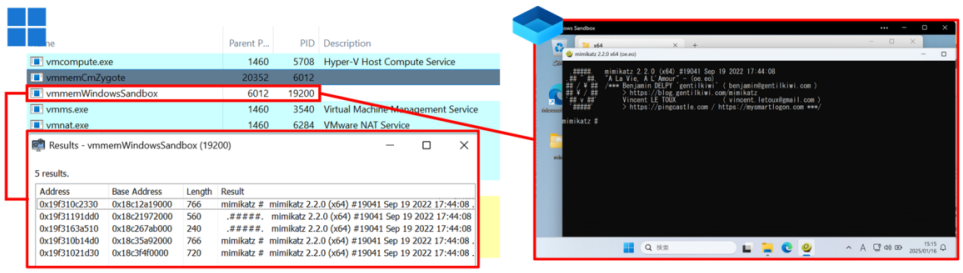

Figure 10. Exposed part of Windows Sandbox memory on the host machine

It has also been found that the process name where the sandbox memory is allocated varies depending on the Windows version. The differences are summarized in the table below.

Process name	OS
vmmem	Windows 10
vmmemWindowsSandbox	Windows 11

Investigation methods

Host machine investigation

Apart from the vmmemWindowsSandbox / vmmem process mentioned above, traces of Windows Sandbox activity are rarely recorded on the host machine. However, certain traces related to the sandbox's activation and startup may remain on the host. Therefore, the following artifacts can be examined to identify potential indicators of sandbox activity.

Classification	Description
$MFT	Creations of WSB file, mount source folders and files, and VHDX files are recorded.
$UsnJrnl	The creation of the WSB file, the creation of the mount source folder and file, and the creation of the VHDX file are recorded.
Prefetch	Loading of WSB and VDHX files may be recorded.

Classification	Description
Registry	The application associations are set . - HKLM\SOFTWARE\Classes\Applications\WindowsSandbox.exe - HKLM\SOFTWARE\Classes\Windows.Sandbox\shell\open\command - HKLM\SOFTWARE\Microsoft\Windows Sandbox\Capabilities\FileAssociations

EventLog records the activities related to the sandbox on the host machine.

Classification	Evtx	Source	Event ID	Description
Eventlog	System	Microsoft-Windows-Hyper-V-VmSwitch	102	Virtual machine network driver settings
			232	Virtual machine NIC port related information
			233	Virtual machine NIC related information
	Security	Microsoft-Windows-Security-Auditing	4624	•An account was successfully logged on. •Account Domain : NT VIRTUAL MACHINE •Process Name : C:\Windows\System32\vmcompute.exe
			4648	•A logon was attempted using explicit credentials. •Account Domain : NT VIRTUAL MACHINE •Process Name : C:\Windows\System32\vmcompute.exe
			4672	•Special privileges assigned to new logon. •Account Domain : NT VIRTUAL MACHINE
	Microsoft-Windows-Hyper-V-Worker-Admin	Microsoft-Windows-Hyper-V-SynthStor Microsoft-Windows-Hyper-V-Worker	12148	Virtual machine startup information
			12582	
			12597	Virtual network connection information
			18500	Virtual machine startup information
			18502	Information about powering down virtual machines
			18516	Virtual machine suspension information
			18596	Virtual machine restore information
			18601	Virtual machine startup information
			18609	Virtual machine initialization information
	Microsoft-Windows-Hyper-V-Worker-Operational	Microsoft-Windows-Hyper-V-VSmb	301	Information about the folder from which to mount the virtual machine
	Microsoft-Windows-Hyper-V-Compute-Operational	Microsoft-Windows-Hyper-V-Compute	2500	Process creation and command execution related information
	Setup	Microsoft-Windows-Servicing	9	Selectable update Containers-DisposableClientVM of package Microsoft-Windows-Containers-OptionalFeatures was successfully turned on.
			13	A reboot is necessary before the selectable update Containers-DisposableClientVM of package Microsoft-Windows-Containers-OptionalFeatures can be turned on.
	Microsoft-Windows-VHDMP-Operational	Microsoft-Windows-VHDMP-Operational	1	Information about virtual disks (mount/unmount/online/offline, etc.)
			2	
			12	
			17,18	
			22~28	
			31~34	
			50,51	

Figure 11. EventLog and Event ID for Windows Sandbox

Investigation for Windows Sandbox

Windows Sandbox is composed of VHDX files. Therefore, if the VHDX file is retrieved from the host machine while preserving the parent and differential virtual disk chain, it can be mounted to analyze the data within the sandbox. If any sandbox-related processes are detected, all folders associated with the VHDX file should be preserved as volatile data for further investigation.

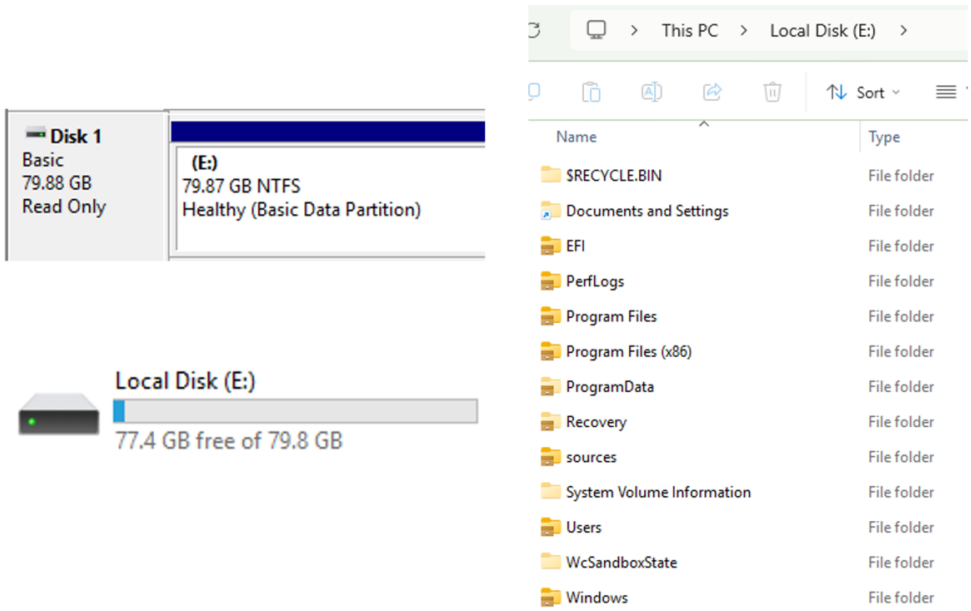

Figure 12. Mounted VHDX file and allocated drive

We mounted the VHDX file of the Windows Sandbox and examined artifacts that could be useful for forensic investigations. As a result, while some artifacts were disabled or not logged, we found that many valuable artifacts remained, which can be highly useful for forensic analysis.

Classification	Available	Description
$MFT	Yes	No operations on shared folders from the host were recorded.
$UsnJrnl	Yes	No operations on shared folders from the host were recorded.
Prefetch	No	Not recorded.

Classification	Available	Description
Registry	Yes	We were unable to confirm any Amcache updates during our test.
Browser History	Yes	The browsing history of the pre-installed Edge was confirmed. The browsing history was also retained for browsers installed by the user.
SRUM	No	Not recorded.
Evtx	Yes	The default log storage size is 20,480 KB, and some useful events (such as task schedules) are not recorded. We observed logons such as successful logon (Event ID 4624), failed logon (Event ID 4625), logon with explicit credentials (Event ID 4648), and service installation (Event ID 7045).

Control measures

Since Windows Sandbox is disabled by default, it is recommended to keep it in a disabled state. However, in case it gets enabled, monitoring and detecting events related to its activation is highly recommended.

Don’t grant administrative privileges to users

Enabling Windows Sandbox requires administrator privileges. Therefore, if it is not needed for business purposes, restricting administrator privileges for users can prevent unauthorized activation of the sandbox. Additionally, if Windows Sandbox is unintentionally enabled, monitoring the related event logs, as mentioned earlier, can help detect its activation.

Apply AppLocker policy

AppLocker is a security feature in Windows OS that prevents users from running unauthorized applications. Even if Windows Sandbox is already enabled or a user has permission to enable it, AppLocker can be used to control its execution.

> Reference
>
> AppLocker

By distributing a policy to the host machine that defines Windows Sandbox as an unauthorized application, it can block its execution. Additionally, when AppLocker blocks Windows Sandbox, the execution result is recorded in the event log.

Evtx	Source	Event ID	Description
Microsoft-Windows-AppLocker/EXE and DLL	Microsoft-Windows-AppLocker	8002	Indicates an AppLocker rule allowed the .exe or .dll file.
		8003	Indicates that AppLocker recorded the .exe or .dll file listed on an AppLocker policy. Shown only when Audit only enforcement mode is enabled.
		8004	AppLocker blocked the named EXE or DLL file. Shown only when the Enforce rules enforcement mode is enabled.

Figure 13. Event ID of AppLocker

Conclusions

In this article, we have provided a detailed analysis of Windows Sandbox, which was abused by MirrorFace in 2024. Based on our technical investigation, we explored its features, abusing techniques, forensic investigation methods, and defense strategies. Since antivirus solutions and EDR on the host machine may not detect threats within the sandbox, proactive measures such as enhanced monitoring, thorough investigations, and effective management are essential to minimize risks. Threat actors often exploit blind spots and gaps beyond our expectations. It’s important to continue making efforts to predict the unexpected and counter threats.

Acknowledgements

This research is supported by the following organizations:

- National Police Agency
- Information-technology Promotion Agency (J-CRAT)
- ESET, spol. s.r.o.

References

- Operation AkaiRyū: MirrorFace invites Europe to EXPO 2025 and revives ANEL backdoor

- MirrorFaceによるサイバー攻撃について（注意喚起）
- 別添資料【Windows Sandbox を悪用した手口及び痕跡・検知策】

Appendix

References

Test environment

Host OS version	Windows Sandbox version	wsb version
Microsoft Windows 10 Enterprise 10.0.19045	Microsoft Windows 10 Enterprise 10.0.19041	N/A
Microsoft Windows 10 Enterprise 10.0.19045	Microsoft Windows 10 Enterprise 10.0.19045	N/A
Microsoft Windows 11 Pro 10.0.26100	Microsoft Windows 11 Enterprise 10.0.26100	0.3.1.0