

# You snooze you lose: RPC-Racer winning RPC endpoints against services

---

Ron Ben Yizhak  
Security Researcher, SafeBreach



# About Me



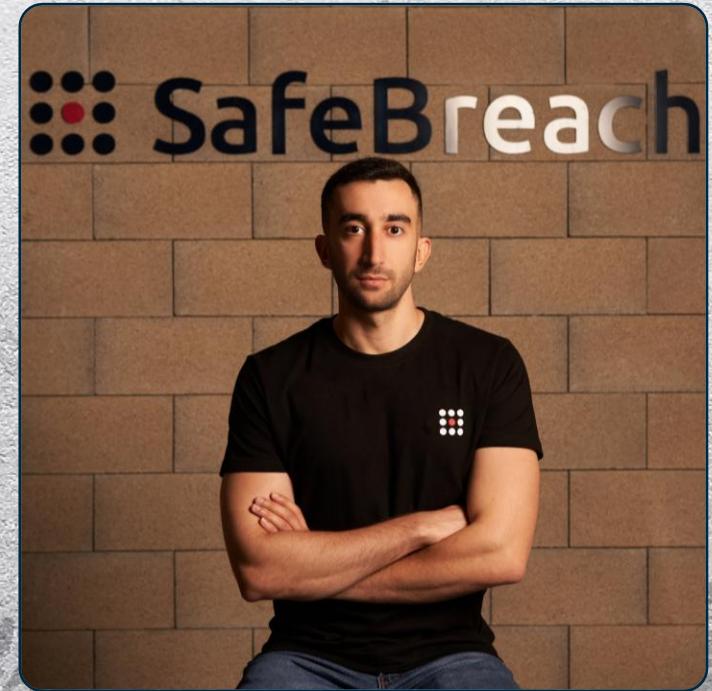
Security Researcher @ SafeBreach



Interested in reverse engineering, exploitation techniques and logical vulnerabilities



Enjoys rock climbing and playing piano

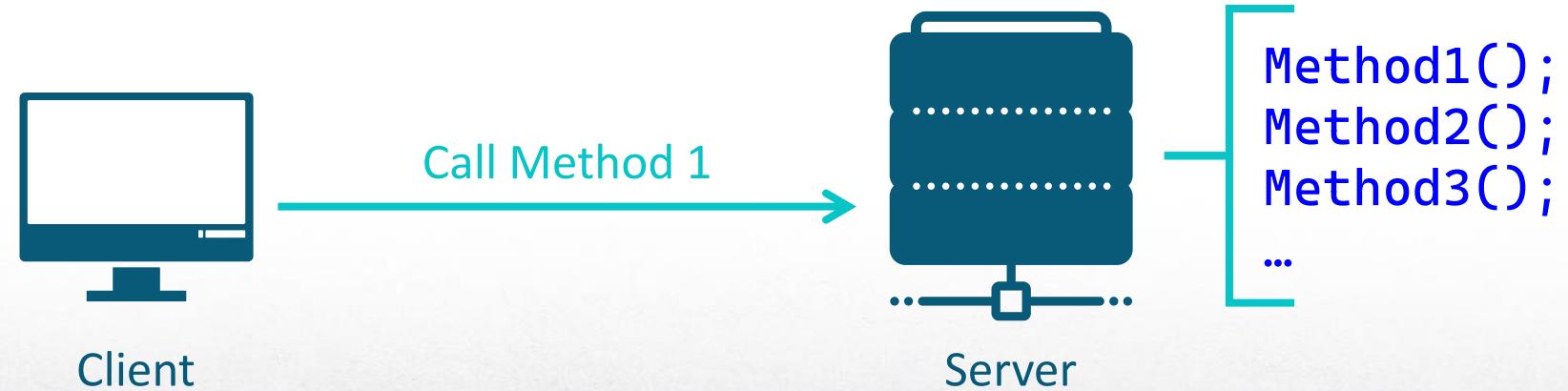


# Agenda

1. Intro – RPC fundamentals
2. Poisoning core RPC component
3. Performing recon for vulnerable servers
4. Manipulating built-in services
5. Leveraging machine account NTLM authentication
6. Conclusion

# RPC Fundamentals

- Remote Procedure Call
- The server exposes functionalities
- The client asks the server to execute functions



# RPC Fundamentals

RPC interfaces are defined by IDL files

```
[  
    uuid(8d864136-6900-4894-aece-66d455b552de),  
    version(1.0),  
]  
interface MyRpcInterface  
{  
    long SendRpcMessage([in, string] const wchar_t* Message);  
}
```

# RPC Fundamentals

Binding Handles components

ncacn\_ip\_tcp:ronb-vm[8888]

ncalrpc:ronb-vm[MyRpcEndpoint]

ncacn\_np:ronb-vm[\pipe\MyNamedPipe]

ProtSeq

NetworkAddr

Endpoint

# RPC Fundamentals

## Well-known endpoints vs dynamic endpoints

svhost.exe (1544) Properties

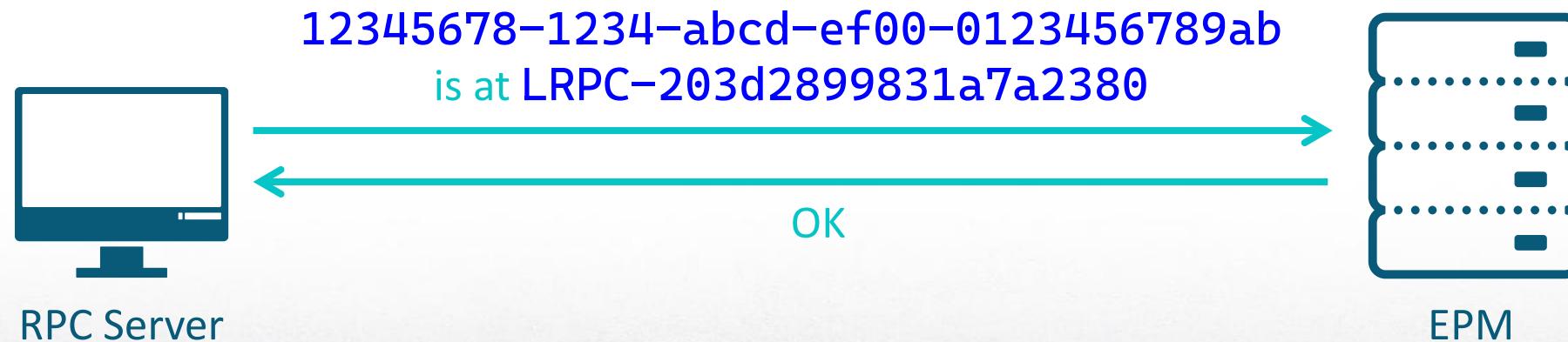
General Statistics Performance Threads Token Modules Memory Environment Handles

Options

Type	Name
ALPC Port	Connection: \RPC Control\OLE6DF1C8637045BD17C110F0CA98F5 DCOM
ALPC Port	Connection: \RPC Control\LSMApi Well-Known Endpoint
ALPC Port	Connection: \RPC Control\LRPC-13d920216e1f4f6d5e Dynamic Endpoint

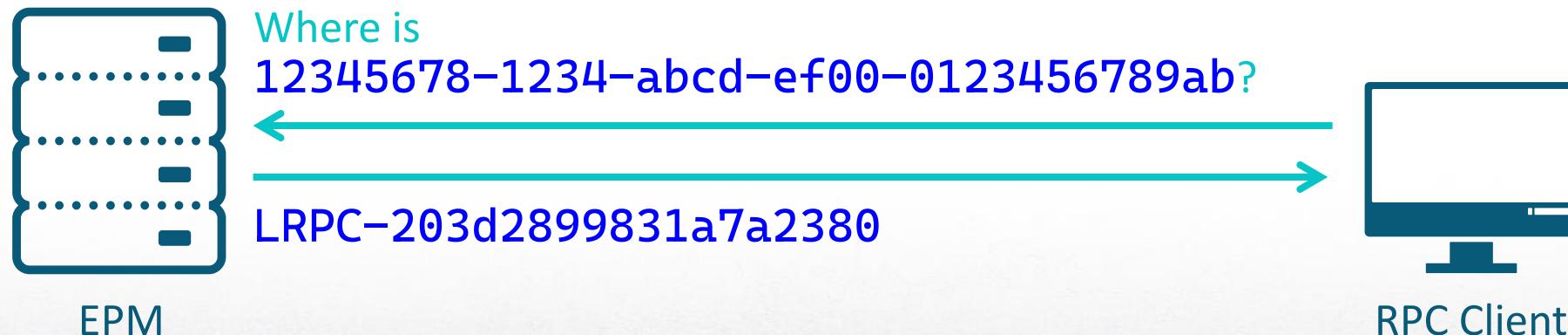
# RPC Fundamentals

- The Endpoint Mapper (EPM)
- The server registers UUID to endpoint



# RPC Fundamentals

- The Endpoint Mapper (EPM)
- The client queries UUID to endpoint



# RPC Fundamentals

- The Endpoint Mapper (EPM)
- Implemented in C:\Windows\System32\RpcEpMap.dll
- Hosted by RpcSs

Protocol	Name
ncacn_ip_tcp	135
ncacn_np	\pipe\epmapper
ncalrpc	epmapper



# RPC Fundamentals

## The Endpoint Mapper (EPM)

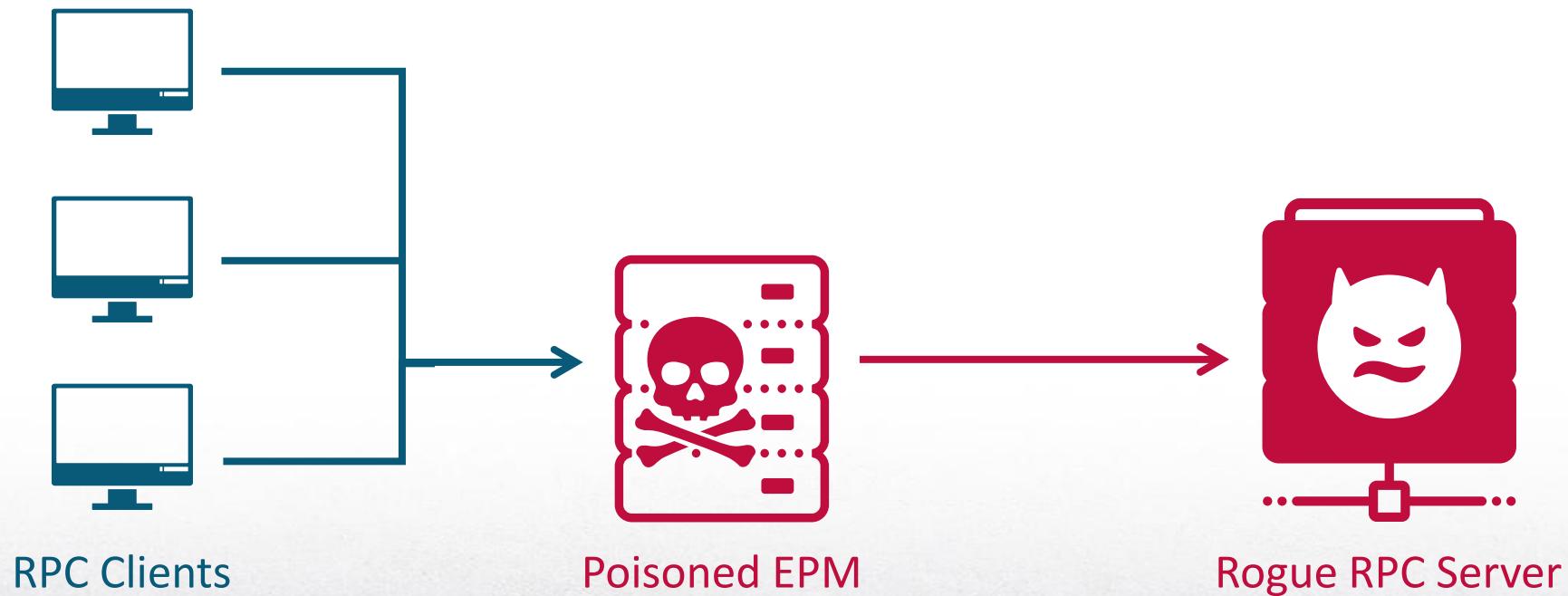


```
(kali㉿kali)-[~]
$ impacket-rpcdump ronb-insider
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

[*] Retrieving endpoint list from ronb-insider
Protocol: N/A
Provider: N/A
UUID      : 51A227AE-825B-41F2-B4A9-1AC9557A1018 v1.0 Ngc Pop Key Service
Bindings:
    ncacn_ip_tcp:172.28.134.141[49664]
    ncalrpc:[samss lpc]
    ncalrpc:[SidKey Local End Point]
    ncalrpc:[protected_storage]
    ncalrpc:[lsasspirpc]
    ncalrpc:[lsapolICYlookup]
    ncalrpc:[LSA_EAS_ENDPOINT]
    ncalrpc:[LSA_IDPEXT_ENDPOINT]
    ncalrpc:[lsacap]
    ncalrpc:[LSARPC_ENDPOINT]
    ncalrpc:[securityevent]
    ncalrpc:[audit]
    ncacn_np:\RONB-INSIDER[\pipe\lsass]
    ncalrpc:[imsfk]
    ncalrpc:[clipsfk]
```

# RPC Fundamentals

## The Endpoint Mapper (EPM)



# Previous RPC Exploits

Most exploitation tools target RPC Servers





We'll target RPC clients

# Research Goals

Poison  
the EPM

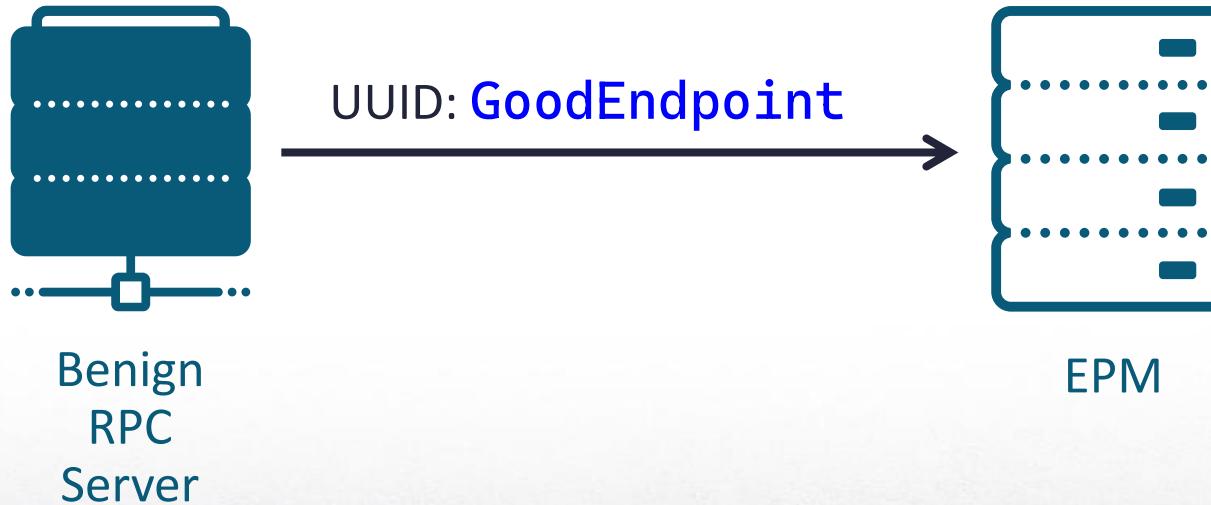
Masquerade as  
a legitimate RPC  
server

Manipulate RPC  
clients

Achieve  
local/domain  
privilege  
escalation

# Register as built-in interface

Registration is made by calling `RpcEpRegister`



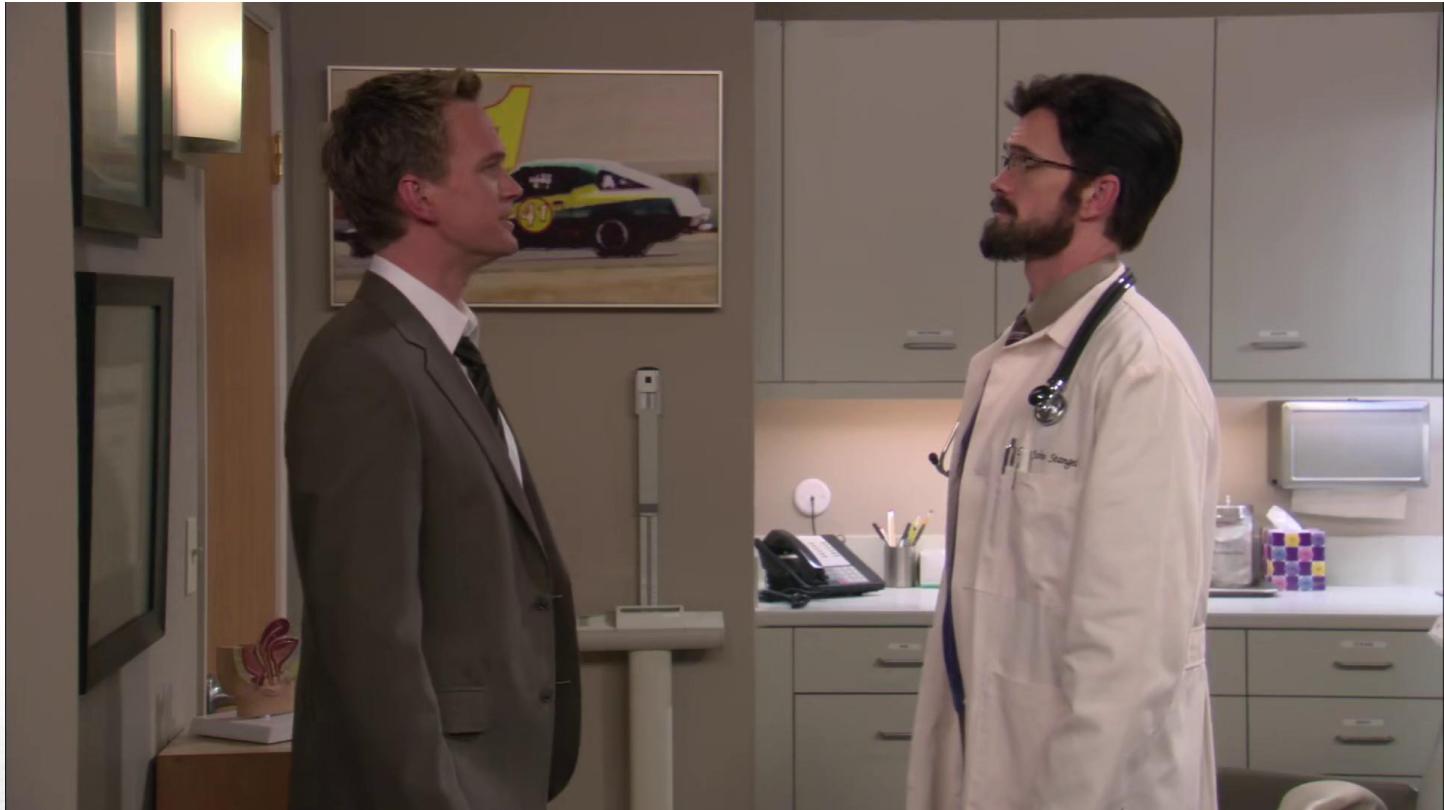
# Register as built-in interface

The rogue server will mimic this behavior



# Register as built-in interface

There is no verification  
on registering built-in  
interfaces



# Register Before a legitimate server

---

If the service is not running, we can register first

---

Registering first causes clients to connect to us

---



RPC client

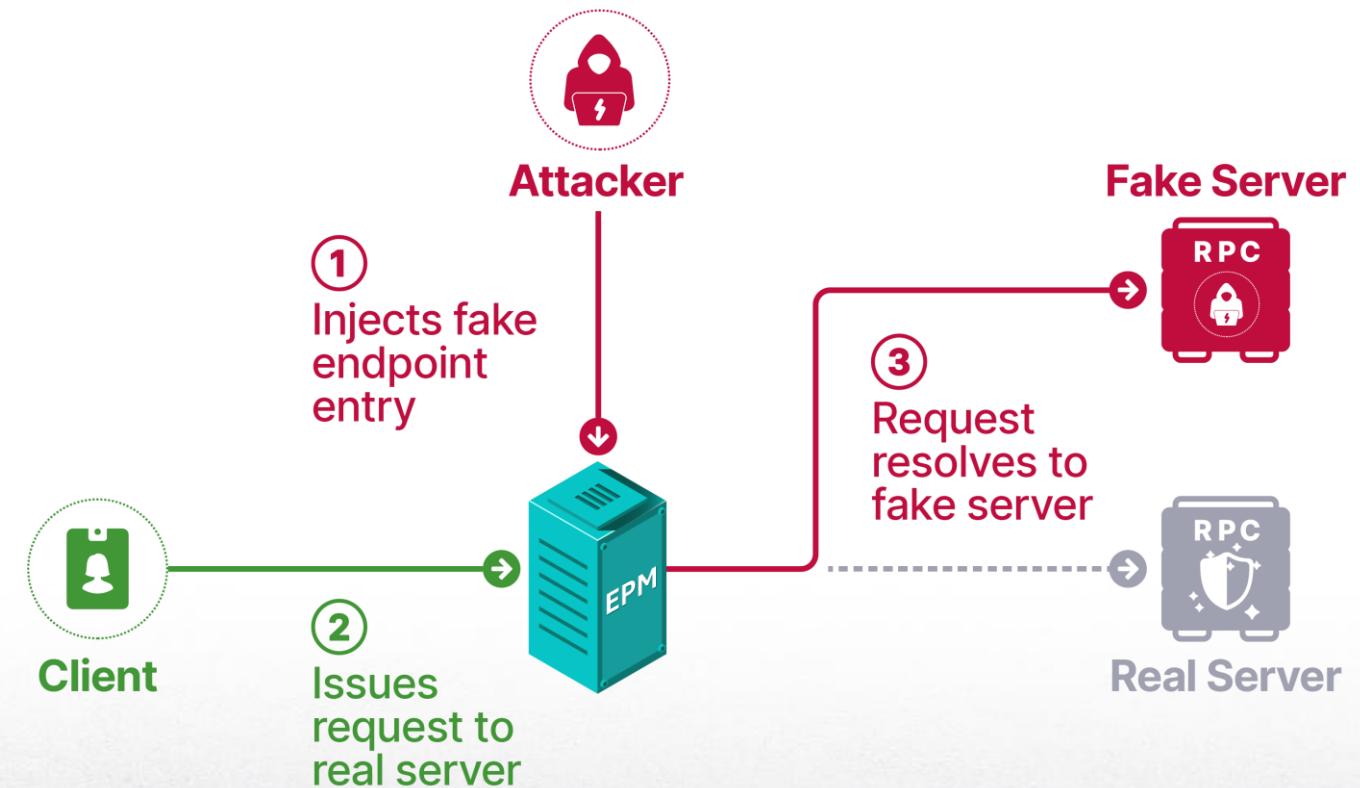
They're the same picture.

# EPM Poisoning

Novel manipulation technique

Destabilize the core of MSRPC

Doesn't require admin privileges



# Finding Delayed Services

Services with  
manual startup pose  
a security risk

Their RPC interface won't  
be registered on boot

System Informer [RONB-INSIDER\user]++ (Administrator)						
System View Tools Users Help						
Refresh Options		Find handles or DLLs		System information		
Processes	Services	Network	Disk	Firewall	Devices	
Name	PID	Display name	Type	Status	Start type	
edgeupdate		Microsoft Edge Update Service (edge...)	Own process	Stopped	Demand start (delay...	
dmwappushservice		Device Management Wireless Applica...	Share process	Stopped	Demand start (delay...	
dcsvc		Declared Configuration(DC) service	Own process	Stopped	Demand start (delay...	
WinRM		Windows Remote Management (WS-...)	Share process	Stopped	Demand start (delay...	
MSDTC		Distributed Transaction Coordinator	Own process	Stopped	Demand start (delay...	
BITS		Background Intelligent Transfer Service	Share process	Stopped	Demand start (delay...	
XboxGipSvc		Xbox Accessory Management Service	Share process	Stopped	Demand start (trigger)	
XblGameSave		Xbox Live Game Save	Share process	Stopped	Demand start (trigger)	
wuauserv		Windows Update	Share process	Running	Demand start (trigger)	
WPDBusEnum		Portable Device Enumerator Service	Share process	Stopped	Demand start (trigger)	
wlpasvc		Local Profile Assistant Service	Share process	Stopped	Demand start (trigger)	
wlidsvc		Microsoft Account Sign-in Assistant	Share process	Running	Demand start (trigger)	
wisvc		Windows Insider Service	Share process	Stopped	Demand start (trigger)	
WFDSConMgrSvc		Wi-Fi Direct Services Connection Man...	Share process	Stopped	Demand start (trigger)	
WerSvc		Windows Error Reporting Service	Own process	Stopped	Demand start (trigger)	
WEHOSTSVC		Windows Encryption Provider Host Se...	Share process	Stopped	Demand start (trigger)	
webthreatdefsvc		Web Threat Defense Service	Share process	Running	Demand start (trigger)	
WebClient		WebClient	Share process	Stopped	Demand start (trigger)	
WbioSrv		Windows Biometric Service	Share process	Stopped	Demand start (trigger)	
WarpJITSvc		Warp JIT Service	Own process	Stopped	Demand start (trigger)	

# Finding Delayed Services

The EPM can  
be queried  
programmatically

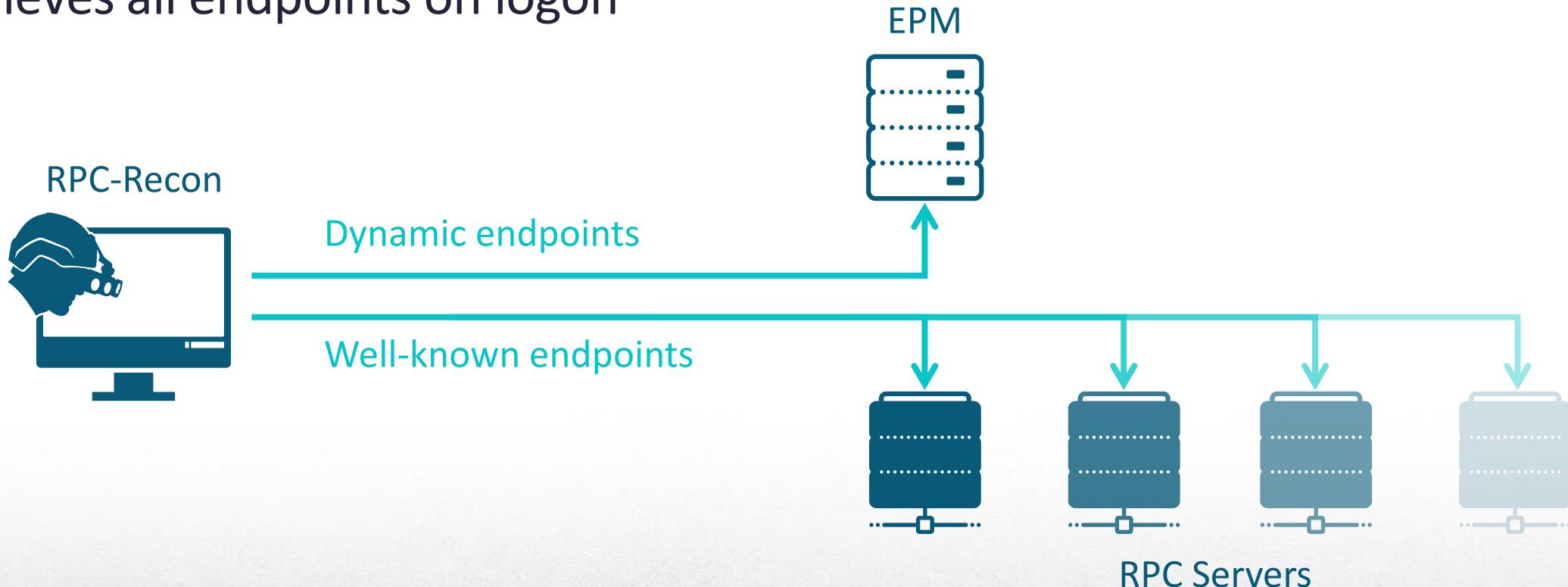
Well-known endpoints  
need to be extracted from  
memory



Interfaces				
Pid	Uuid	Location	Procs	
1976	2a82bb21-e44f-4791-9aa1-dfae788e2f43	C:\Windows\System32\ubpm.dll	4	
1976	33d84484-3626-47ee-8c6f-e7e98b113be1	C:\Windows\System32\WPTaskSchedul...	13	
1976	86d35949-83c9-4044-b424-db363231fd0c	C:\Windows\System32\schedsvc.dll	20	
1976	3a9ef155-691d-4449-8d05-09ad57031823	C:\Windows\System32\schedsvc.dll	7	
1484	326731e3-c1c0-4a69-ae20-7d9044a4ea5c	C:\Windows\System32\profsvc.dll	8	
1484	c9ac6db5-82b7-4e55-ae8a-e464ed7b42...	C:\Windows\System32\sysntfy.dll	15	
1484	18f70770-8e64-11cf-9af1-0020af6e72f4	C:\Windows\System32\combbase.dll	5	
2064	7ea70bcf-48af-4f6a-8968-6a440754d5fa	C:\Windows\System32\nsisvc.dll	9	
2100	18f70770-8e64-11cf-9af1-0020af6e72f4	C:\Windows\System32\combbase.dll	5	

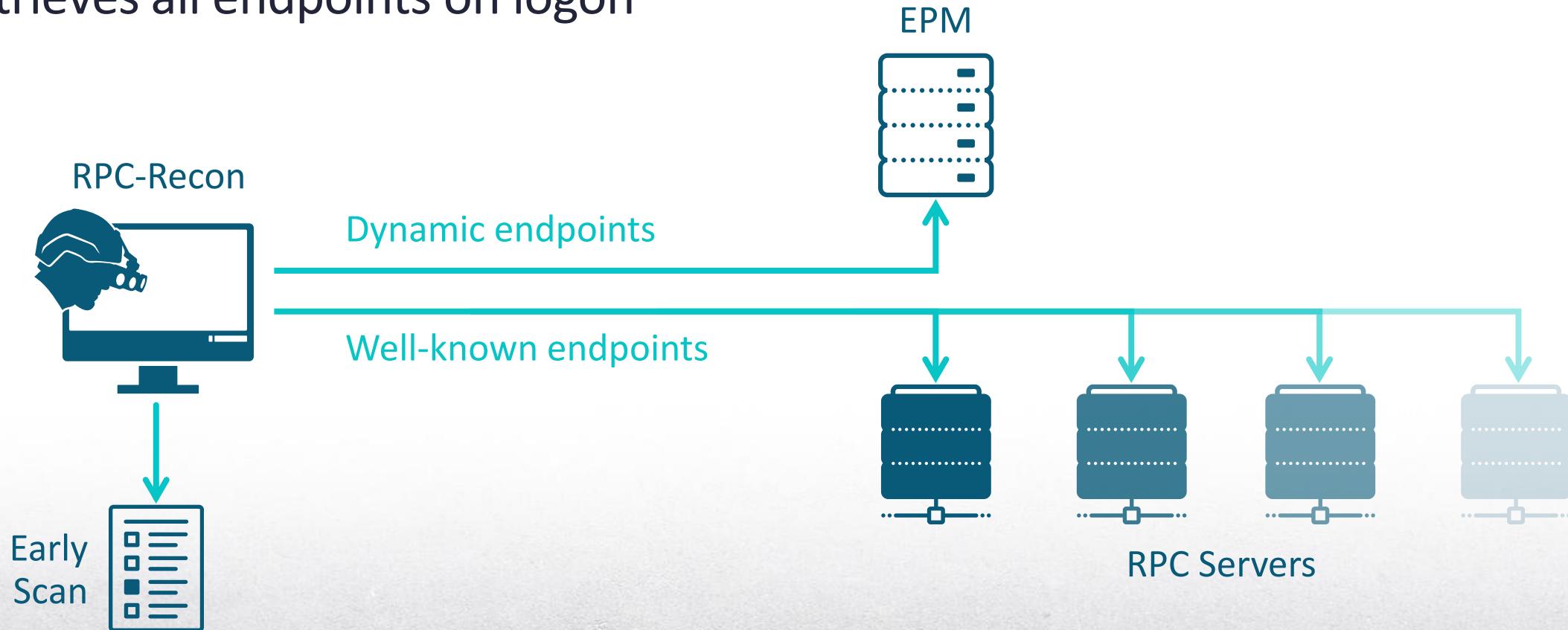
# RPC-Recon

Retrieves all endpoints on logon

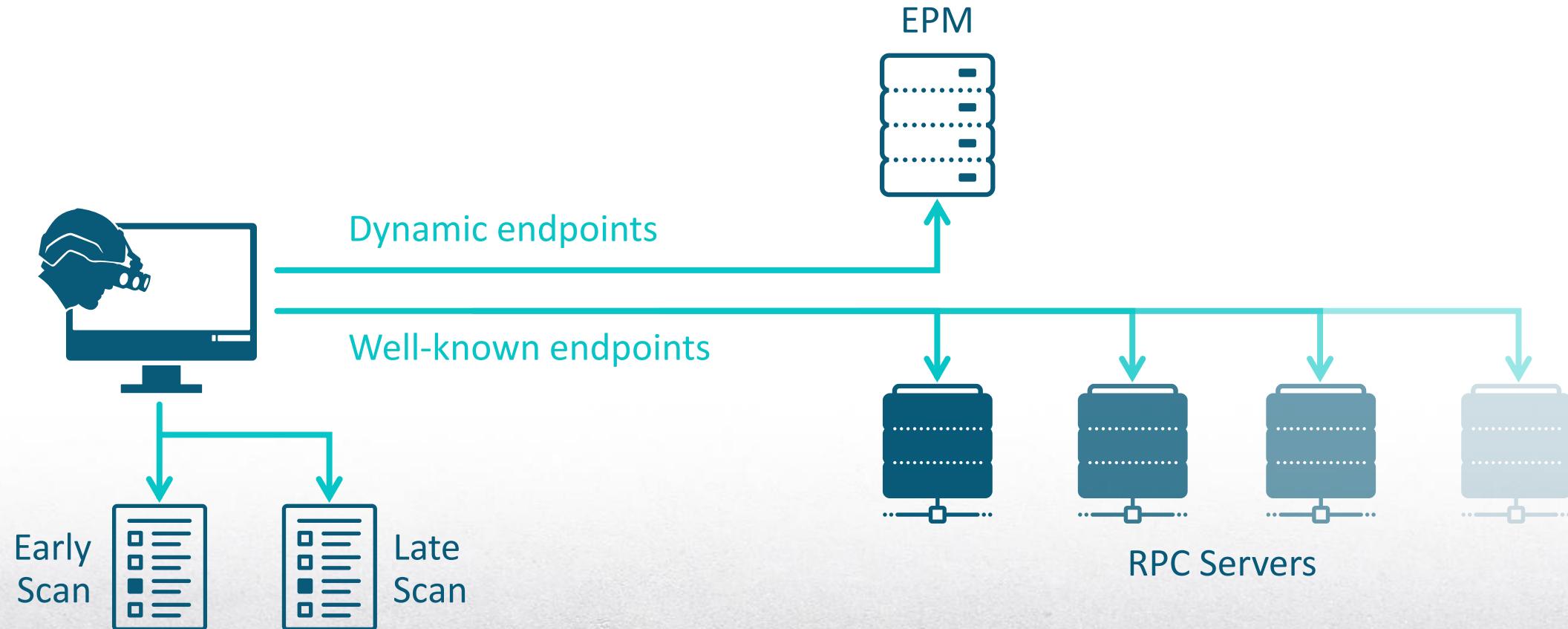


# RPC-Recon

Retrieves all endpoints on logon

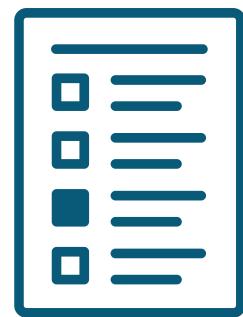


# RPC-Recon



# RPC-Recon

Comparing the lists reveals vulnerable interfaces



Late  
Scan

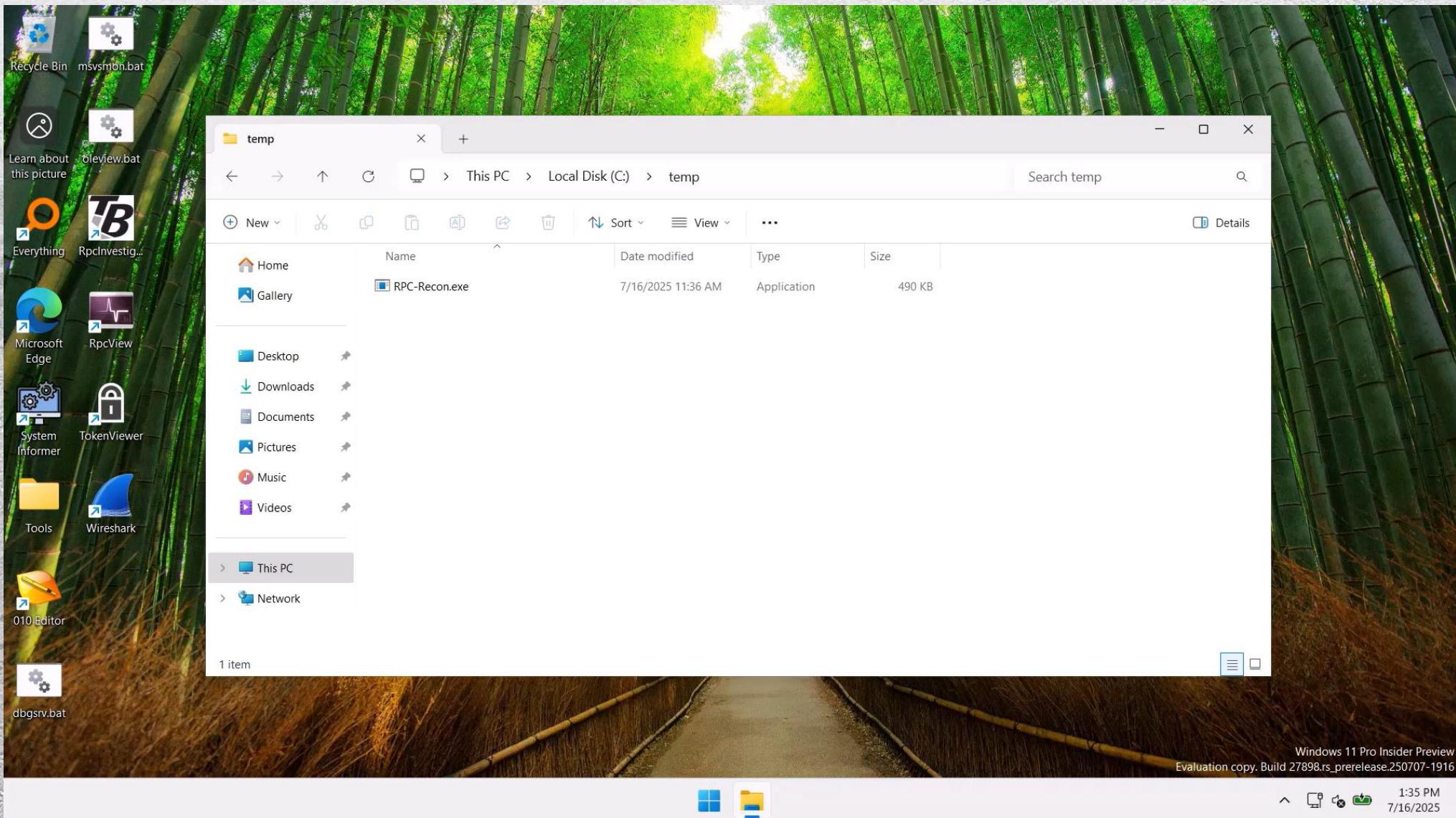


Early  
Scan



Vulnerable  
Interfaces

# RPC-Recon Demo



# Launching Rogue Server

---

Known interfaces were registered without admin privileges

---



---

The server received connections

---



---

The clients were services running as  
“NT AUTHORITY\SYSTEM”!

---



# Leveraging The Connections

---

Can `RpcImpersonateClient` be used?

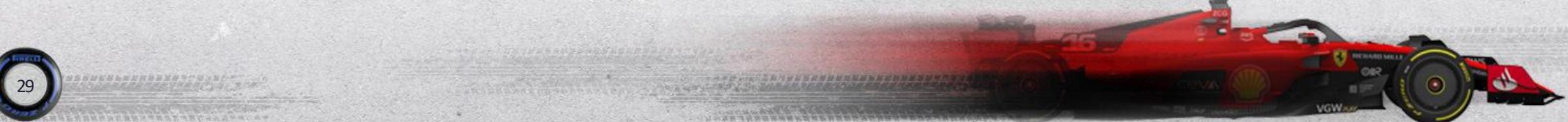
---

The level received is Identification

---

WinAPI will fail due to `ERROR_BAD_IMPERSONATION_LEVEL`

---



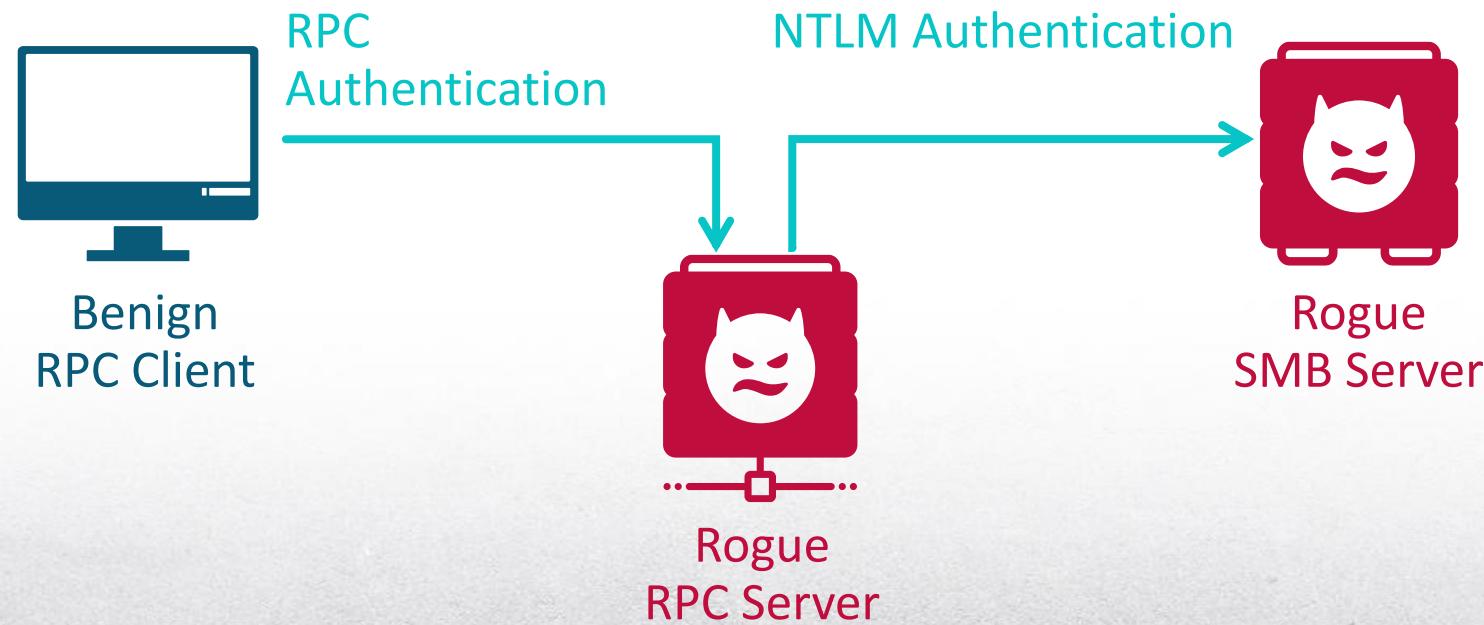
# Leveraging The Connections

```
typedef enum _SECURITY_IMPERSONATION_LEVEL {  
    SecurityAnonymous,  
    SecurityIdentification,  
    SecurityImpersonation,  
    SecurityDelegation  
} SECURITY_IMPERSONATION_LEVEL, * PSECURITY_IMPERSONATION_LEVEL;
```



# Leveraging The Connections

- Maybe we can force an NTLM authentication?
- Accessing remote resources requires delegation





# Looking for Credentials

- Can we gain credentials by registering the right interface?
- We can imitate services used for authentication
- No connections received due to Security mechanisms

Service	Interface	Method
VaultSvc	bb8b98e8-84dd-45e7-9f34-c3fb6155eed	VltAddItem
wlidsvc	cc105610-da03-467e-bc73-5b9e2937458d	WLIDSetAuthData, WLIDCreateIdentity
OneSyncSvc	923c9623-db7f-4b34-9e6d-e86580f8ca2a	AccountsMgmtRpcCreateAccount

# RPC Security Mechanisms

---

The client can specify the expected privileges of the server

---

The RPC runtime will verify it before the connection

---

```
typedef struct _RPC_SECURITY_QOS_V4_W {  
    unsigned long Version;  
    unsigned long Capabilities;  
    unsigned long IdentityTracking;  
    unsigned long ImpersonationType;  
    unsigned long AdditionalSecurityInfoType;  
    union  
    {  
        RPC_HTTP_TRANSPORT_CREDENTIALS_W *HttpCredentials;  
    } u;  
    void *Sid;  
    unsigned int EffectiveOnly;  
} RPC_SECURITY_QOS_V4_W, *PRPC_SECURITY_QOS_V4_W;
```

# RPC Security Mechanisms

```
*&sid.Revision = 0x101;
*&sid.IdentifierAuthority.Value[2] = 0x5000000;
sid.SubAuthority[0] = 0x12;
SecurityQOS.Version = 5;
SecurityQOS.Capabilities = 1;
memset(&SecurityQOS.IdentityTracking, 0, 24);
v3 = RpcStringBindingComposeW(0LL, L"ncalrpc", 0LL, 0LL, L"Security=impersonation dynamic false", StringBinding);
v4 = v3 <= 0;
if ( v3
    || (v3 = RpcBindingFromStringBindingW(StringBinding[0], &Binding), v4 = v3 <= 0, v3)
    || (v3 = RpcBindingSetAuthInfoExW(Binding, 0LL, 6u, 0xAu, 0LL, 0, &SecurityQOS), v4 = v3 <= 0, v3) )
```

wlidcli.dll (CWLIDBinding::Bind)



# Steps Taken



Attack

Impersonating privileged clients

Masquerading as authentication service



Defense

Impersonation policy

Security quality of service



# Manipulating File Access

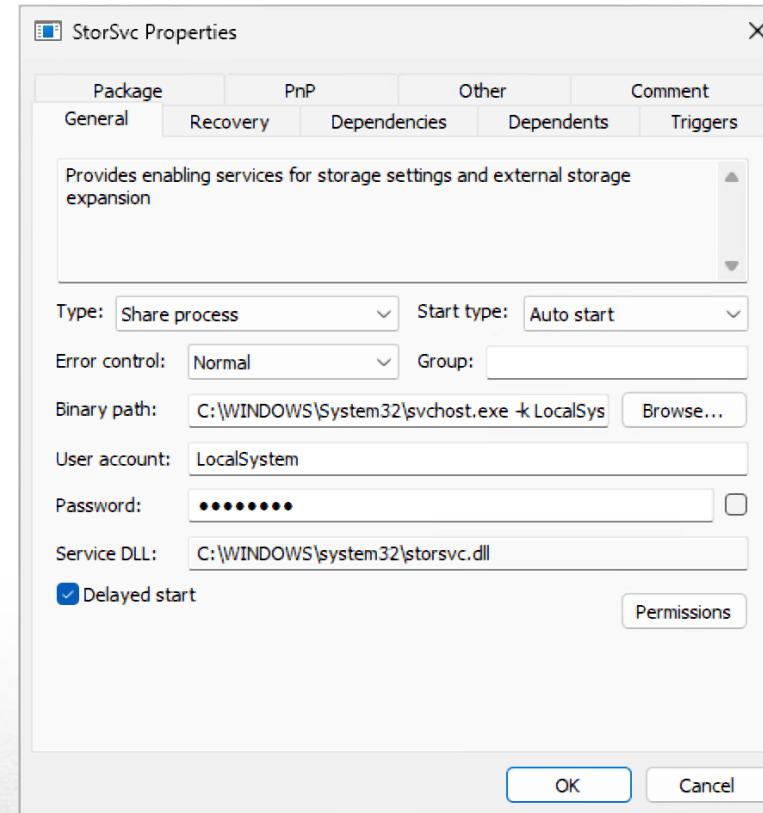
---

Can we achieve arbitrary write to protected directories?

---

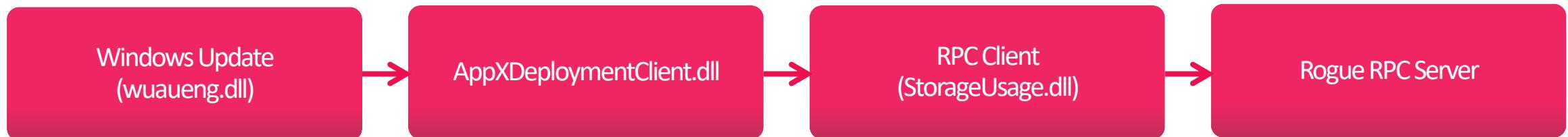
Looking for file system services led to “Storage Service”

---



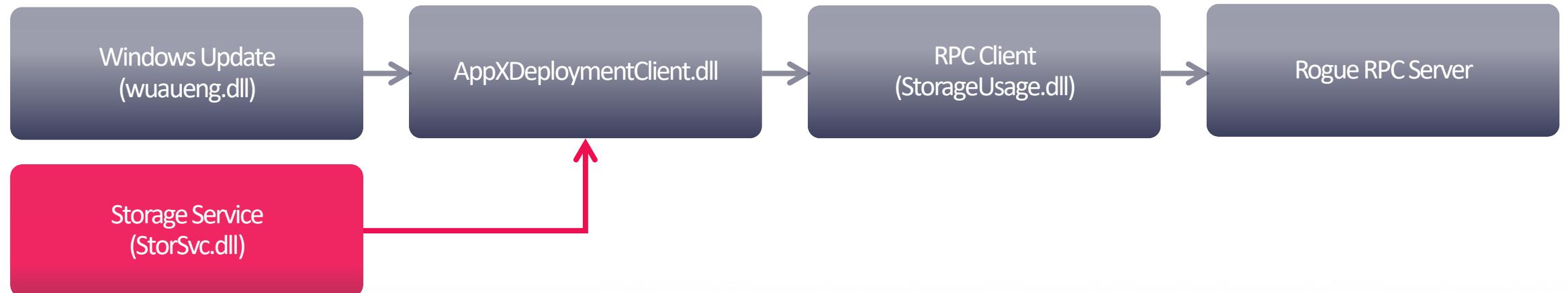
# Manipulating File Access

Registering StorSvc interface resulted in several connections:



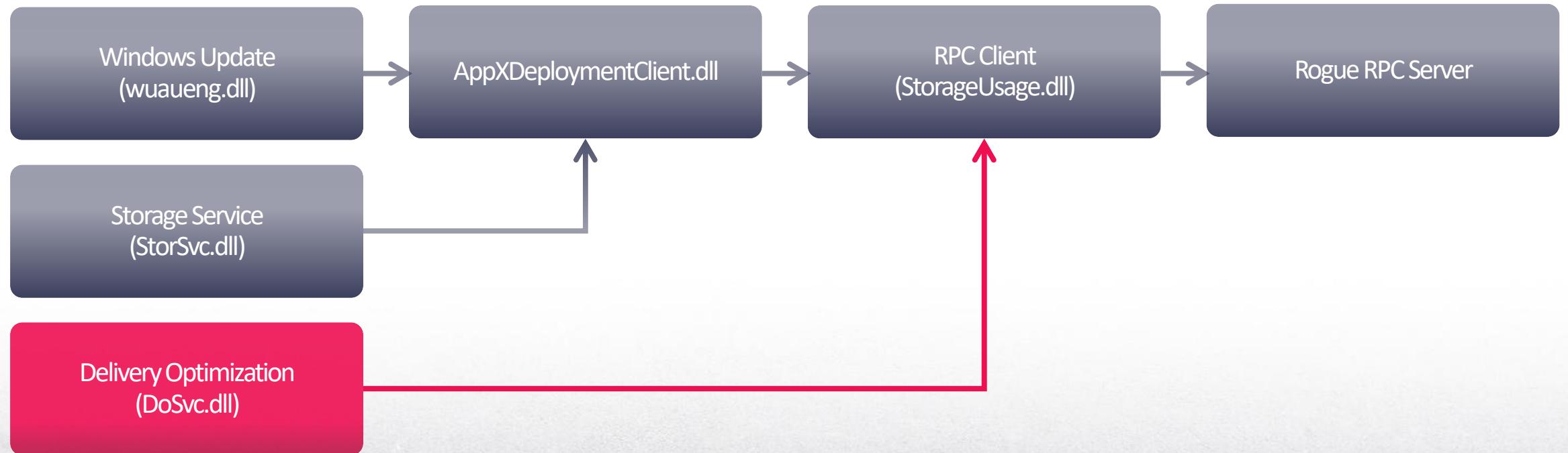
# Manipulating File Access

Registering StorSvc interface resulted in several connections:



# Manipulating File Access

Registering StorSvc interface resulted in several connections:



# Manipulating File Access

The clients invoked 3 undocumented methods

```
long GetStorageInstanceCount([in] short arg1, [out] long* arg2);
long GetStorageSettings([in] short arg1, [in] long arg2, [in] short arg3, [out] long* arg4);
long GetStorageDeviceInfo([in] int arg1, [in] long arg2, [in, out] struct Struct_34_t* arg3);
```

# Manipulating File Access

- Examining the client callstack led to [AppXDeploymentClient.dll](#)
- The public symbols contain definitions of the undocumented methods



# Manipulating File Access

```
typedef struct _STORAGE_DEVICE_INFO
{
    unsigned int Size;
    wchar_t PathName[ 260 ];
    STORAGE_DEVICE_PROPERTIES DeviceProperties;
    STORAGE_PRESENCE_STATE PresenceState;
    STORAGE_DISMOUNT_REASON DismountReason;
    STORAGE_VOLUME_STATUS VolumeStatus;
    STORAGE_FREE_SPACE_STATE FreeSpaceState;
    STORAGE_TEMP_CLEANUP_STATE TempCleanupState;
    GUID StorageId;
    STORAGE_APP_PAIRING_STATUS AppPairingStatus;
    unsigned __int64 ReservedSize;
    wchar_t FriendlyName[ 260 ];
    unsigned int BusType;
    unsigned int FileSystemType;
    unsigned int PersistentVolumeState;
} STORAGE_DEVICE_INFO;
```



# Manipulating File Access

- DoSvc calls `CreateDirectory` with the path returned
- This service is running as “NT AUTHORITY\NETWORK SERVICE”



# Manipulating File Access

Event Properties

Event    Process    Stack

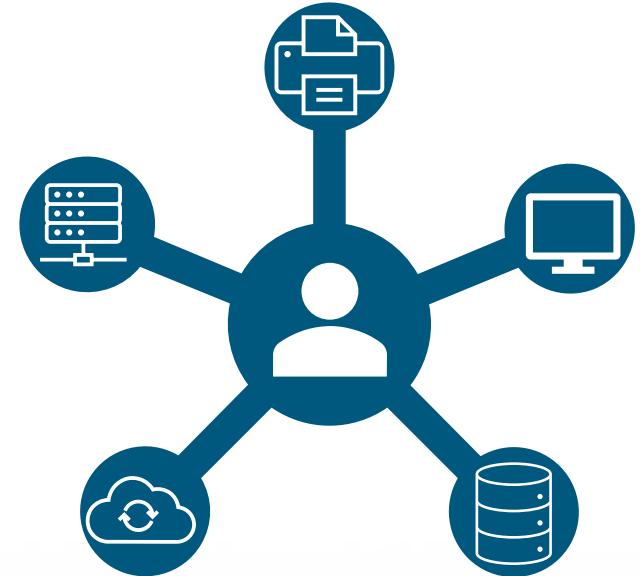
Date: 6/24/2025 6:16:58.0074231 PM  
Thread: 10368  
Class: File System  
Operation: CreateFile  
Result: SUCCESS  
Path: \\ronb-insider\c\$\Windows\ServiceProfiles\NetworkService\Hello DEF CON 33  
Duration: 0.0004970

---

Desired Access: Read Data/List Directory, Synchronize  
Disposition: Create  
Options: Directory, Synchronous IO Non-Alert, Open Reparse Point  
Attributes: N  
ShareMode: Read, Write  
AllocationSize: 0  
OpenResult: Created

# NT AUTHORITY\NETWORK SERVICE

- Fewer capabilities than the local system account
- Authenticates remote resources with the machine account



# Machine Account

- Represented as COMPUTERNAME\$
- Used when the computer accesses remote resources

The screenshot shows the Windows Active Directory Users and Computers (ADUC) interface. The left pane displays a tree view of the directory structure under 'test.domain', with 'Computers' selected. The right pane shows a list of objects, where 'RONB-INSIDER' is selected. A detailed properties window for 'RONB-INSIDER' is open, showing the object type as 'Computer'. The 'Member Of' tab is selected, listing the groups the computer belongs to: 'Active Directory Domain Services Folder' and 'Domain Computers test.domain/Users'.

Name	Type	Description
RONB-INSIDER	Computer	

RONB-INSIDER Properties	
Location	Managed By
General	Operating System
Member Of	Delegation
LAPS	

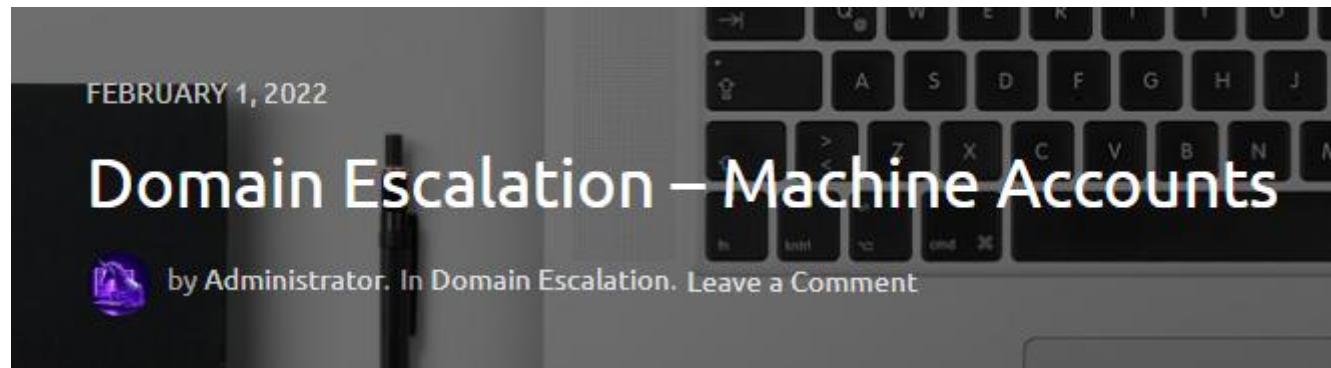
Member of:

Name	
Active Directory Domain Services Folder	
Domain Computers	test.domain/Users

# Machine Account

# Using machine account passwords during an engagement

Posted on 30th October 2017 by Adam Chester



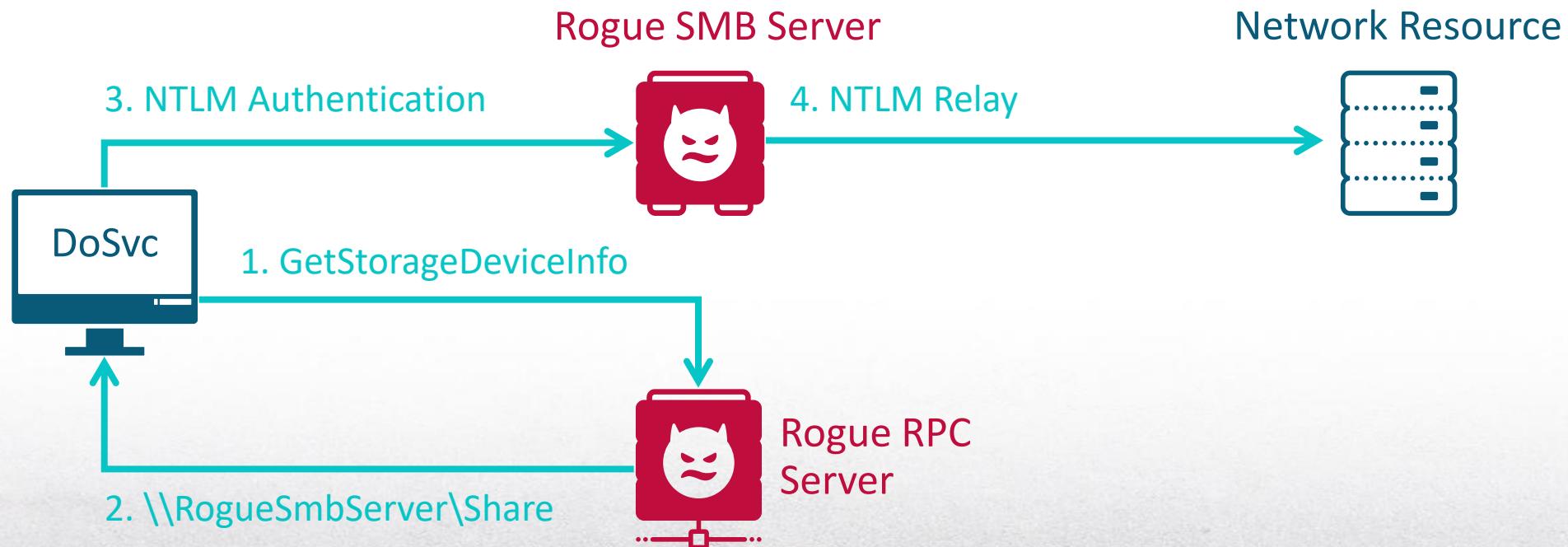
PINNED > ACTIVE DIRECTORY & KERBEROS ABUSE

## Pass the Hash with Machine\$ Accounts

Copy

# Forcing Machine Account Authentication

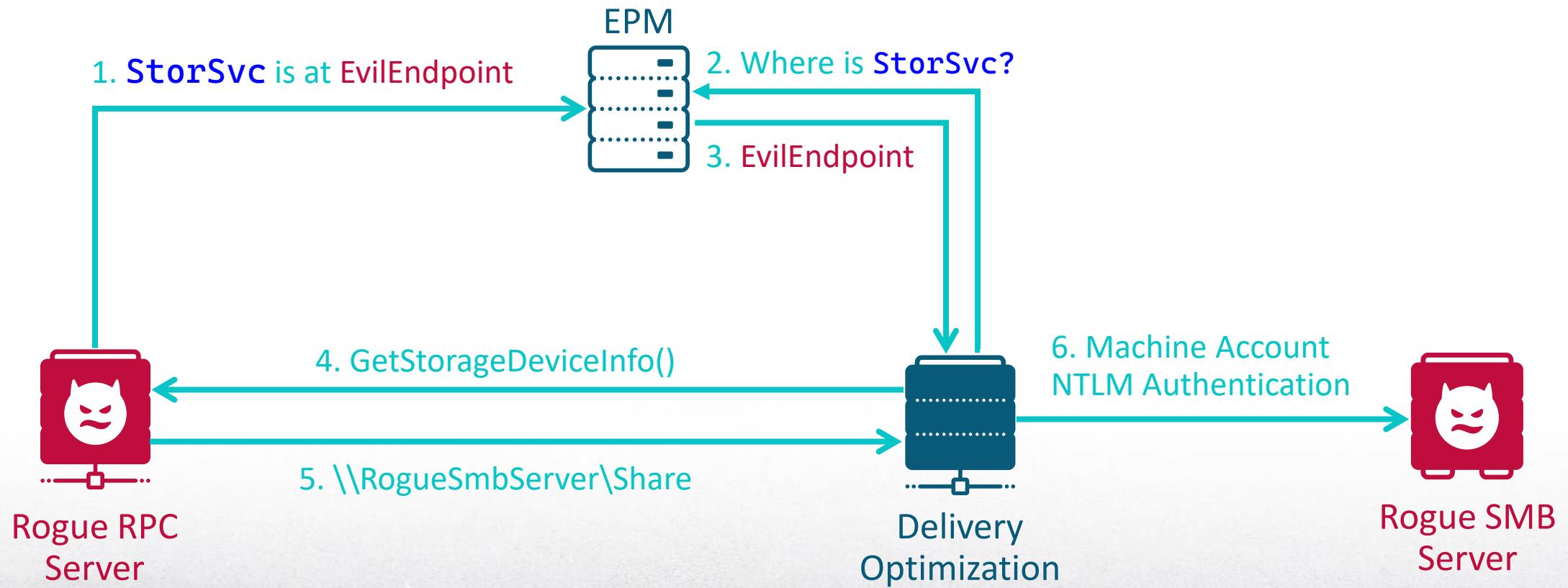
- Returning a network share forces an authentication
- It can be used for NTLM relay



Success!

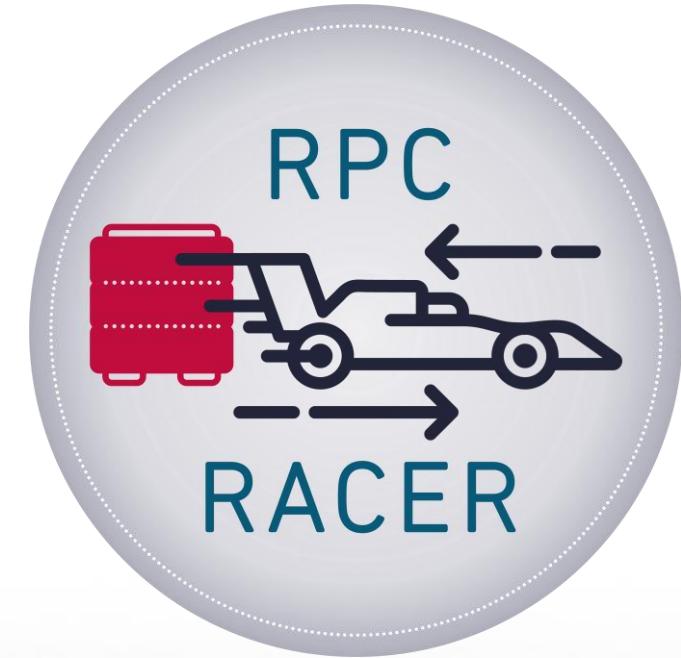


# Attack Flow

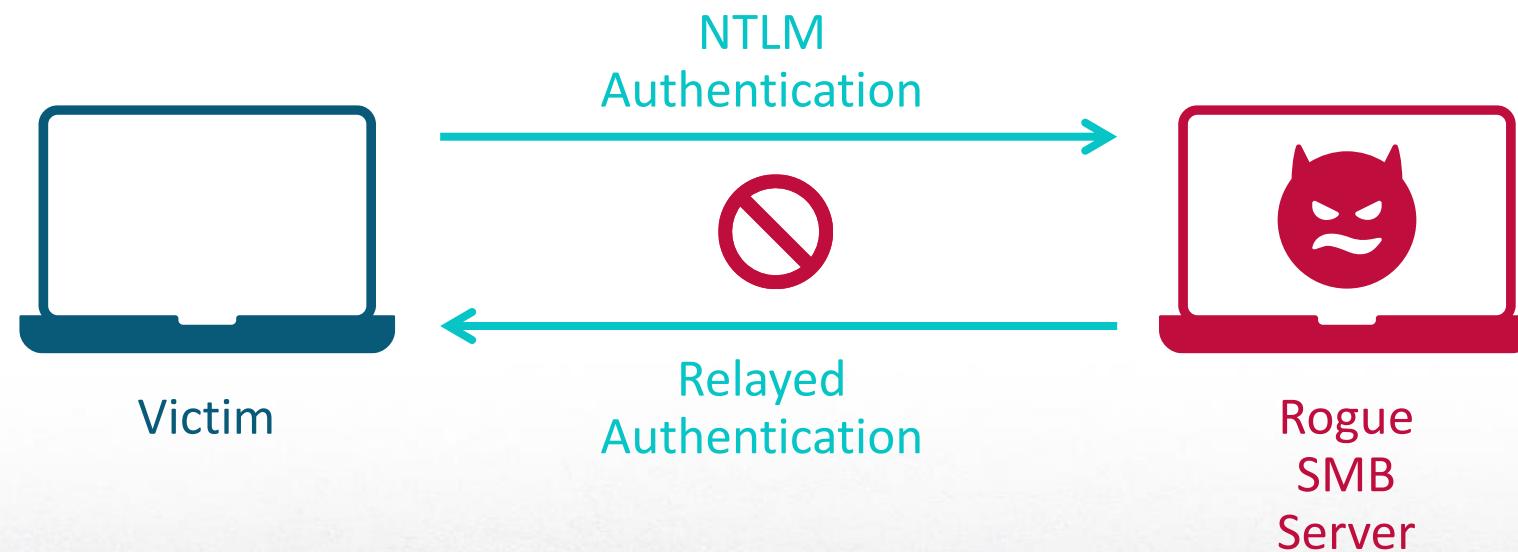


# RPC-Racer

- New tool to launch rogue RPC interfaces
- Designed to be executed on logon
- Forces authentication of the machine account
- Patch for CVE-2025-49760 was released on July 8th, 2025



# Where can we relay the machine account authentication?



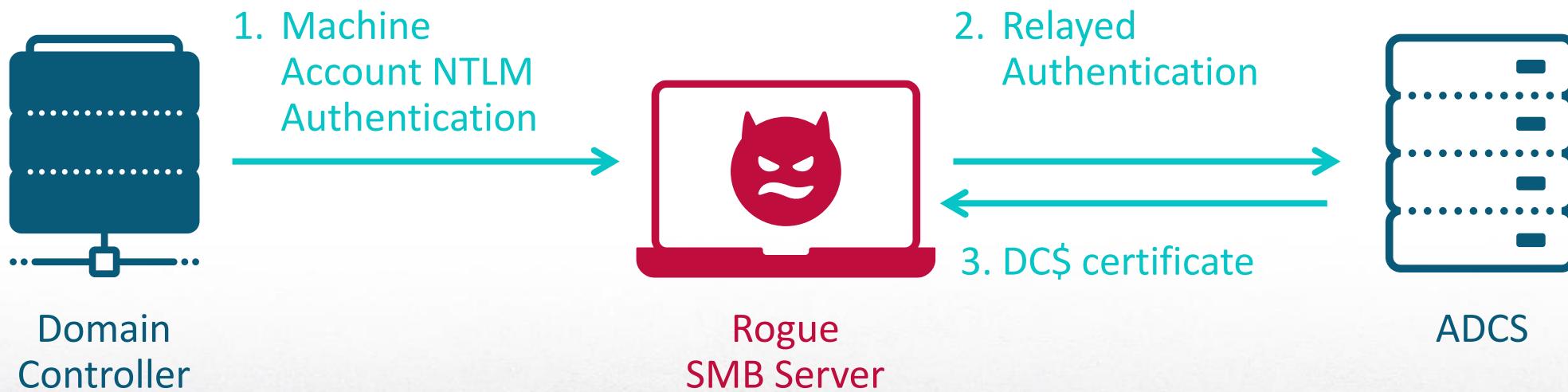
# ESC8

- Targets Active Directory Certificate Service (ADCS)
- ADCS web server can enroll certificates
- Certificates can be used for authentication



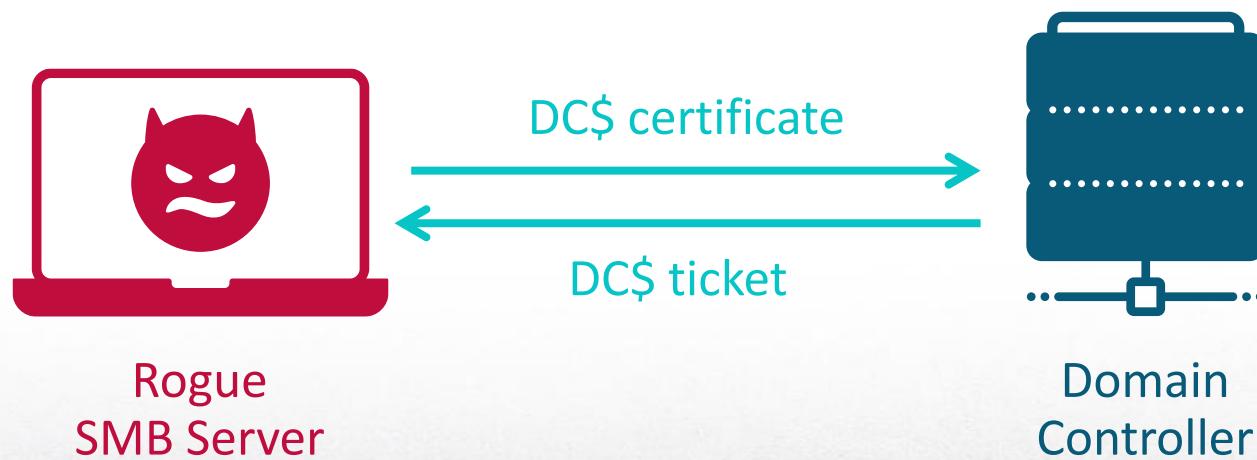
# ESC8

## Step 1: Requesting a certificate for DC\$



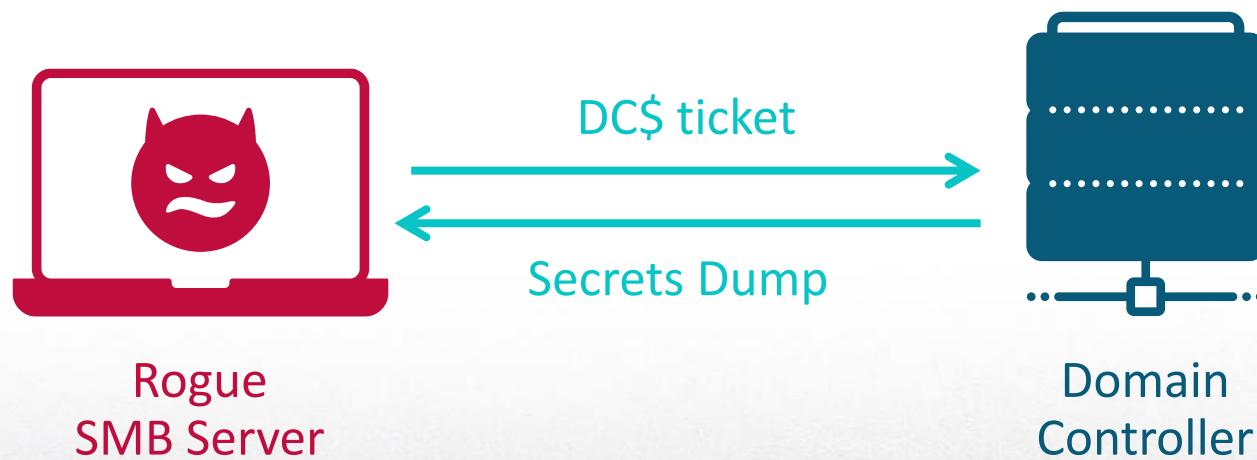
# ESC8

## Step 2: using the certificate to request TGT

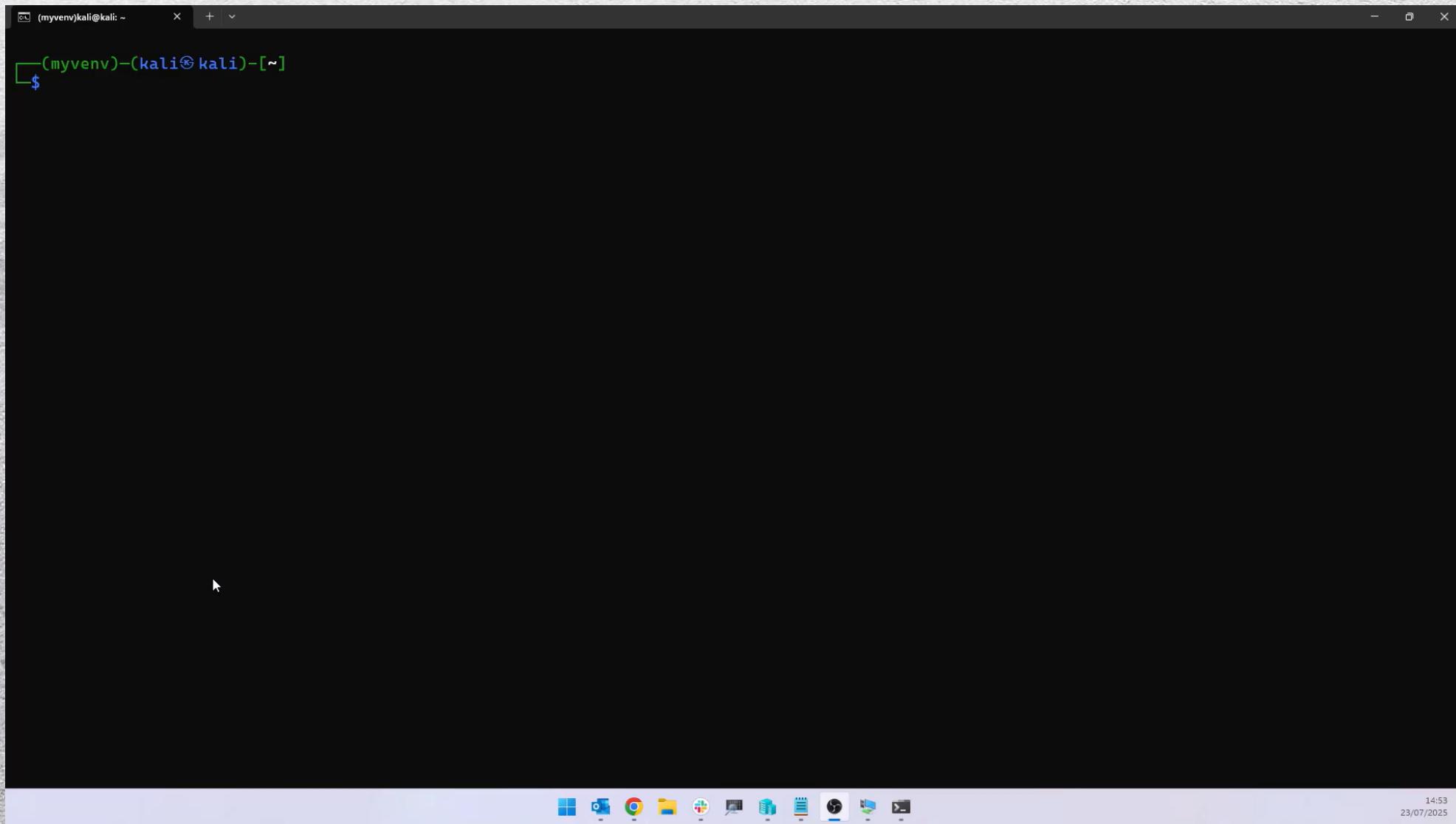


# ESC8

## Step 3: using the TGT to dump password hashes



# RPC-Racer Demo



# Implications



Man in the  
Middle



Denial of  
Service



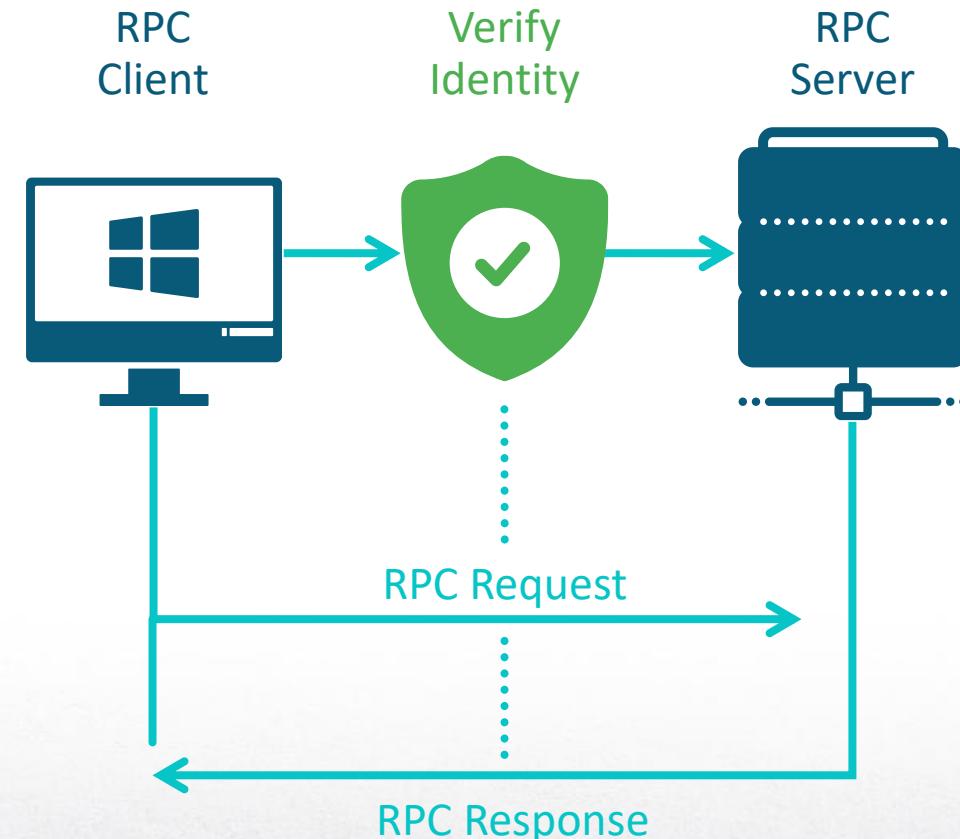
Stealing  
Credentials

# Takeaways

---

The destination address of every protocol should be verified by the source

---



# Takeaways

---

Any stage where untrusted code can be executed should be considered unsafe

---

Services should be launched as early as possible

---



# CVE-2025-49760

StorageUsage.dll  
was patched

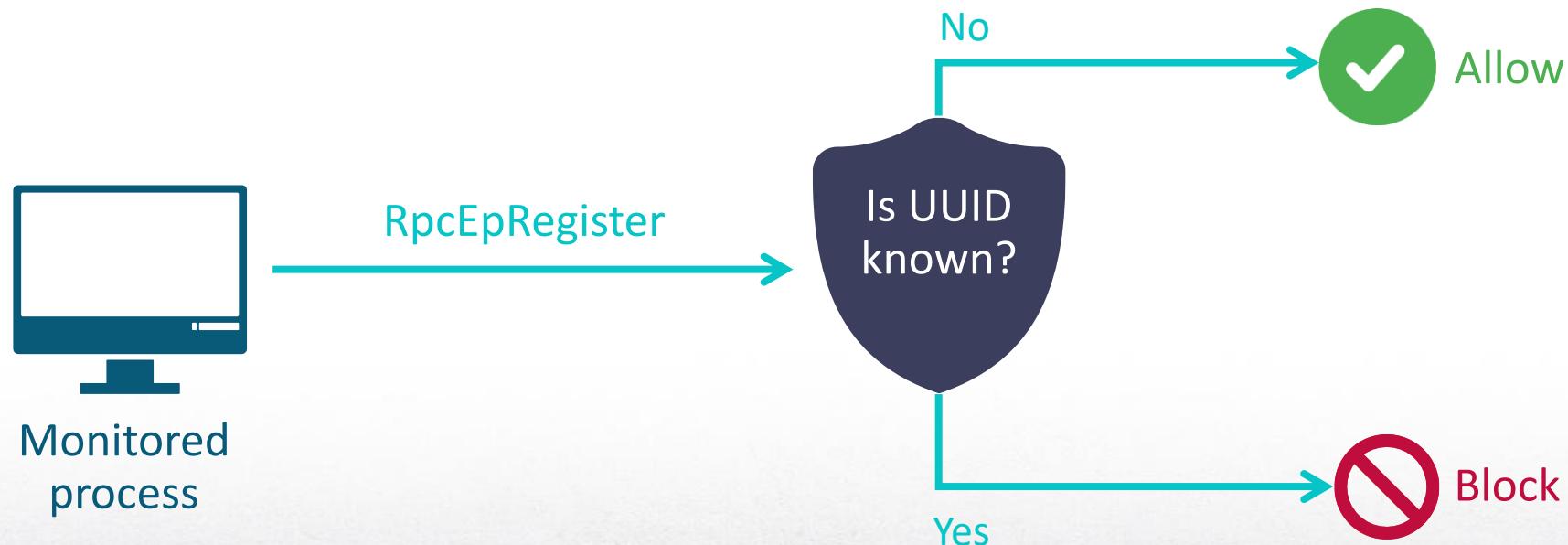
Security QOS applied to  
Binding handle

```
_int64 __fastcall StorageSvcInit(RPC_BINDING_HANDLE *Binding)
{
    // [COLLAPSED LOCAL DECLARATIONS. PRESS NUMPAD "+" TO EXPAND]

    *&pIdentifierAuthority.Value[4] = 0x500;
    String = 0LL;
    *pIdentifierAuthority.Value = 0;
    returnValue = 0;
    pSid = 0LL;
    memset(&SecurityQOS, 0, sizeof(SecurityQOS));
    wil::details::FeatureImpl<_WilFeatureTraits_Feature_3064785210>::__private_IsEnabled(&`wil::FeatureImpl<_WilFeatureTraits_Feature_3064785210>::`__private_IsEnabled)
    if ( v3 )
    {
        if ( !AllocateAndInitializeSid(&pIdentifierAuthority, 1u, 0x12u, 0, 0, 0, 0, 0, 0, 0, &pSid) )
        {
            LastError = GetLastError();
            returnValue = LastError;
            if...
            goto LABEL_5;
        }
        tmpSecQOS.Version = 5;
        *(&tmpSecQOS.AdditionalSecurityInfoType + 1) = 0;
        tmpSecQOS.Sid = pSid;
        tmpSecQOS.Capabilities = 17;
        tmpSecQOS.u.HttpCredentials = 0LL;
    }
}
```

# Detection

Validate registrations to the EPM



# Detection

## Monitor ETW

Event 5, RPC (Microsoft-Windows-RPC)

General Details

Friendly View  XML View

[ **ProcessID** ] 9816 (**Delivery Optimization**)  
[ **ThreadID** ] 6676

**Channel**

**Computer** RONB-INSIDER

**Security**

- **EventData**

**InterfaceUuid** {44d1520b-6133-41f0-8a66-d37305ecc357}  
**ProcNum** 4  
**Protocol** 3  
**NetworkAddress** NULL  
**Endpoint** LRPC-4f19b232ab44a076a2

Event 6, RPC (Microsoft-Windows-RPC)

General Details

Friendly View  XML View

[ **ProcessID** ] 14100 (**RPC-Racer**)  
[ **ThreadID** ] 13212

**Channel**

**Computer** RONB-INSIDER

**Security**

- **EventData**

**InterfaceUuid** {44d1520b-6133-41f0-8a66-d37305ecc357} (**Storage Service**)  
**ProcNum** 4 (**GetStorageDeviceInfo**)  
**Protocol** 3  
**NetworkAddress** NULL  
**Endpoint** LRPC-4f19b232ab44a076a2

# Further Research

## Force DoSvc to use poisoned config files

```
{  
    "KeyValue_EndpointUri": "https://kv801.prod.do.dsp.mp.microsoft.com/",  
    "KeyValue2_EndpointUri": "https://kv801.prod.do.dsp.mp.microsoft.com/",  
    "Discovery_EndpointUri": "https://disc801.prod.do.dsp.mp.microsoft.com/",  
    "Discovery2_EndpointUri": "https://disc801.prod.do.dsp.mp.microsoft.com/",  
    "ContentPolicy_EndpointUri": "https://cp801.prod.do.dsp.mp.microsoft.com/",  
    "ContentPolicy2_EndpointUri": "https://cp801.prod.do.dsp.mp.microsoft.com/",  
    "KeyValue_EndpointFullUri": "https://kv801.prod.do.dsp.mp.microsoft.com/all",  
    "KeyValue2_EndpointFullUri": "https://kv801.prod.do.dsp.mp.microsoft.com/all",  
    "Discovery_EndpointFullUri": "https://disc801.prod.do.dsp.mp.microsoft.com/v2/content/{contentId}",  
    "Discovery2_EndpointFullUri": "https://disc801.prod.do.dsp.mp.microsoft.com/content/{contentId}",  
    "ContentPolicy_EndpointFullUri": "https://cp801.prod.do.dsp.mp.microsoft.com/v3/content",  
    "ContentPolicy2_EndpointFullUri": "https://cp801.prod.do.dsp.mp.microsoft.com/content/{contentId}/contentpolicy",  
    "Geo_EndpointFullUri": "https://geo.prod.do.dsp.mp.microsoft.com/geo",  
    "GeoVersion_EndpointFullUri": "https://geover.prod.do.dsp.mp.microsoft.com/geoversion",  
}
```

C:\Windows\ServiceProfiles\NetworkService\AppData\Local\Microsoft\Windows\DeliveryOptimization\State\keyValueLKG.dat

# Further Research

# Force DoSvc to use poisoned config files

Key name	Value Name	Value Type	Data
RBC	RBC	RegBinary	RBC
C:\Users\ronb\Desktop\researches\rpc-hij...	FileId	RegSz	99e2f6f80f1b15eaff192caff421e2c4c139796d
{d780e00e-1931-0994-a997-d63a58062523}	State	RegDword	5
DeliveryOptimization	Rank	RegSz	0.946397
Jobs	UploadForSecs	RegDword	259200
6466a264-29c8-4e3d-8a46-0e0bb2146e44	IsPinned	RegDword	0
Files_0	ExpireAtTimeFT	RegQword	1390538612670392
b893113f-5e06-4a73-824a-a718b20d20f1	ExpireAtOverride	RegDword	0
Files_0	CdnURL	RegSz	http://au.download.windowsupdate.com/d/msdownload/update/software/upr...
46d10f61-8273-459b-ae11-329001cd76fa	WorkingDir	RegSz	C:\WINDOWS\ServiceProfiles\NetworkService\AppData\Local\Microsoft\Win...
Swarms	PeerId	RegBinary	FC-FC
9ea15831-be9a-454d-a072-3ed97...	SessionId	RegBinary	31-F2-A1-2E-2A-BE-4D-45-A0-73-2E-D0-7C-2D-2A-12

C:\Windows\ServiceProfiles\NetworkService\AppData\Local\Microsoft\Windows\DeliveryOptimization\State\dosvcState.dat

# Further Research

Find additional RPC servers with  
RPC-Recon

Security Center is vulnerable

## Security at a glance

See what's happening with the security and health of your device  
and take any actions needed.



Virus & threat protection

No action needed.



Account protection

No action needed.



Firewall & network protection

No action needed.



App & browser control

No action needed.

# Conclusion

New attack discovered – EPM Poisoning

RPC-Recon is released to map targets

Methods to analyze clients were shown

RPC-Racer is released to force NTLM authentication

# Thank you



@RonB\_Y



[www.linkedin.com/in/ron-by](https://www.linkedin.com/in/ron-by)



[github.com/SafeBreach-Labs/RPC-Racer](https://github.com/SafeBreach-Labs/RPC-Racer)