Detection and Triage of Domain Persistence

Joshua Prager & Nico Shyne

Intro Bio

Joshua Prager

- Principal Consultant - Adversary Detection
- AMU (‘18) & NYU (’24)
- Dad of Two No Limit Soldiers
- Lover of Texas Wine & Whiskey

Nico Shyne

- Consultant - Adversary Detection
- USNA ('17) & UVA ('24)
- Former SWO/IP Officer
- Love movies and live music

SPECTEROPS

BSIDES NYC

Attack Path Overview

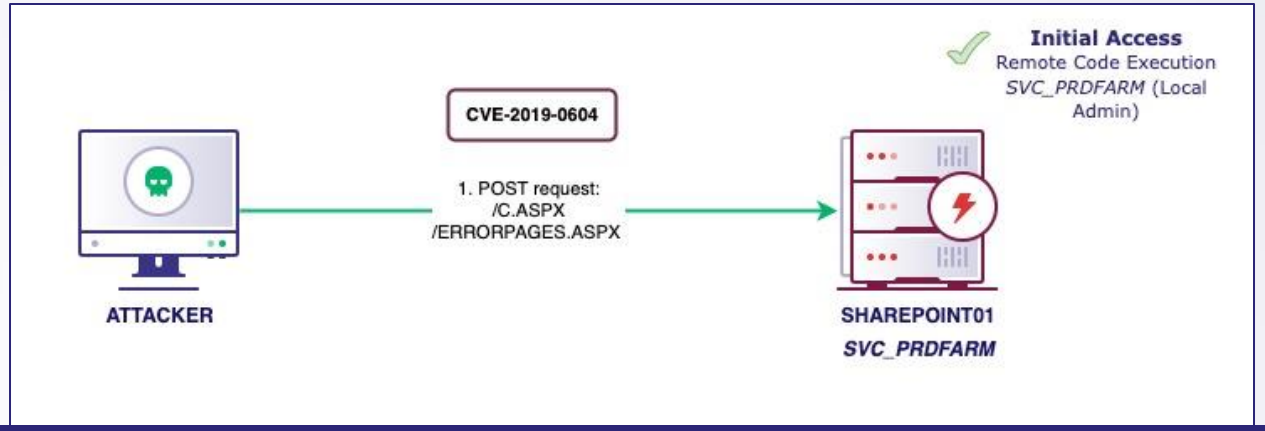
From initial access to elevated domain persistence in a few easy steps…

Attack Path Overview

Initial Access

Initial access was achieved by exploiting RCE vulnerability (CVE-2019-0604) against a publicly accessible Microsoft SharePoint server.

- RCE executed via ASPX files uploaded to SharePoint server
- SharePoint server farm account was compromised
  - Service account with ***local administrative*** access to SharePoint server

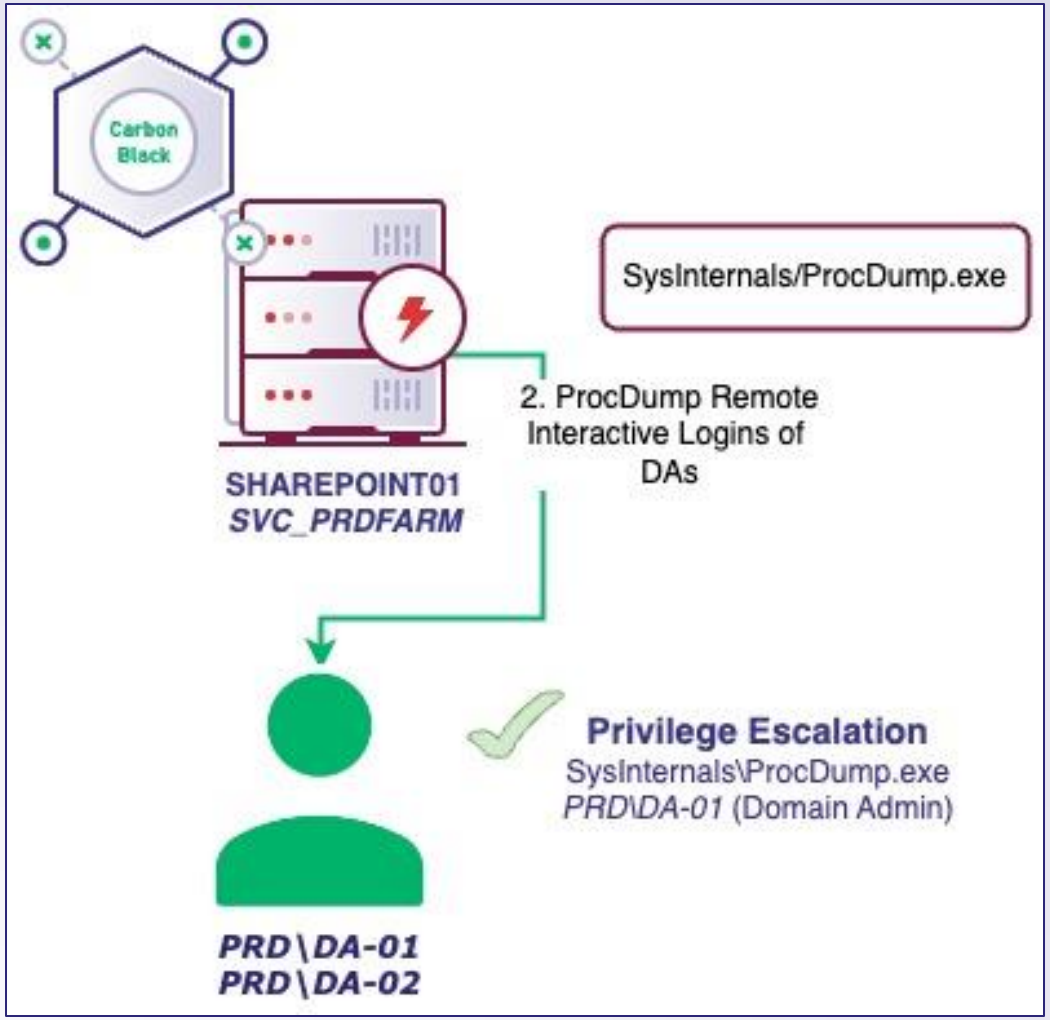


Attack Path Overview

Privilege Escalation

Privilege escalation occurred via the use of administrative tools (SysInternals) located on the file system (*ProcDump.exe*)

- Domain administrators had active remote interactive sessions
  - Credentials are cached during interactive logins
- Carbon Black Application Control was implemented which stopped execution of non-Microsoft signed binaries
  - SysInternals tooling bypasses this control as they are signed by Microsoft

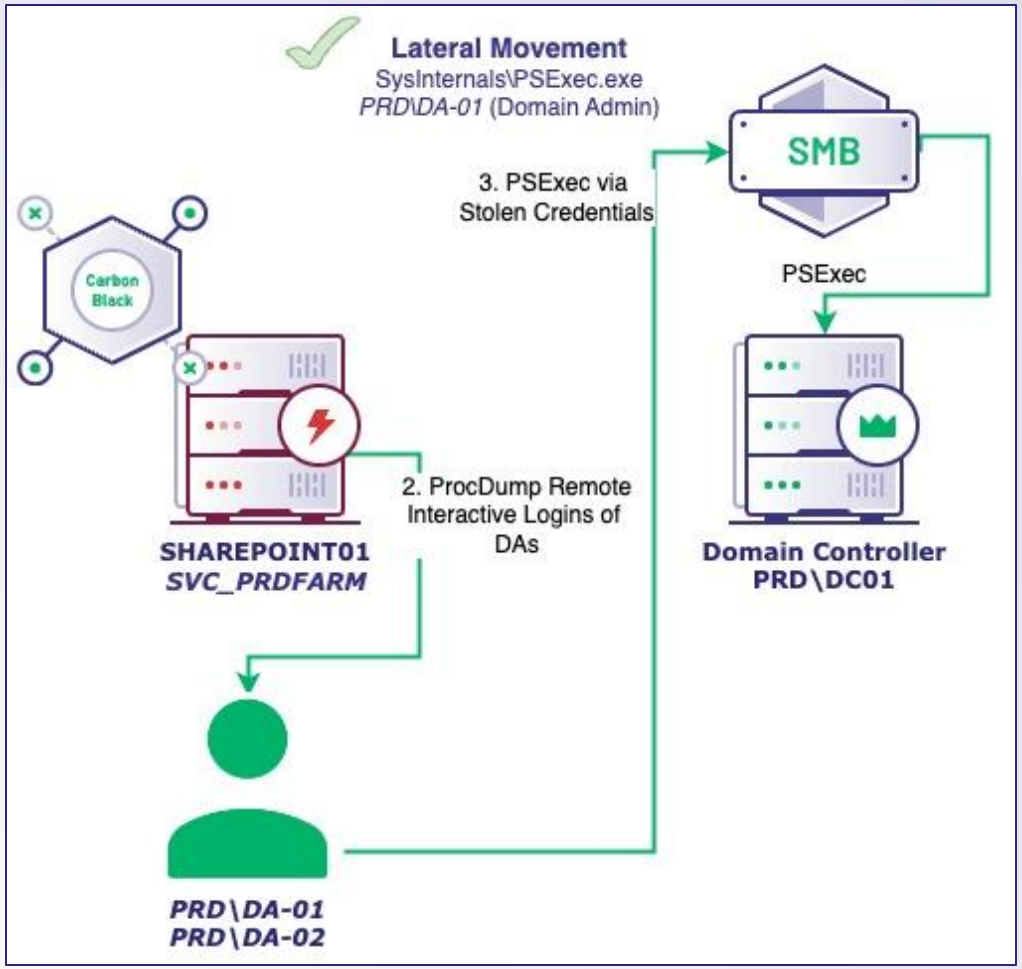


Attack Path Overview

Lateral Movement

Lateral movement between the SharePoint server and the domain controller occurred via PSExec

- Using the compromised domain administrator credentials, the adversaries utilized PSExec to execute commands on the DC
  - PSExec was frequently used for administration, thus blending with baseline behavior

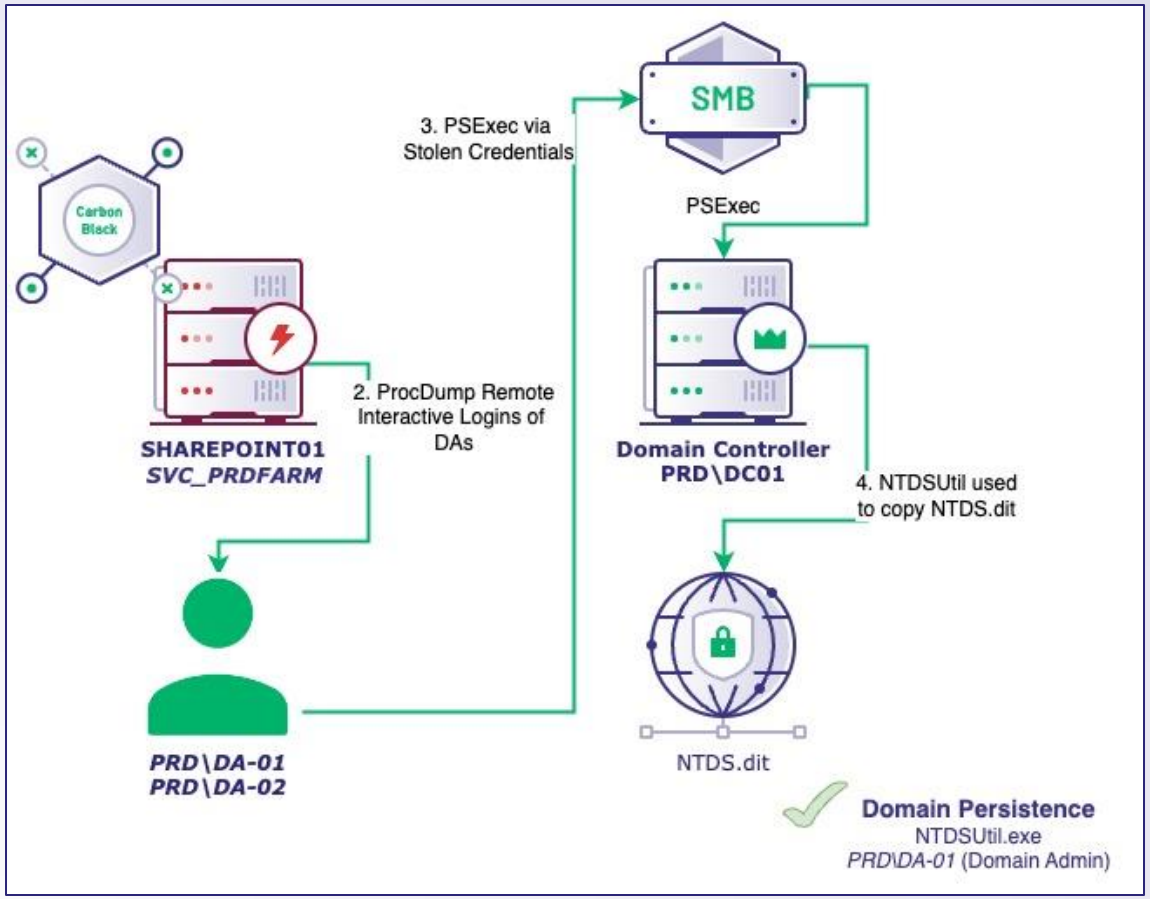


Attack Path Overview

Domain Persistence

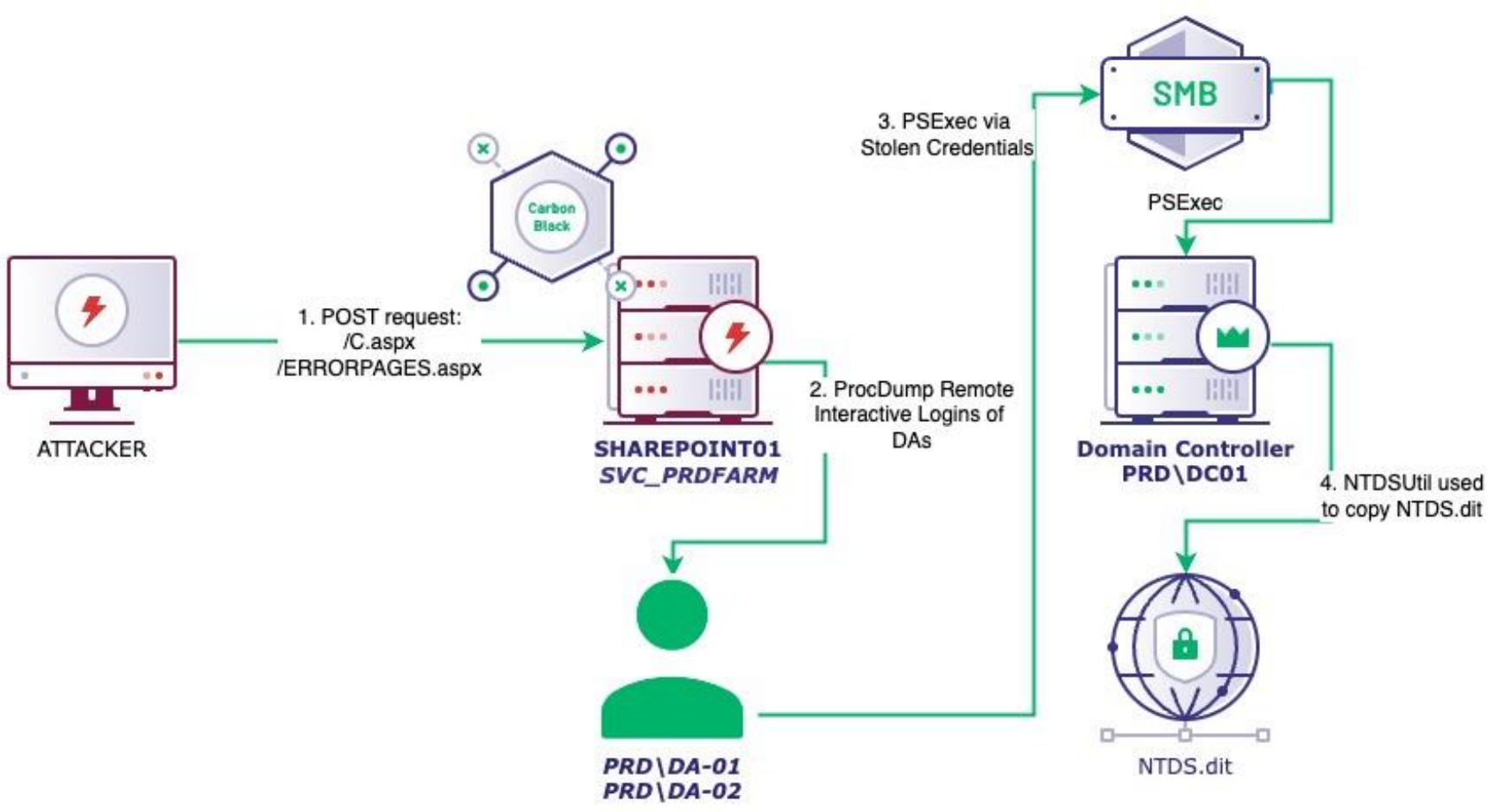
The attacker utilized NTDSUtil upon the domain controller to generate a volume shadow copy of the NTDS.dit

- NTDS.dit = Active Directory Users and Computers database file
- NTDS.dit contains DPAPI Domain Backup Key
  - NTDS.dit is then exfiltrated via C2



Attack Path Overview

Attack Landscape



Defining Domain Persistence

Domain Persistence

- These techniques can be credential theft methods, authentication functionality abuses, or endpoint management abuses

Common Denominators

- Evidence of these techniques usually represent a larger attack path
- The techniques represent the adversary obtained Tier 0 access
- Difficult to scope from an IR pers

Domain Persistence Techniques

- Credential Theft on the Domain Controller via LSASS Memory
- NTDS Access
- DCSync
- Golden Ticket
- Diamond Ticket
- AD CS Certificate Abuse
- SCCM RECON
- SCCM CRED-1
- SCCM TAKEOVER-1
- SCCM EXEC-1

Domain Persistence Techniques

Credential Theft on the Domain Controller via LSASS Memory

This technique can be conducted via many publicly available tools and native Windows binaries (e.g., Task Manager).

The goal of credential theft via LSASS memory is to read the virtual memory space of the LSASS.exe process and retrieve cached credential material.

The typical operational flow:

- Identify the LSASS.exe process (Usually a PID)
- *Open a handle to the LSASS.exe process*
- *Read the LSASS.exe virtual memory space*
- Parse for cached credential material

Domain Persistence Techniques

Credential Theft on the Domain Controller via LSASS Memory

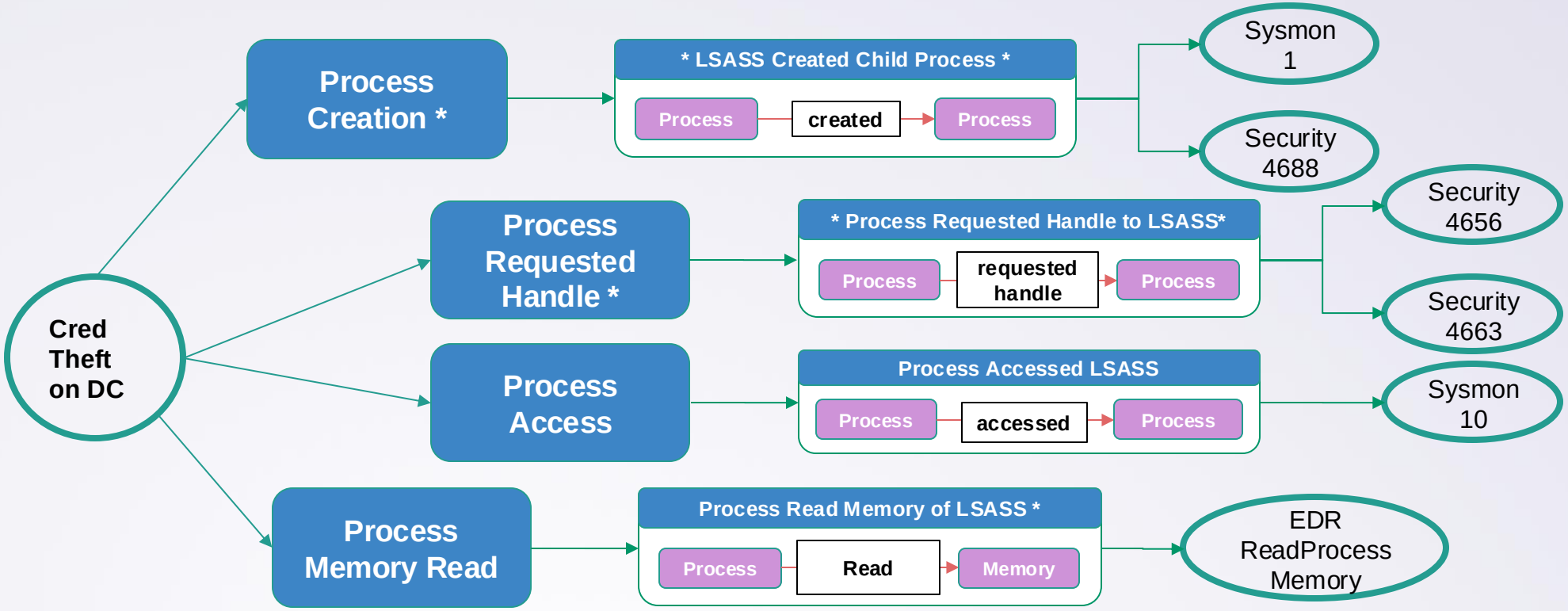
Operationally the Same

- Credential theft via LSASS memory on a domain controller is conducted operationally the same as client credential theft

Key Differences

- Lack of Preventive Controls
  - Generally, No CredGuard
- Availability to Tier 0 Accounts
  - Domain Admin interactive logins

Credential Theft on the Domain Controller via LSASS Memory



Domain Persistence Techniques

NTDS Access

Obtaining the NTDS.dit file of organizations by accessing or copying the database file enables the harvesting of credentials from the organization.

Several native Windows binaries exist for generating backups of the Active Directory database and copying the deadlocked *NTDS.dit* file.

The typical operational flow:

- NTDS backup utility is executed targeting the NTDS.dit file
- Volume Shadow Copy (VSS) service is started
- Backup utility and VSS use the VSS API and the BackupComponents interface to create the snapshot of the NTDS.dit

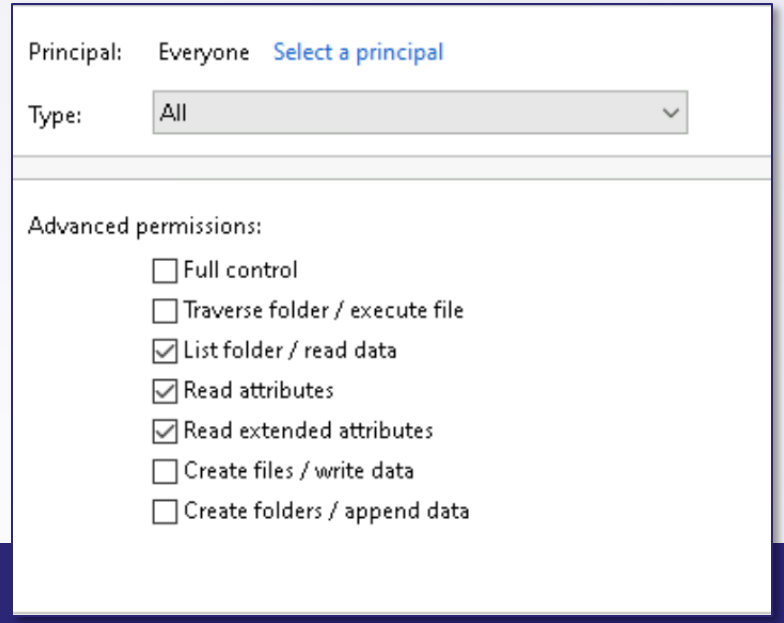
Domain Persistence Techniques

NTDS Access

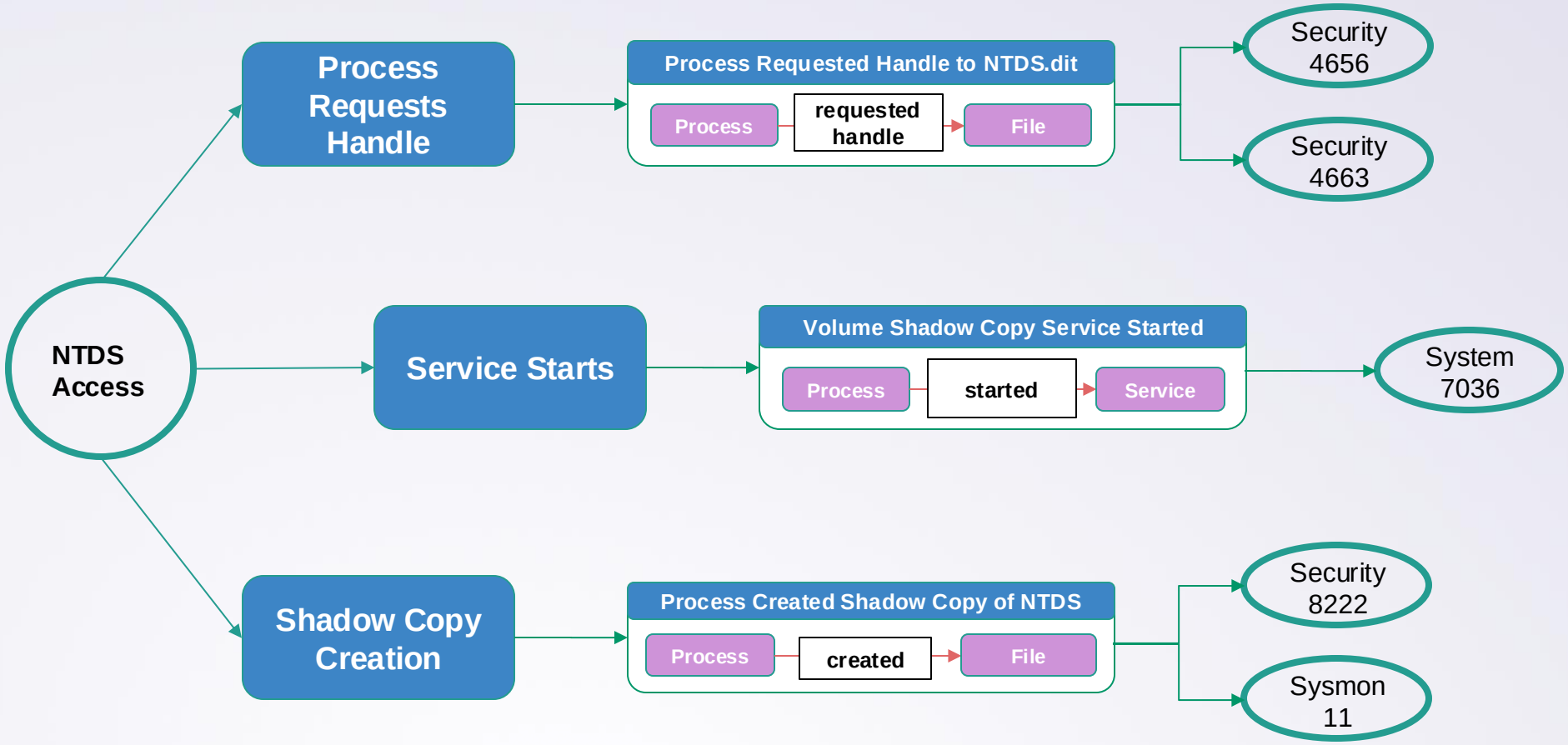
Manipulation of the NTDS.dit file generates several forms of telemetry however this telemetry is not generally enabled by default.

The System Access Control List (SACL) must be enabled to audit the access rights used to read the file.

- Read File Attributes
- Read File Extended Attributes
- Read File Data



NTDS Access


NTDS Access
Process Requests Handle
Process Requested Handle to NTDS.dit
Process
requested handle
File
Security 4656
Security 4663
Service Starts
Volume Shadow Copy Service Started
Process
started
Service
System 7036
Shadow Copy Creation
Process Created Shadow Copy of NTDS
Process
created
File
Security 8222
Sysmon 11

Domain Persistence Techniques

DCSync

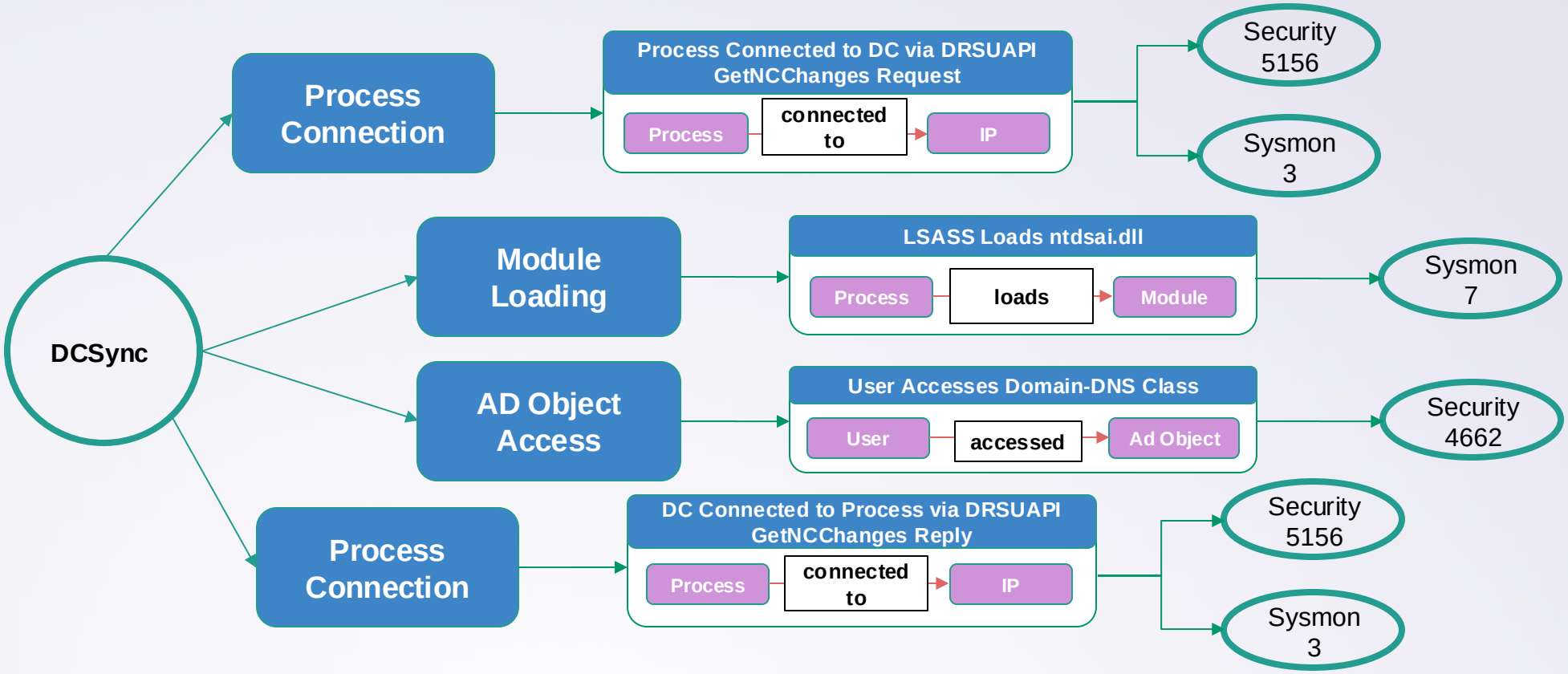
Directory Replication Service uses the MS-DRSR RPC protocol and the **GetNCChanges** RPC method to sync account and organizational container changes across multiple domain controllers.

Syncing credentials is a method by which Tier 0 accounts can be leveraged to retrieve credentials for service accounts related to authentication protocols.

The typical operational flow:

- Compromised client uses RPC method *GetNCChanges Request* to remotely request to sync account information from domain controller
- Domain controller's LSASS process loads *ntdsai/ntdsapi(.dll)* to utilize DRSUAPI RPC interface to access NTDS.dit
- Domain controller remotely syncs the credentials to compromised client via RPC method *GetNCChange Reply*

DCSync


DCSync
Process Connection
Process Connected to DC via DRSUAPI GetNCChanges Request
Process
connected to
IP
Security 5156
Sysmon 3
Module Loading
LSASS Loads ntdsai.dll
Process
loads
Module
Sysmon 7
AD Object Access
User Accesses Domain-DNS Class
User
accessed
Ad Object
Security 4662
Process Connection
DC Connected to Process via DRSUAPI GetNCChanges Reply
Process
connected to
IP
Security 5156
Sysmon 3

Domain Persistence Techniques

Golden Ticket

The *KRBTGT* account generates a key (a hash of its account password) and the KDC uses this key to sign and encrypt TGTs. Because Kerberos inherently trusts any TGT encrypted with that *KRBTGT* account hash, an adversary with access to that hash could generate their own TGT (a *golden* TGT) and bypass the KDC entirely.

The typical operational flow:

- An adversary requires the FQDN of the domain, the SID of the domain, an account to impersonate, and a *KRBTGT* password hash
- The adversary passes this data to a new ticket (using a tool like mimikatz or Rubeus), and that ticket can be saved in the current session's ticket cache
- This new ticket gives access to wherever the *KRBTGT* account has access within that domain

Domain Persistence Techniques

Golden Ticket

Data Source:

- Event ID 4768 (TGT Requested)
- Event ID 4769 (Kerberos Service Ticket Requested)
- Event ID 4627 (Group Membership Information)
- Klist

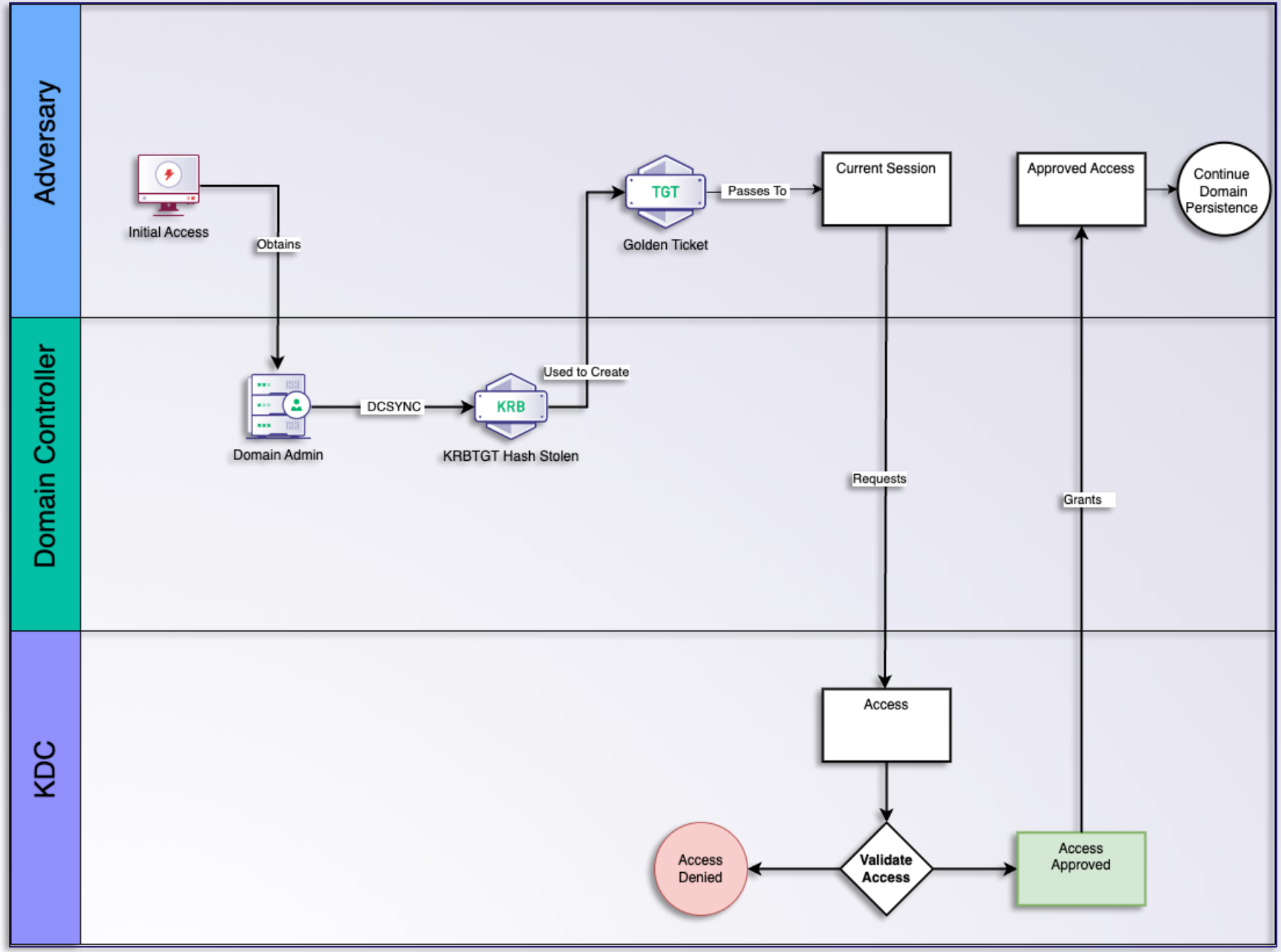
Detection Strategy

- **Focus on KRBTGT Password Hash**: Detection efforts should concentrate on identifying the theft of the KRBTGT password hash and the anomalous use of the KRBTGT account, rather than solely on the ticket requests (Event IDs 4768 and 4769), which appear identical for both legitimate and Golden Ticket attacks.
- **Monitor Group Membership Changes**: Utilize Windows Security event ID 4627 to track changes in group memberships, particularly for signs of unauthorized elevation to privileged groups like Domain Admins, which could indicate a Golden Ticket attack.
- **Track Unmatched TGS-REQs**: Look for TGS-REQs (Event ID: 4769) without a corresponding AS-REQ (Event ID: 4768) and tickets that do not display proper FQDNs, as these may suggest the use of forged tickets.
- **Use klist for Validation**: To confirm suspected Golden Ticket activities, employ the klist command to review the Kerberos ticket cache following unusual logon events (Event ID: 4624), indicating the importation of a stolen KRBTGT ticket.

Golden Ticket

Required Items:

- **Domain FQDN**
- **Domain SID**
- **Account to impersonate**
- ***KRBTGT* password hash**



Domain Persistence Techniques

Diamond Ticket

Instead of creating their own TGT (as with Golden Tickets), adversaries could instead opt to modify a legitimately issued TGT that has already been issued by the KDC.

The typical operational flow:

- Obtain *KRBTGT* password hash
- Request a legitimate TGT
- Decrypt legitimate TGT
- Modify TGT Privilege Attribute Certificate (PAC)
- Re-encrypt TGT

Domain Persistence Techniques

Diamond Ticket

Data Source:

- Event ID 4768 (Kerberos Authentication Ticket Requested)
- Event ID 4648 (A Logon Was Attempted Using Explicit Credentials)
- Event ID 4672 (Special Privileges Assigned to New Logon)
- Anomalous Access Patterns
- Kerberos Ticket Lifetimes

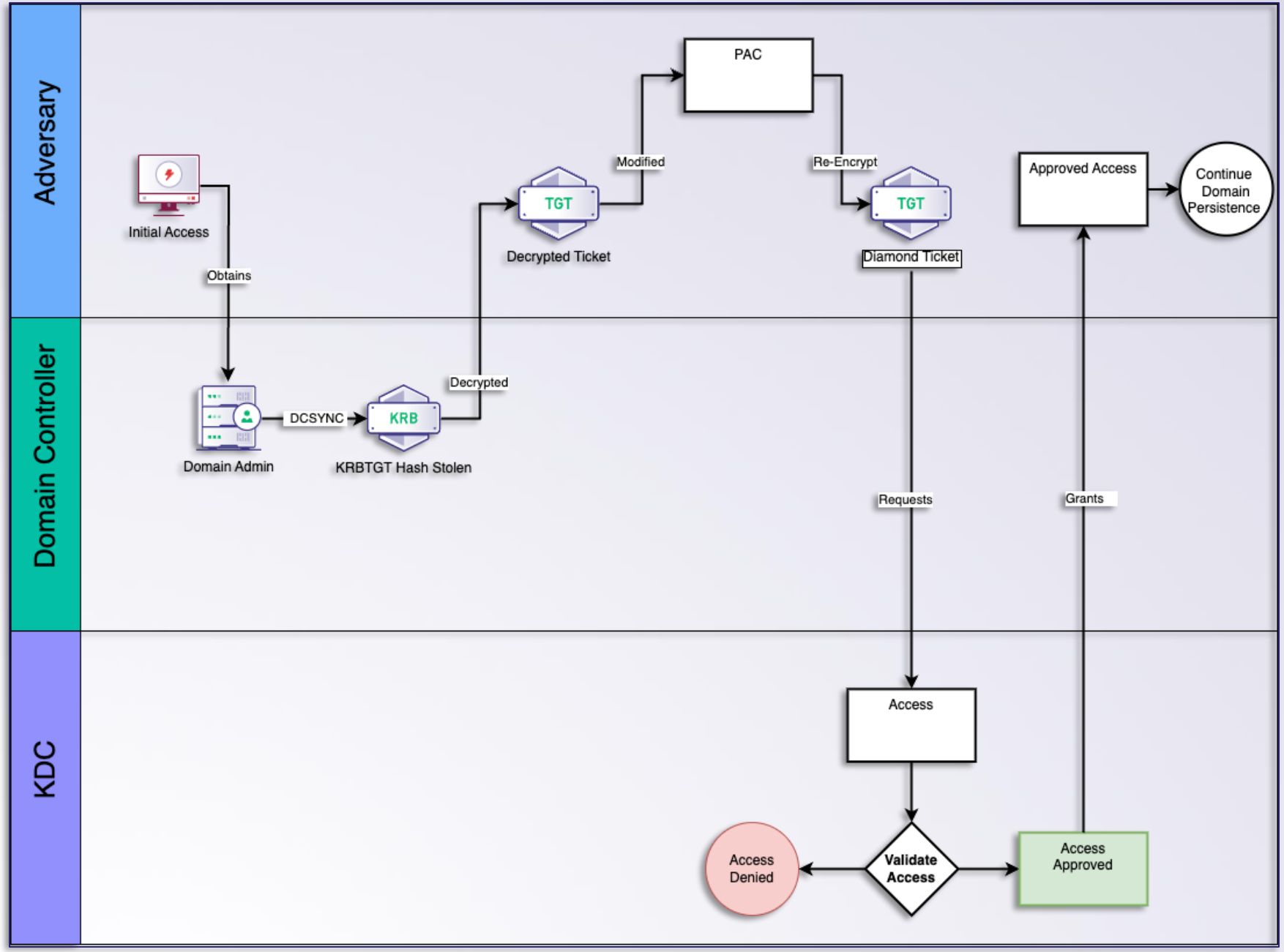
Detection Strategy

- **Monitor KRBTGT Password Hash Theft:** Use detection strategies like lsadump, NTDS.dit access, and DCSync to identify unauthorized access to the KRBTGT account's password hash.
- **Detect Anomalies in Group Membership Changes:** Watch for unexpected changes, such as low-privilege users gaining high-privilege group memberships (e.g., Domain Admins) without corresponding administrative actions.
- **Analyze Kerberos Ticket Requests (Event ID 4768) and Modifications (Event ID 4648):** Look for anomalies in ticket requests and modifications, especially where the PAC of a legitimately issued TGT is altered.
- **Track Anomalies in AS-REQs:** Identify discrepancies in AS-REQs, particularly where the PA-PAC-REQUEST is set to false, indicating potential manipulation of authentication tickets.
- **Employ Additional Validation Techniques:** Use tools like klist or ACE: Get-KerberosTicketCache for targeted investigation and validation of suspicious Kerberos ticket operations.

Diamond Ticket

Required Items:

- **Domain FQDN**
- **Domain SID**
- **Account to impersonate**
- **KRBTGT password hash**



Domain Persistence Techniques

AD CS Certificate Misuse

- **Active/Passive Certificate Theft**
- UnPAC the Hash
- **Golden Certificate**
- **ESC1: Enrollee Supplies Subject**
- **ESC2: Not viable for PKINIT/LDAP**
- **ESC3: Misconfigured Enrollment Agent**
- **ESC4: Modify Certificate Template**
- **ESC5: Modify AD PKI Objects**
- **ESC6: AttributeSubjectAltName2**
- **ESC7: CA Admin/Cert Manager**
- ESC8: Relay to HTTP Enrollment
- ESC9: Implicit Binding Abuse
  - PKINIT
- ESC10: Implicit Binding Abuse
  - LDAP/Schannel
- ESC11: Relay to RPC Enrollment
- ESC12: YubiHSM Specific
- ESC13: OID Group Link
- ESC14: Explicit Binding Abuse
- **Key Trust/Shadow Credentials**
- NTLM Relay to Shadow Credentials

Domain Persistence Techniques

AD CS Certificate Misuse

BLOODHOUND ENTERPRISE | ATTACK PATHS | EXPLORE | POSTURE | GROUP MANAGEMENT

[redacted]

Status: Idle Last Analysis: 2024-10-07 17:04 CDT (GMT-0500)

1.7k
1.5k
1.5k
1.5k
1.5k
1.5k
1.5k
1.5k
1.5k
1.5k
1.5k
1.5k
1.5k

ADCSESC4
GoldenCert
ADCSESC1
HasSession
GenericWrite
Owns
WriteOwner
WriteDacl
ADCSESC3
GenericAll
WriteGPLink
AllExtendedRights
WriteAccountRestrictions
SyncLAPSPassword
AddMember

CORP
Tier Z
3%

Large Default Groups with Generic All Privileges 3 Findings ± 0 vs last month High

Description

The "Generic All" privilege grants principals the ability to perform nearly any action against the object, including abusable actions such as resetting user passwords and adding new users to security groups. Generic All is also known as Full Control.

Large default groups within Active Directory (`DOMAIN USERS`, `DOMAIN COMPUTERS`, `AUTHENTICATED USERS`, `EVERYONE`, `USERS`) should arguably never be granted this privilege, as this is against best practice for least privilege. When an adversary discovers an object where most or all domain authenticated users have this privilege, such objects can provide the attacker with their first step in a critical attack path leading to the compromise of the entire enterprise.

3 Findings Timeline

Accepted

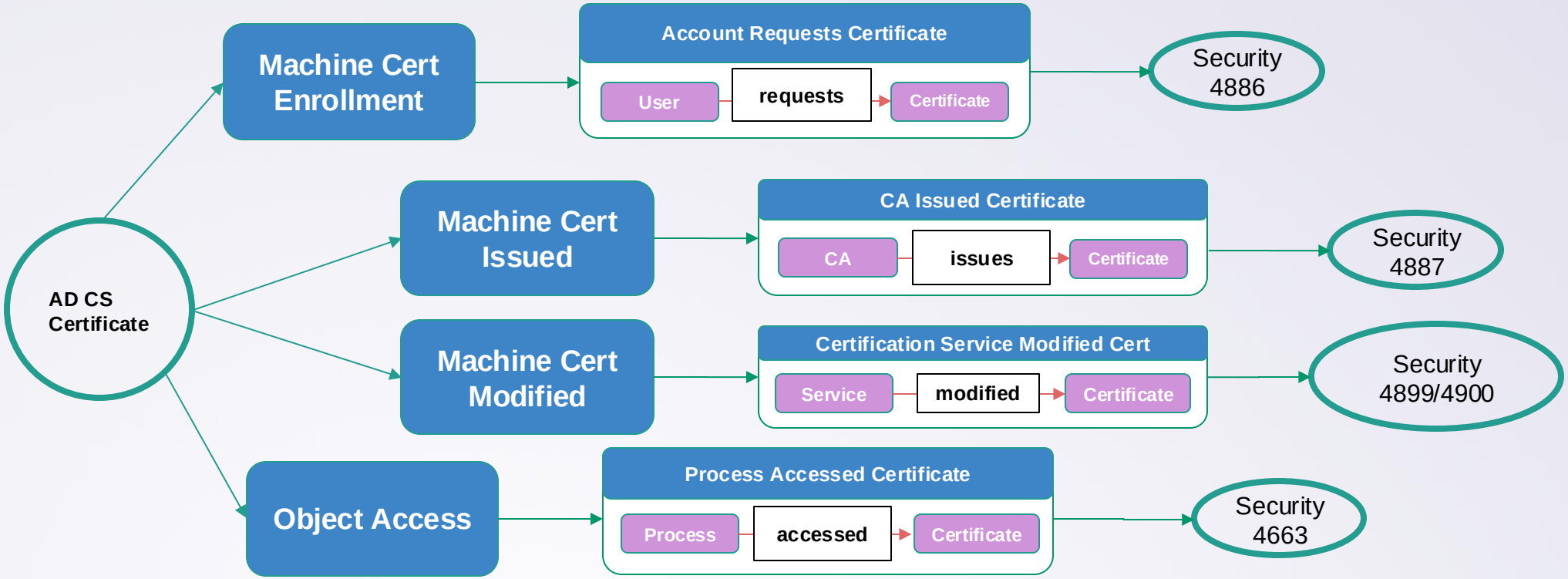
Group	Principal	First Seen
AUTHENTICATED USERS[redacted]	CITRIX_SMARTCARDLOGON@[redacted]	2024-10-07
AUTHENTICATED USERS[redacted]	KEYRECOVERYAGENT@[redacted]	2024-10-07

REMEDIATION

Remove the Generic All privilege held by the large default group against the Active Directory object.

View / Export Full Remediation Plan Export CSV

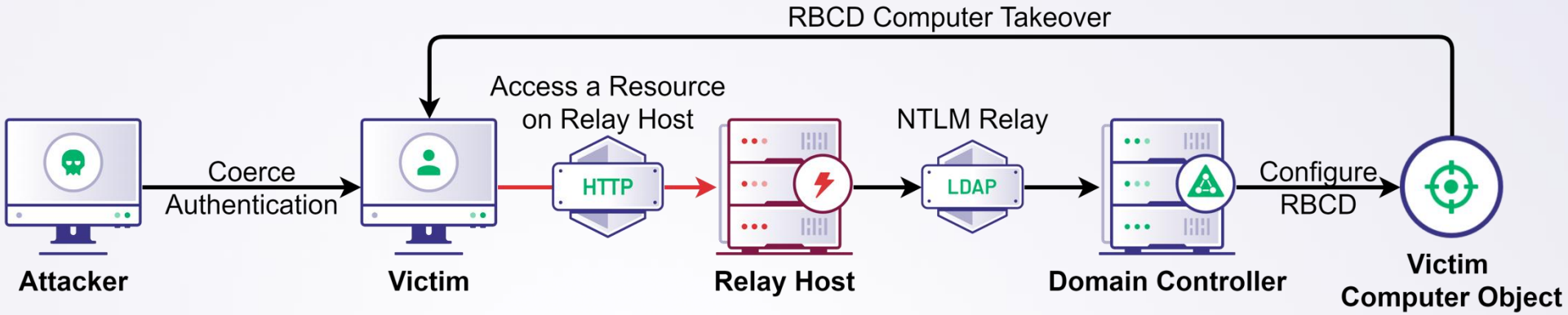
AD CS Certificate Misuse



Domain Persistence Techniques

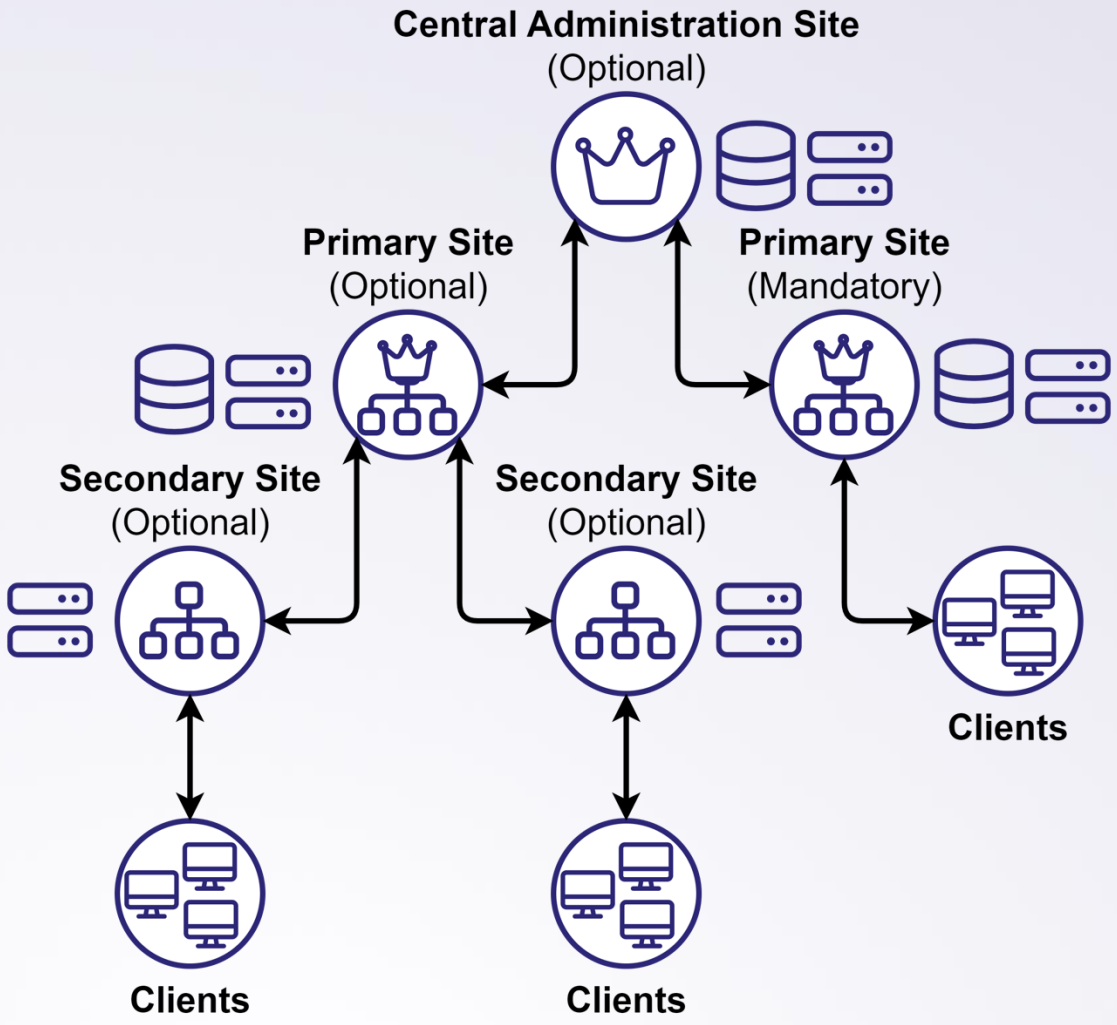
SCCM

- Why do we care about SCCM?
  - Huge amount of technical debt
  - Widespread adoption
  - Legacy implementations are not usually implemented with InTune
  - Telemetry is difficult to find since there hasn't been a centralized repository for SCCM based preventions/detections (or is there??)



Domain Persistence Techniques

SCCM



Misconfiguration Manager Taxonomy

Because "Hierarchy takeover via NTLM coercion and relay to MSSQL on remote site database" does not roll off the tongue…

Attack Techniques

RECON

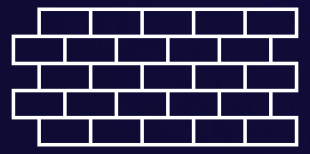
CRED

ELEVATE

TAKEOVER

EXEC

Defense Techniques



PREVENT

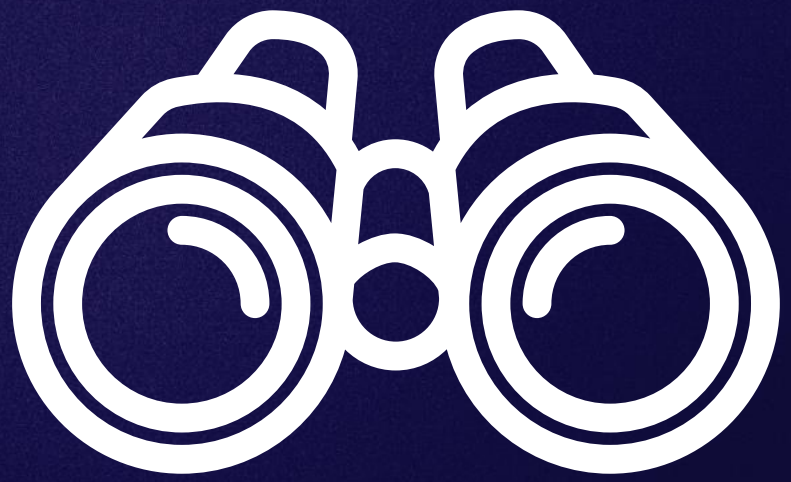
DETECT

CANARY

RECON

Purpose

- Methods for identifying SCCM systems
- Uses LDAP/SMB/ HTTP to enumerate possible systems of interest



Existing Cases

- Enumerating SCCM assets
- Using existing tools like SMS Provider to find interesting users

https://posts.specterops.io/misconfiguration-manager-overlooked-and-overprivileged-70983b8f350d

CRED

Purpose

- Identify privileged credentials
- Leads to direct hierarchy takeover or domain compromise

Existing Cases

- Extracting credentials
- Leads to lateral movement

https://posts.specterops.io/misconfiguration-manager-overlooked-and-overprivileged-70983b8f350d

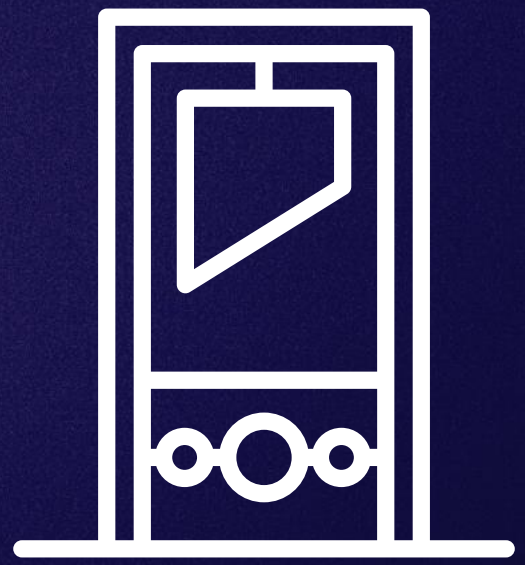
EXEC

Purpose

- Various steps required to deploy an application
- Facilitates lateral movement
- Use of secondary tools like Powershell/WMI Providers

Existing Cases

- Deploying an application whose installation path is a UNC path that we control



https://posts.specterops.io/misconfiguration-manager-overlooked-and-overprivileged-70983b8f350d

TAKEOVER

Purpose

- Various steps required to take over a SCCM hierarchy
- Leads to complete domain takeover

Existing Cases

- NTLM coercion
- Relay to MSSQL/SMB/Endpoint of choice

https://posts.specterops.io/misconfiguration-manager-overlooked-and-overprivileged-70983b8f350d

Domain Persistence Techniques

SCCM CRED-1

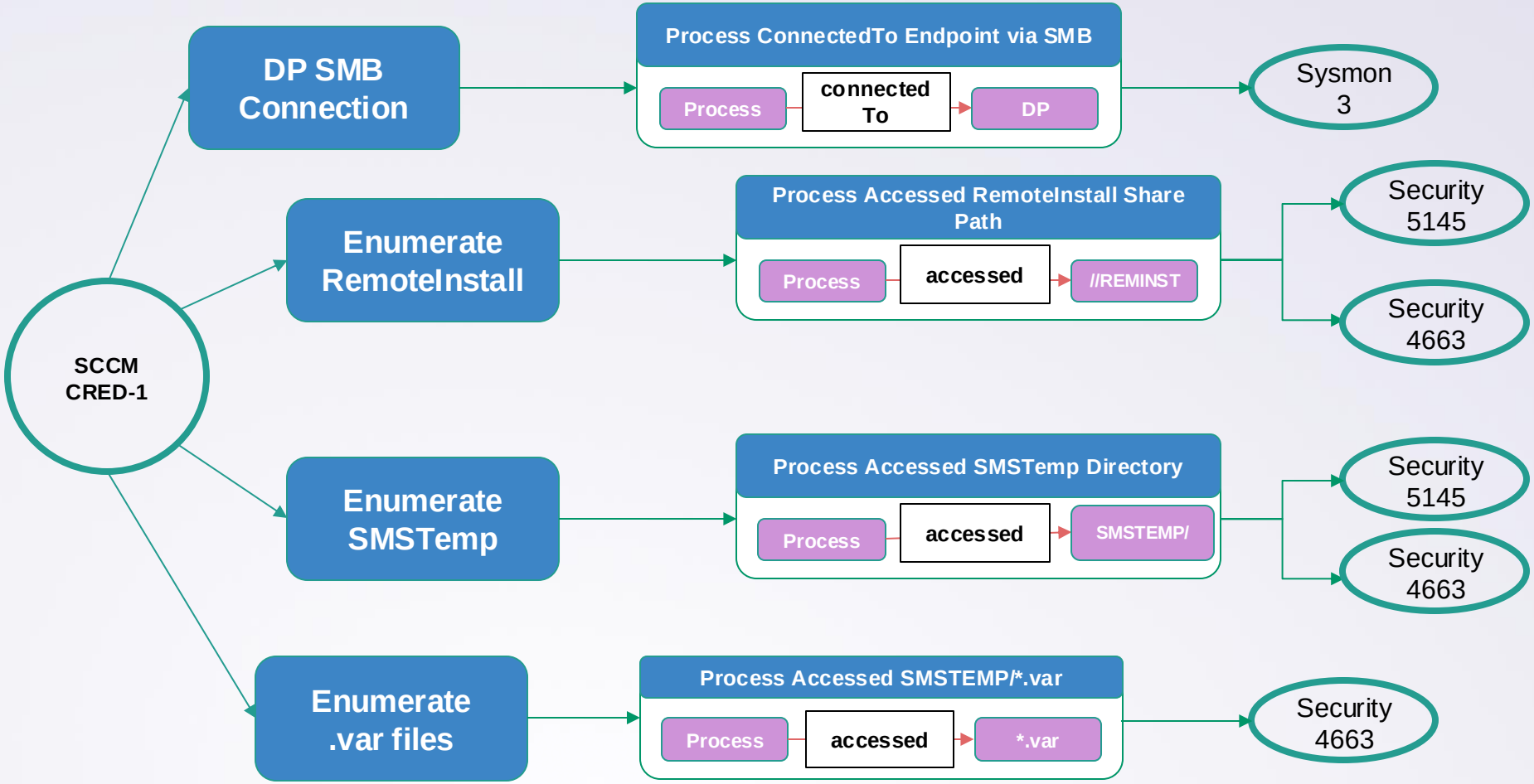
SCCM contains a preboot execution environment (PXE) feature which allows systems to load a specific operating system image on boot.

Attackers can recover domain credentials from PXE media if weak passwords are used, potentially transitioning from an unauthenticated network context to a domain-authenticated one, allowing for privilege escalation and lateral movement.

The typical operational flow:

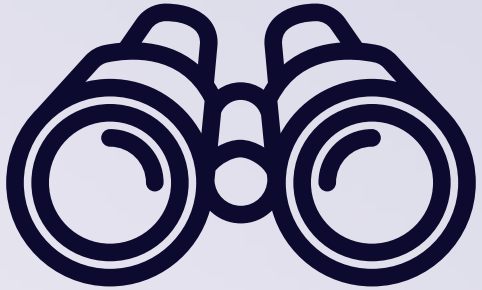
- Connect to Distribution Point via SMB
- Enumerate “REMINST” (Remote Install) share (Windows Deployment Services (WDS) and often contains PXE boot files)
- Enumerate SMSTemp directory
- Spider .var extension, which likely contain PXE boot configuration variables

SCCM CRED-1


SCCM CRED-1
DP SMB Connection
Process ConnectedTo Endpoint via SMB
Process
connected To
DP
Sysmon 3
Enumerate RemoteInstall
Process Accessed RemoteInstall Share Path
Process
accessed
//REMINST
Security 5145
Security 4663
Enumerate SMSTemp
Process Accessed SMSTemp Directory
Process
accessed
SMSTEMP/
Security 5145
Security 4663
Enumerate .var files
Process Accessed SMSTEMP/*.var
Process
accessed
*.var
Security 4663

Domain Persistence Techniques

SCCM RECON



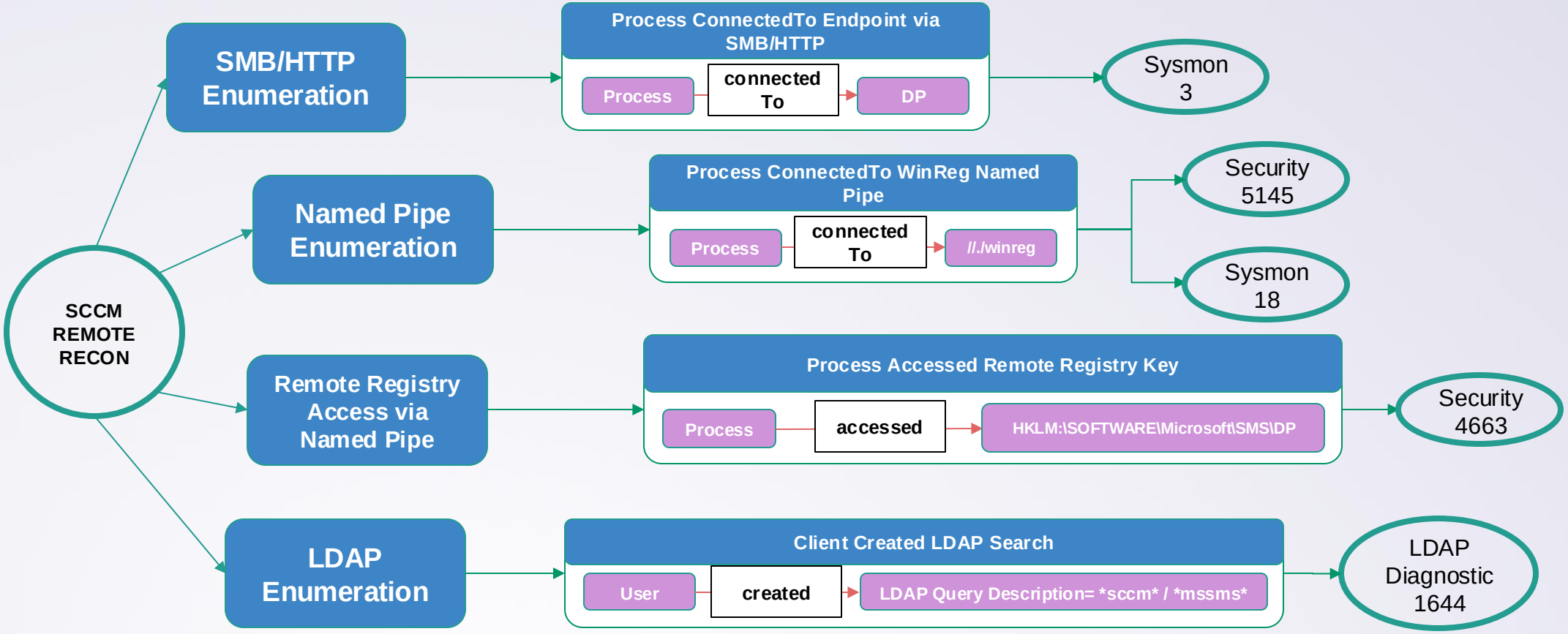
Authenticated domain accounts can leverage LDAP, SMB/SMB named pipes, HTTP to *remotely* identify primary (including CAS), secondary site servers, MPs, and DPs.

In an SCCM environment domain controllers contain a container called “System Management” which references the SCCM infrastructure. Some offensive tooling will connect and enumerate the referenced machine accounts in this container.

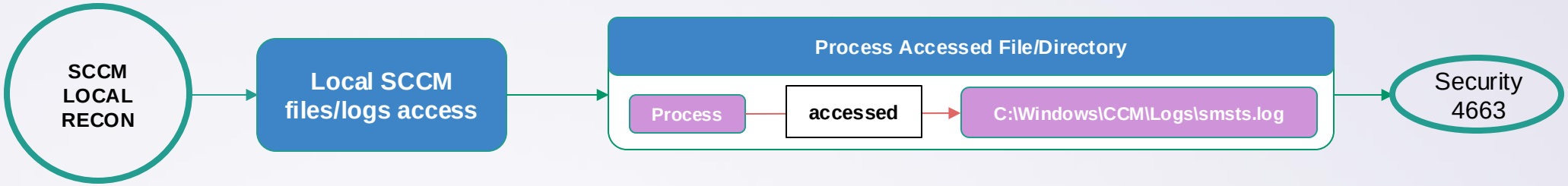
Additionally, local enumeration on SCCM clients works just as well by enumerating key files such as:

- C:\Windows\CCM\Logs\smsts.log
- C:\Windows\ccmcache
- C:\Windows\ccmsetup

SCCM Remote Recon



SCCM Local Recon

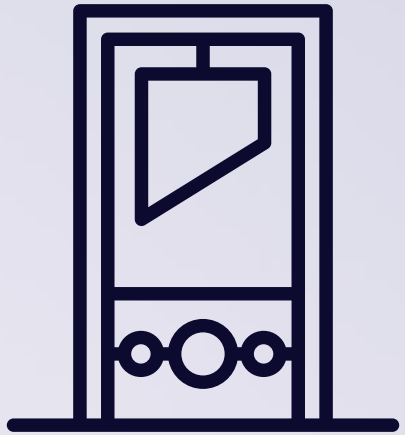


The SACL could be set on all the following:

- C:\Windows\CCM
- C:\Windows\CCM\Logs\smsts.log
- C:\Windows\ccmcache
- C:\Windows\ccmsetup

Domain Persistence Techniques

SCCM EXEC-1



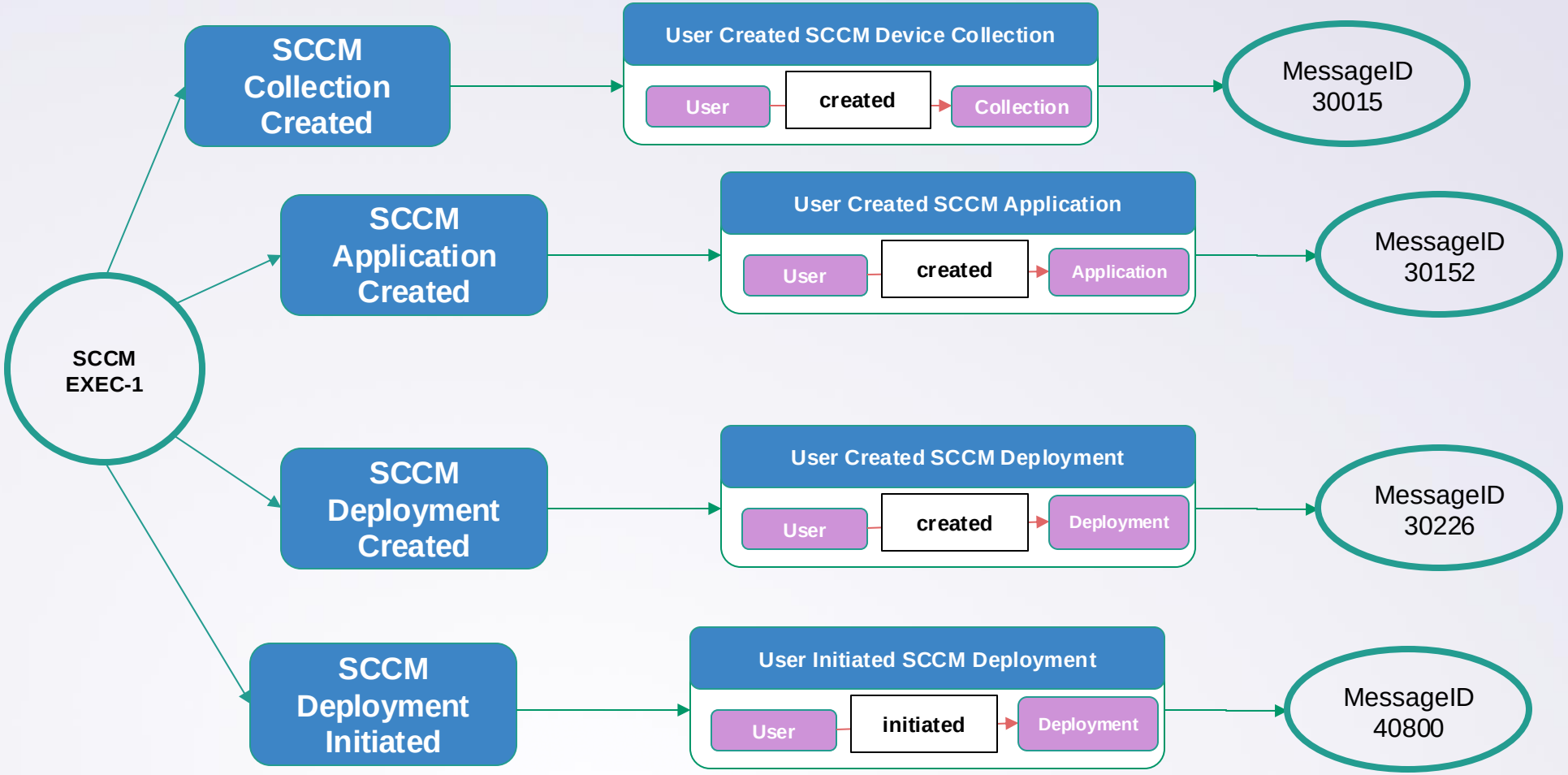
SCCM allows administrators to deploy applications to client devices from a specified UNC path, running them as SYSTEM, the currently logged-in user, or a specific user. This functionality can be exploited to execute malicious applications on remote systems.

Attackers can abuse this feature to deploy applications, execute malicious binaries, or relay NTLM authentication to gain lateral movement. Applications can be hidden from the SCCM console, making detection difficult.

The typical operational flow:

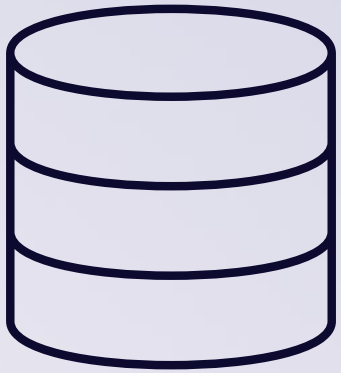
- Create a Collection of devices
- Create an Application and Scope
- Create a Deployment
- Initiate Deployment

SCCM EXEC-1


SCCM EXEC-1
SCCM Collection Created
User Created SCCM Device Collection
User
created
Collection
MessageID 30015
SCCM Application Created
User Created SCCM Application
User
created
Application
MessageID 30152
SCCM Deployment Created
User Created SCCM Deployment
User
created
Deployment
MessageID 30226
SCCM Deployment Initiated
User Initiated SCCM Deployment
User
initiated
Deployment
MessageID 40800

Domain Persistence Techniques

SCCM TAKEOVER-1



By coercing NTLM authentication from a primary site server, SMS Provider, or passive site server, and relaying it to the SCCM site database, an adversary can exploit default permissions to grant an arbitrary domain account the SCCM "Full Administrator" role, enabling full control over the SCCM hierarchy.

An attacker with "Full Administrator" privileges can execute arbitrary programs as SYSTEM or as any user, query client devices in real-time using tools like *CMPivot*, and perform lateral movement and privilege escalation across the network.

The typical operational flow:

- Coerce NTLM authentication from a target SCCM site server via SMB
- Relay coerced authentication to site MSSQL database
- Utilize SQL commands to grant themselves ”Full Administrator” role
- Site Took Over :p

Domain Persistence Techniques

SCCM Site Takeover

Data Source:

- **WinSec 4624 (Successful Logon)**
  - Logon Type 3 AND AuthPackage=NTLMSSP
- **WinSec 7040 (Service Stop/Disabled)**
  - LANMANSERVER | SRV2 | SRVNET
- **WinSec 5145 | Sysmon 18 (Detailed File Share | Named Pipe Conn)**
  - Share Name \ Named Pipe includes: \\.\LSARPC

Detection Strategy:

- **4624 Detection:**
  - Compare the Subject\Account Name field to that of the Source host that the logon originated from.
  - Filter **Subject\Account Name** based on SCCM primary (including CAS), secondary site servers.
  - Filter **Source host** based on SCCM database servers.
- **7040 Detection:**
  - Disabling these services stops the Windows Kernel from binding to port 445. Identify anomalous service stop/disabling
- **5145 Detection:**
  - LSARPC named pipe is utilized all the time natively. **Heavy baselining is required.**

Conclusion

Domain Persistence: Detection

Many organizations do not have custom detections designed to identify domain persistence behavior. These detections are important because they represent a larger attack path.

Domain Persistence: Recovery

Reducing adversary dwell time after identifying these techniques is critical. Organizations that have pre-planned/documented restore playbooks can confidently recover from these scenarios quickly.

Questions & Resources

Research and Validate

The below link includes many of the books, blogs, and references that we dove into while researching these topics:

- GitHub Link

Special Thanks:

- Alex Sou
- Jared Atkinson
- Chris Thompson
- Garrett White
- Garrett Foster
- Will Schroder
- Lee Chagolla-Christensen

Thank you

Josh Prager | jprager@specterops.com

Nico Shyne | nshyne@specterops.io