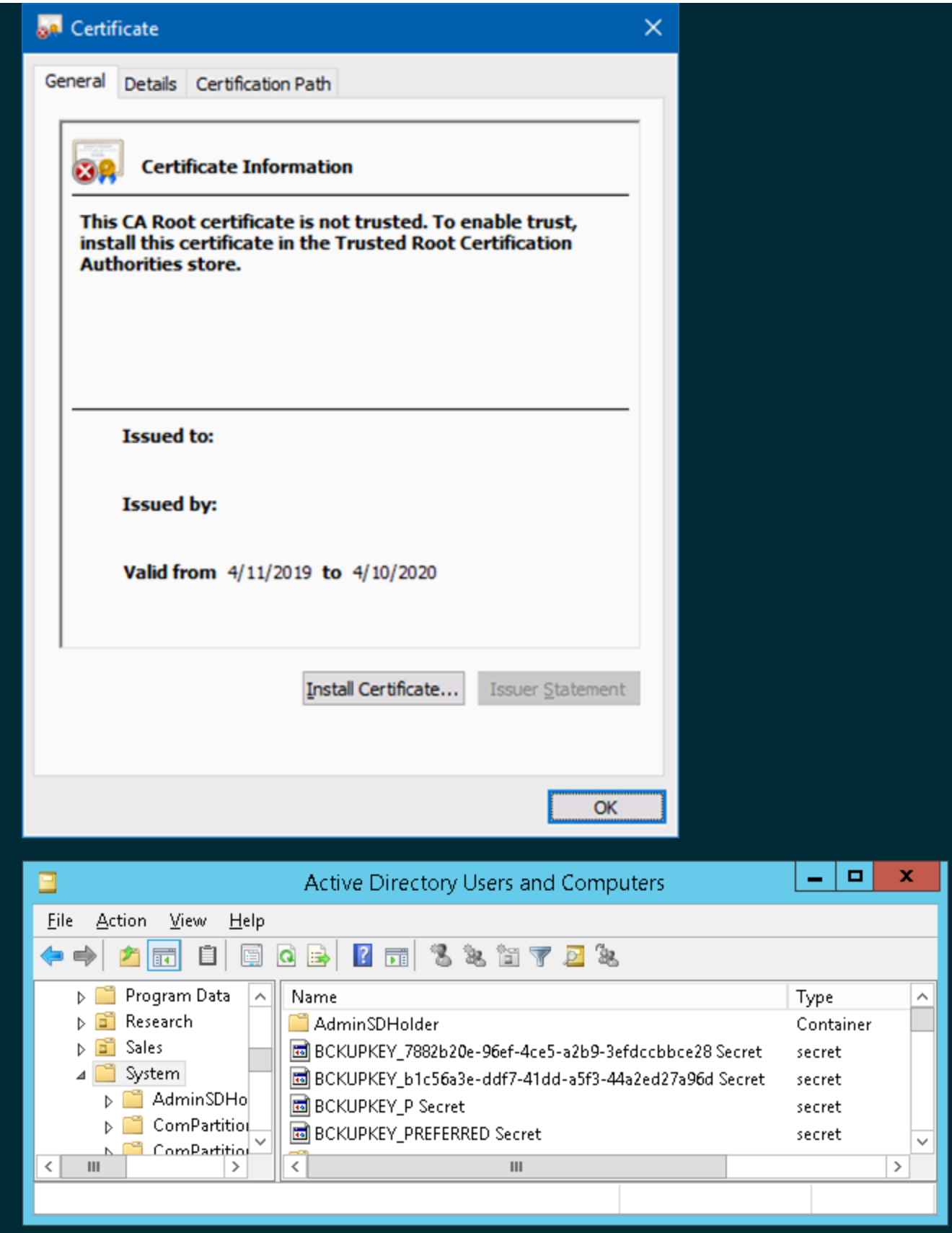
Detecting DPAPI Backup Key Theft

Jul 31, 2023 Michael Grafnetter

Introduction

The Data Protection API (DPAPI) in Windows is used to encrypt passwords saved by browsers, certificate private keys, and other sensitive data. Domain controllers (DCs) hold backup master keys that can be used to decrypt all such secrets encrypted with DPAPI on domain-joined computers. These backup keys are stored as self-signed certificates in Active Directory (AD) objects of type `secret` called `BCKUPKEY_*`:



Attackers with sufficient permissions can fetch these backup keys from AD through the Local Security Authority (Domain Policy) Remote Protocol (MS-LSAD / LSARPC) and use them to decrypt any secrets protected by DPAPI

on all domain-joined Windows machines.

It is therefore important for organizations to be able to detect the theft of DPAPI backup keys from AD by malicious actors. This article describes various ways of discovering this attack technique.

Attack Classification

MITRE ATT&CK® Tactic	Credential Access (TA0006)
MITRE ATT&CK® Technique	Credentials from Password Stores (T1555)
MITRE ATT&CK® Sub-Technique	Unsecured Credentials: Private Keys (T1552.004)
Tenable® Indicator of Attack	DPAPI Domain Backup Key Extraction
Microsoft® Defender for Identity Alert	Malicious request of Data Protection API master key (alert ID 2020)

Detection on Domain Controllers

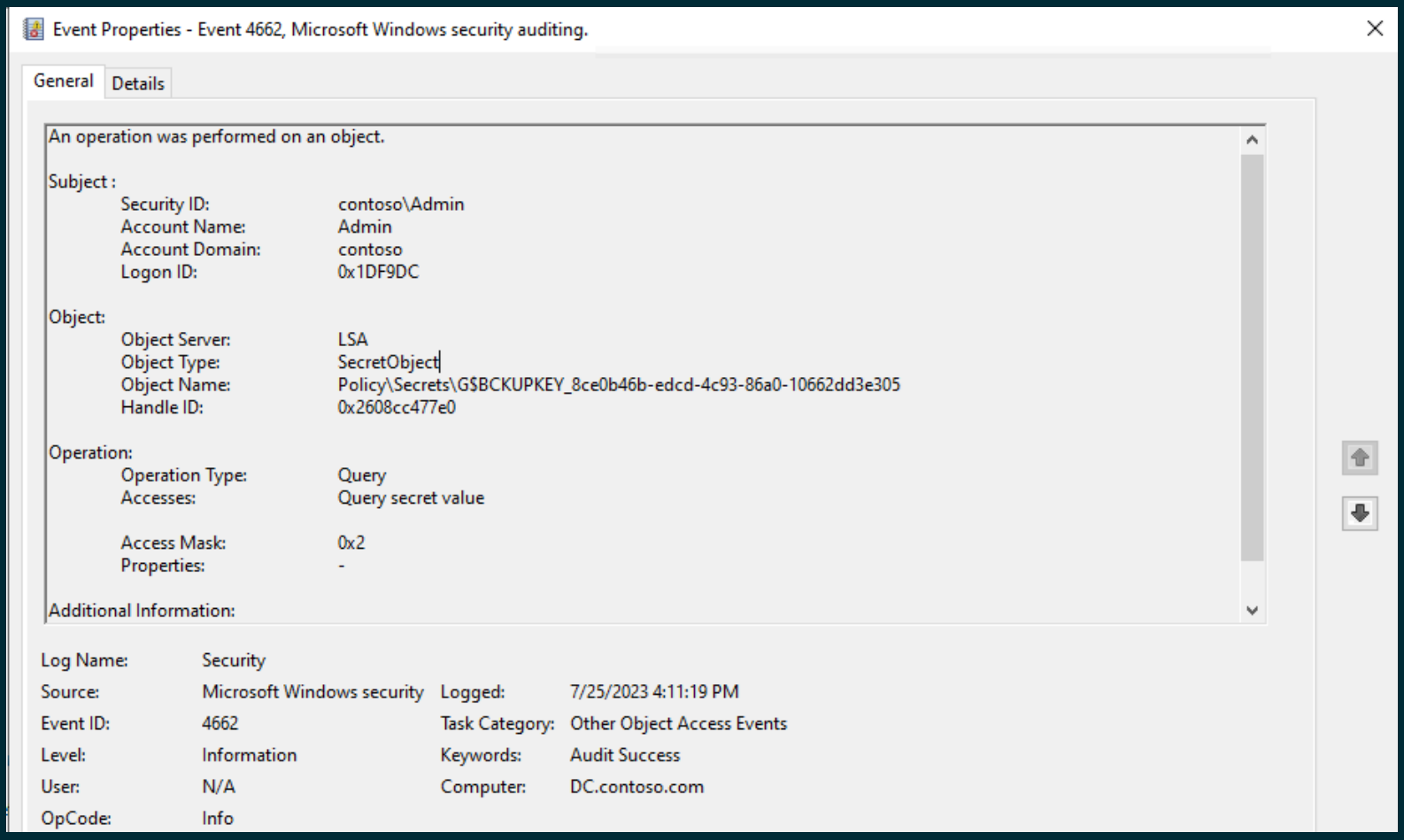
The most reliable way of detecting this attack technique is to monitor domain controllers for suspicious operations.

Domain Controller Security Event Logs

When a DPAPI backup key is retrieved from a domain controller (DC) through the MS-LSAD protocol, an undocumented event with the following properties is generated on that DC:

Log Name	Security
Event ID	4662
Keywords	Audit Success
Task Category	Other Object Access Events

Object Server	LSA
Object Type	SecretObject
Accesses	Query secret value
Object Name	Policy\Secrets\G$BCKUPKEY_*



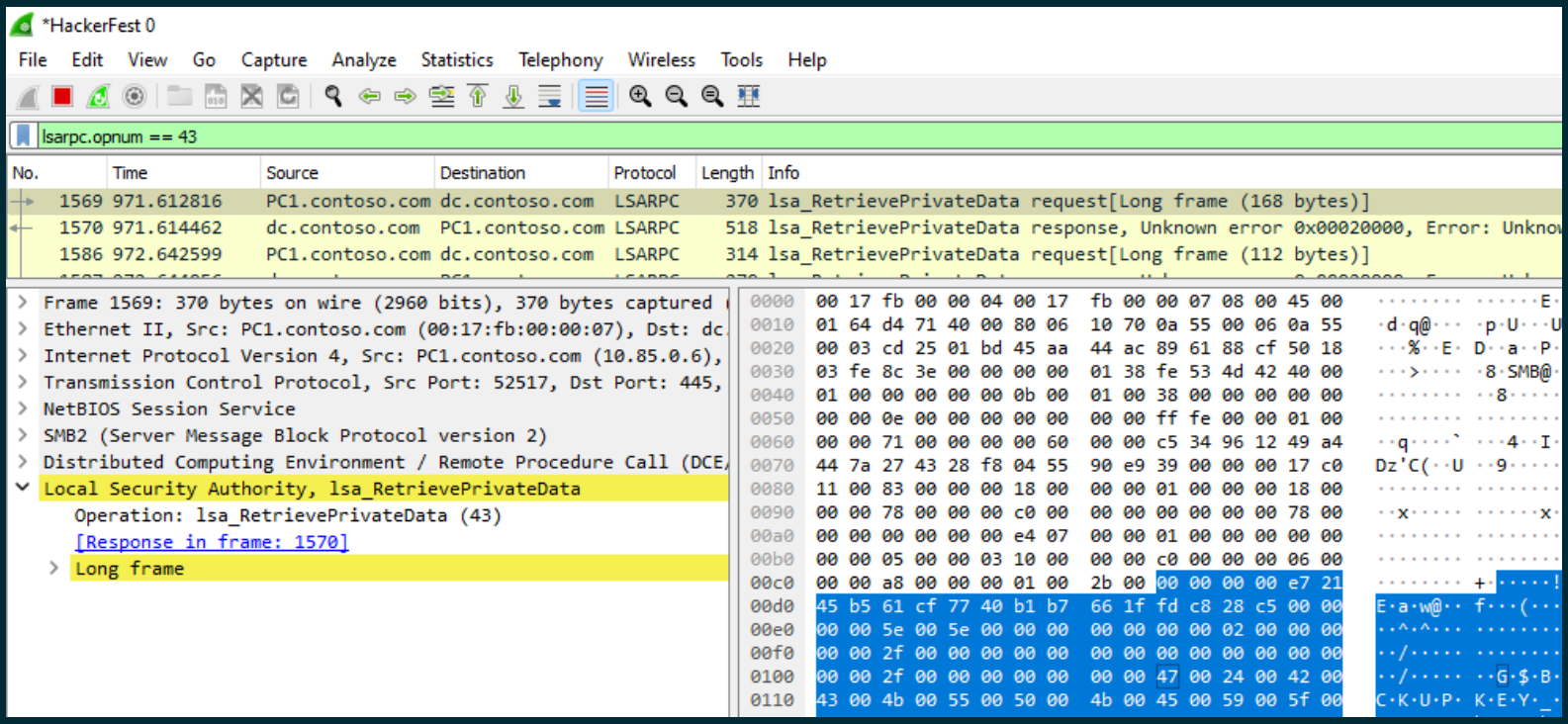
Auditing of `Success` events of type `Audit Other Object Access Events` from the `Object Access` category in `Advanced Audit Policy Configuration` must first be enabled on all DCs.

Domain Controller Network Traffic

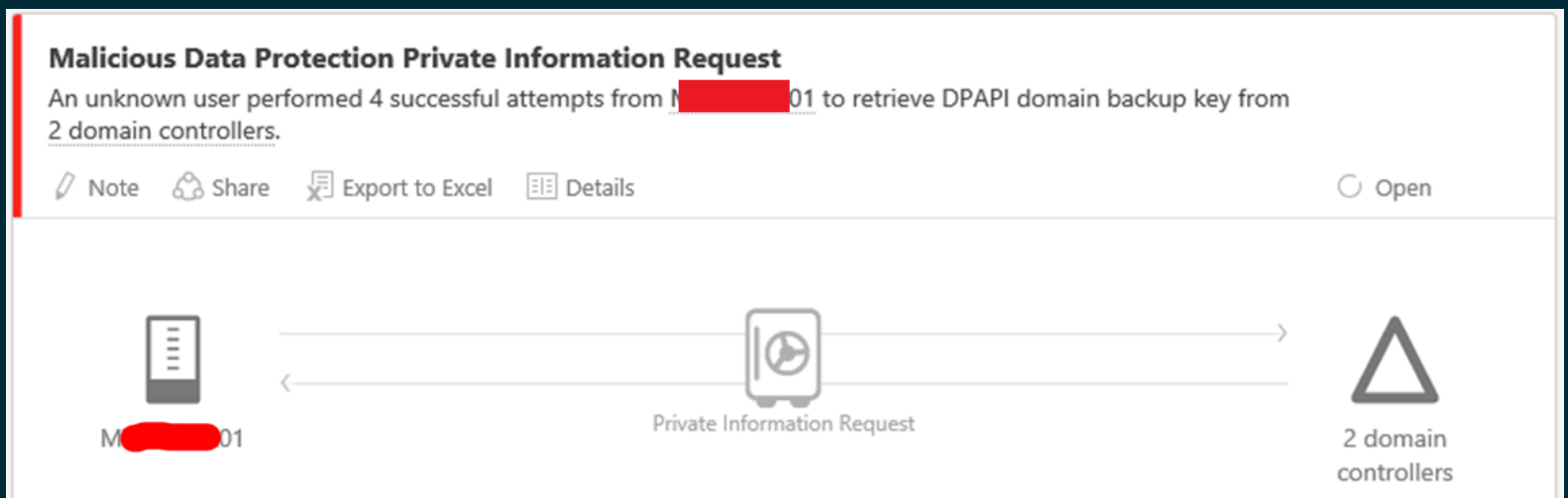
The misuse of the MS-LSAD / LSARPC protocol can also be detected through deep packet inspection of domain controller traffic:

RPC protocol UUID	12345778-1234-ABCD-EF00-0123456789AB	
RPC operation name	LsarRetrievePrivateData	
RPC operation number	43	

Both `RPC/TCP` (TCP port 135 + ephemeral port) and `RPC/NP` (TCP port 445) bindings can be used by clients. In WireShark, the `lsarpc.opnum == 43` display filter can be used to identify this type of traffic:



This detection technique is most probably used by Microsoft Defender for Identity and the already discontinued Advanced Threat Analytics (ATA):



(Un)Fortunately, some organizations are slowly deploying SMB3 encryption even on DCs, which breaks this detection method, when the `RPC/NP` binding is used. IPSec tunneling would additionally break the detection at the network level for the `RPC/TCP` binding, but IPSec is rarely used.

Detection on Endpoints

EDR solutions could theoretically be used to detect when corporate endpoints are misused to retrieve DPAPI backup keys from remote domain controllers. Unfortunately, all detection techniques listed in this section can easily be bypassed by obfuscation.

Malicious Commands

Execution the the following off-the-shelf hacktools should raise an alert:

- `mimikatz.exe` tool with the `lsadump::backupkeys` parameter.
- `SharpDPAPI.exe` tool with the `backupkey` parameter.
- `Get-LsaBackupKey` PowerShell cmdlet from the DSInternals module.

This detection technique is used by Microsoft Defender for Endpoint, among others.

Suspicious File Names

Both Mimikatz and DSInternals export stolen DPAPI backup keys into files with the following name format:

- `ntds_capi_*.pfx`
- `ntds_capi_*.pvk`

The presence of these files should thus be considered an indicator of compromise. This detection technique is utilized by Elastic Security for endpoint, among others.

Suspicious Win32 API Calls

All 3 hacktools mentioned in this chapter perform calls to the `LsaRetrievePrivateData` function from `advapi32.dll`, which can also be picked up by EDRs. This appears to be the most reliable detection method on endpoints, but it could still be bypassed by directly performing the respective RPC calls.

Alternative Attack Techniques

Usage of the MS-LSAD protocol is one of many ways of extracting DPAPI backup keys from domain controllers. Other techniques include, but are not limited to:

- Fetching the keys through the directory replication protocol.
- Extracting the keys from `ntds.dit` database files.

The detection of these techniques is out-of-scope of this article.

Additional Resources

- Microsoft: DPAPI backup keys on Active Directory domain controllers
- DSInternals: Retrieving DPAPI Backup Keys from Active Directory
- CQURE: Extracting Roamed Private Keys from Active Directory
- SpecterOps: HomeOperational Guidance for Offensive User DPAPI Abuse
- Sygnia: The Downfall Of Dpapi Top Secret Weapon