GitHub - GhostPack/PSPKIAudit: PowerShell toolkit for AD CS auditing based on the PSPKI toolkit.

By leechristensen

Archived: 2026-04-05 13:45:39 UTC

PowerShell toolkit for auditing Active Directory Certificate Services (AD CS).

It is built on top of PKISolution's PSPKI toolkit (Microsoft Public License). This repo contains a newer version of PSPKI than what's available in the PSGallery (see the `PSPKI` directory). Vadims Podans (the creator of PSPKI) graciously provided this version as it contains patches for several bugs.

This README is only meant as a starting point- for complete details and defensive guidance, please see the "Certified Pre-Owned" whitepaper.

The module contains the following main functions:

1. Invoke-PKIAudit - Audits the current Forest's AD CS settings, primarily analyzing the CA server and published templates for potential privilege escalation opportunities.
2. Get-CertRequest - Examines a CA's issued certificates by querying the CA's database. Primary intention is to discover certificate requests that may have abused a certificate template privilege escalation vulnerability. In addition, if a user or computer is compromised, incident responders can use it to find certificates the CA server had issued to the compromised user/computer (which should then be revoked).

WARNING: This code is beta! We are confident that `Invoke-PKIAudit` will not impact the environment as the amount of data it queries is quite limited. We have not done rigorous testing with `Get-CertRequest` against typical CA server workloads. `Get-CertRequest` queries the CA's database directly and may have to process thousands of results, which might impact performance.

IF THERE ARE NO RESULTS, THIS IS NOT A GUARANTEE THAT YOUR ENVIRONMENT IS SECURE!!

WE ALSO CANNOT GUARANTEE THAT OUR MITIGATION ADVICE WILL MAKE YOUR ENVIRONMENT SECURE OR WILL NOT DISRUPT OPERATIONS!

It is your responsibility to talk to your Active Directory/PKI/Architecture team(s) to determine the best mitigations for your environment.

If the code breaks, or we missed something, please submit an issue or pull request for a fix!

Setup

Requirements

Install the following on a Windows machine using an elevated PowerShell prompt (PowerShell verion 5.1 or above):

- RSAT's **Certificate Services** and **Active Directory** features. Install with the following command:

```
Get-WindowsCapability -Online -Name "Rsat.*" | where Name -match "CertificateServices|ActiveDirectory" | Add-W
```

- The PSPKI PowerShell module. Install with the following command:

```
Install-Module -Name PSPKI
```

Import

Download the module extract it to a folder. Then, import the module using the following commands:

```
 cd PSPKIAudit
Get-ChildItem -Recurse | Unblock-File

Import-Module .\PSPKIAudit.psd1
```

Auditing AD CS Misconfigurations

Running `Invoke-PKIAudit` will run all auditing checks against AD CS in the current domain, including enumerating various Certificate Authority and Certificate Template settings. To audit a specific CA, you can run `Invoke-PKIAudit -CAComputerName CA.DOMAIN.COM` or `Invoke-PKIAudit -CAName X-Y-Z`.

Any misconfigurations (ESC1-8) will appear as properties on the CA/template results displayed to identify the specific misconfiguration found.

If you want to change the groups/users used to test enrollment/access control, modify the `$CommonLowprivPrincipals` regex at the top of `Invoke-PKIAudit.ps1`

If you want to export all CA information to a csv, run: `Get-AuditCertificateAuthority [-CAComputerName CA.DOMAIN.COM | -CAName X-Y-Z] | Export-Csv -NoTypeInformation CAs.csv`

If you want to export ALL published template information to a csv (not just vulnerable templates), run: `Get-AuditCertificateTemplate [-CAComputerName CA.DOMAIN.COM | -CAName X-Y-Z] | Export-Csv -NoTypeInformation templates.csv`

Output Explanation

There are two main sections of output, details about discovered CAs and details about potentially vulnerable templates.

For certificate authority results:

Certificate Authority Property	**Description**
ComputerName	The system the CA is running on.
CAName	The name of the CA.
ConfigString	The full COMPUTER\CA_NAME configuration string.
IsRoot	If the CA is a root CA.
AllowsUserSuppliedSans	If the CA has the `EDITF_ATTRIBUTESUBJECTALTNAME2` flag set.
VulnerableACL	Whether the CA has a vulnerable ACL setting.
EnrollmentPrincipals	Principals who have the `Enroll` privilege at the CA level.
EnrollmentEndpoints	The CA's web services enrollment endpoints.
NTLMEnrollmentEndpoints	The CA's web services enrollment endpoints that have NTLM enabled.
DACL	The full access control information.
Misconfigurations	ESCX indicating the specific misconfiguration present (if any).

For certificate template results:

Property	**Description**
CA	The full CA ConfigString the template is published on (null for not published).
Name	The template name.

Property	Description
SchemaVersion	The schema version (1/2/3) of the template.
OID	The unique object identifier for the template.
VulnerableTemplateACL	True if the template has a vulnerable ACL setting.
LowPrivCanEnroll	True if low-privileged users can enroll in the template.
EnrolleeSuppliesSubject	True if the `CT_FLAG_ENROLLEE_SUPPLIES_SUBJECT` flag is present (i.e., users can supply arbitrary SANs).
EnhancedKeyUsage	The usage EKUs enabled in the template.
HasAuthenticationEku	True if the template has an EKU that allows for authentication.
HasDangerousEku	True if the template has a "dangerous" (Any Purpose or null) EKU.
EnrollmentAgentTemplate	True if the template has the "Certificate Request Agent" EKU.
CAManagerApproval	True if manager approvals are needed for enrollment.
IssuanceRequirements	Authorized signature information.
ValidityPeriod	How long the certificate is valid for.
RenewalPeriod	The renewal period for the certificate.
Owner	The principal who owns the certificate.
DACL	The full access control information.
Misconfigurations	ESCX indicating the specific misconfiguration present (if any).

ESC1 - Misconfigured Certificate Templates

Details

This privilege escalation scenario occurs when the following conditions are met:

1. **The Enterprise CA grants low-privileged users enrollment rights.** The Enterprise CA's configuration must permit low-privileged users the ability to request certificates. See the "Background - Enrollment" section at the beginning of the whitepaper paper for more details.
2. **Manager approval is disabled.** This setting necessitates that a user with certificate "manager" permissions review and approve the requested certificate before the certificate is issued. See the "Background - Issuance Requirements" section at the beginning of the whitepaper paper for more details.

3. **No authorized signatures are required.** This setting requires any CSR to be signed by an existing authorized certificate. See the "Background - Issuance Requirements" section at the beginning of the whitepaper for more details.

4. **An overly permissive certificate template security descriptor grants certificate enrollment rights to low-privileged users.** Having certificate enrollment rights allows a low-privileged attacker to request and obtain a certificate based on the template. Enrollment Rights are granted via the certificate template AD object's security descriptor.

5. **The certificate template defines EKUs that enable authentication.** Applicable EKUs include Client Authentication (OID 1.3.6.1.5.5.7.3.2), PKINIT Client Authentication (OID 1.3.6.1.5.2.3.4), or Smart Card Logon (OID 1.3.6.1.4.1.311.20.2.2).

6. **The certificate template allows requesters to specify a subjectAltName (SAN) in the CSR.** If a requester can specify the SAN in a CSR, the requester can request a certificate as anyone (e.g., a domain admin user). The certificate template's AD object specifies if the requester can specify the SAN in its mspki-certificate-name-flag property. The mspki-certificate-name-flag property is a bitmask and if the CT_FLAG_ENROLLEE_SUPPLIES_SUBJECT flag is present, a requester can specify the SAN.

TL;DR This situation means that a unprivileged users can request a certificate that can be used for domain authentication, where they can specify an arbitrary alternative name (like a domain admin). This can result in a working certificate for an elevated user like a domain admin!

Example

```
 [!] Potentially vulnerable Certificate Templates:

CA                      : dc.theshire.local\theshire-DC-CA
Name                    : ESC1Template
SchemaVersion           : 2
OID                     : ESC1 Template (1.3.6.1.4.1.311.21.8.10395027.10224472.4213181.15714845.1171465.9.10657
VulnerableTemplateACL   : False
LowPrivCanEnroll        : True
EnrolleeSuppliesSubject : True
EnhancedKeyUsage        : Client Authentication (1.3.6.1.5.5.7.3.2)|Secure Email (1.3.6.1.5.5.7.3.4)|Encrypting
HasAuthenticationEku    : True
HasDangerousEku         : False
EnrollmentAgentTemplate : False
CAManagerApproval       : False
IssuanceRequirements    : [Issuance Requirements]
                            Authorized signature count: 0
                            Reenrollment requires: same criteria as for enrollment.
ValidityPeriod          : 1 years
RenewalPeriod           : 6 weeks
Owner                   : THESHIRE\localadmin
DACL                    : NT AUTHORITY\Authenticated Users (Allow) - Read
```

```
                          THESHIRE\Domain Admins (Allow) - Read, Write, Enroll
                          THESHIRE\Domain Users (Allow) - Enroll
                          THESHIRE\Enterprise Admins (Allow) - Read, Write, Enroll
                          THESHIRE\localadmin (Allow) - Read, Write
Misconfigurations       : ESC1
```

Mitigations

There are a few options. First, right click the affected certificate template in the Certificate Templates Console (certtmpl.msc) and click "Properties"

1. Remove the CT_FLAG_ENROLLEE_SUPPLIES_SUBJECT flag via "Subject Name", unchecking "Supply in Request".
    - This prevents arbitrary SAN specification in the CSR. **Unless alternate names are really needed for this template, this is probably the best fix.**
2. Remove the "Client Authentication" and/or "Smart Card Logon" EKUS via "Extensions" -> "Application Policies".
    - This prevents domain authentication with this template.
3. Enable **"CA Certificate Manager Approval"** in "Issuance Requirements".
    - This puts requests for this template in the "Pending Requests" queue that must be manually approved by a certificate manager.
4. Enable **Authorized Signatures"** in "Issuance Requirements" (if you know what you're doing).
    - This forces CSRs to be co-signed by an Enrollment Agent certificate.
5. Remove the ability for low-privileged users from enrolling in this template via "Security" and removing the appropriate **Enroll** privilege.

ESC2 - Misconfigured Certificate Templates

Details

This privilege escalation scenario occurs when the following conditions are met:

1. **The Enterprise CA grants low-privileged users enrollment rights.** Details are the same as in ESC1.
2. **Manager approval is disabled.** Details are the same as in ESC1.
3. **No authorized signatures are required.** Details are the same as in ESC1.
4. **An overly permissive certificate template security descriptor grants certificate enrollment rights to low-privileged users.** Details are the same as in ESC1.
5. **The certificate template defines Any Purpose EKUs or no EKU.** The Any Purpose (OID 2.5.29.37.0) can be used for (surprise!) any purpose, including client authentication. If no EKUs are specified - i.e. the pkiextendedkeyusage is empty or the attribute doesn't exist - then the certificate is the equivalent of a subordinate CA certificate and can be used for anything.

TL;DR This is very similar to ESC1, however with the Any Purpose or no EKU, the CT_FLAG_ENROLLEE_SUPPLIES_SUBJECT flag does not need to be present.

Example

```
 [!] Potentially vulnerable Certificate Templates:

CA                      : dc.theshire.local\theshire-DC-CA
Name                    : ESC2Template
SchemaVersion           : 2
OID                     : ESC2 Template (1.3.6.1.4.1.311.21.8.10395027.10224472.4213181.15714845.1171465.9.7730
VulnerableTemplateACL   : False
LowPrivCanEnroll        : True
EnrolleeSuppliesSubject : False
EnhancedKeyUsage        :
HasAuthenticationEku    : True
HasDangerousEku         : True
EnrollmentAgentTemplate : False
CAManagerApproval       : False
IssuanceRequirements    : [Issuance Requirements]
                            Authorized signature count: 0
                            Reenrollment requires: same criteria as for enrollment.
ValidityPeriod          : 1 years
RenewalPeriod           : 6 weeks
Owner                   : THESHIRE\localadmin
DACL                    : NT AUTHORITY\Authenticated Users (Allow) - Read
                          THESHIRE\Domain Admins (Allow) - Read, Write, Enroll
                          THESHIRE\Domain Users (Allow) - Enroll
                          THESHIRE\Enterprise Admins (Allow) - Read, Write, Enroll
                          THESHIRE\localadmin (Allow) - Read, Write
Misconfigurations       : ESC2
```

Mitigations

There are a few options. First, right click the affected certificate template in the Certificate Templates Console (certtmpl.msc) and click "Properties"

1. Remove the ability for low-privileged users from enrolling in this template via "Security" and removing the appropriate **Enroll** privilege.
    - This is likely the best fix, as these sensitive EKUs should not be available to low-privileged users!
2. Enable **"CA Certificate Manager Approval"** in "Issuance Requirements".
    - This puts requests for this template in the "Pending Requests" queue that must be manually approved by a certificate manager.
3. Enable **Authorized Signatures"** in "Issuance Requirements" (if you know what you're doing).
    - This forces CSRs to be co-signed by an Enrollment Agent certificate.

ESC3 - Misconfigured Enrollment Agent Templates

Details

This privilege escalation scenario occurs when the following conditions are met:

1. **The Enterprise CA grants low-privileged users enrollment rights.** Details are the same as in ESC1.
2. **Manager approval is disabled.** Details are the same as in ESC1.
3. **No authorized signatures are required.** Details are the same as in ESC1.
4. **An overly permissive certificate template security descriptor grants certificate enrollment rights to low-privileged users.** Details are the same as in ESC1.
5. **The certificate template defines the Certificate Request Agent EKU.** The Certificate Request Agent EKU (OID 1.3.6.1.4.1.311.20.2.1) allows a principal to enroll for *another* certificate template on behalf of another user.
6. **Enrollment agents restrictions are not implemented on the CA.**

TL;DR Someone with a Certificate Request (aka Enrollment) Agent certificate can enroll in other certificates on behalf of any user in the domain, for any Schema Version 1 template or any Schema Version 2+ template that requires the appropriate "Authorized Signatures/Application Policy" Issuance Requirement, unless "Enrollment Agent Restrictions" are implemented at the CA level.

Example

```
 [!] Potentially vulnerable Certificate Templates:

CA                      : dc.theshire.local\theshire-DC-CA
Name                    : ESC3Template
SchemaVersion           : 2
OID                     : ESC3 Template (1.3.6.1.4.1.311.21.8.10395027.10224472.4213181.15714845.1171465.9.43003
VulnerableTemplateACL   : False
LowPrivCanEnroll        : True
EnrolleeSuppliesSubject : False
EnhancedKeyUsage        : Certificate Request Agent (1.3.6.1.4.1.311.20.2.1)
HasAuthenticationEku    : False
HasDangerousEku         : False
EnrollmentAgentTemplate : True
CAManagerApproval       : False
IssuanceRequirements    : [Issuance Requirements]
                            Authorized signature count: 0
                            Reenrollment requires: same criteria as for enrollment.
ValidityPeriod          : 1 years
RenewalPeriod           : 6 weeks
```

```
Owner                   : THESHIRE\localadmin
DACL                    : NT AUTHORITY\Authenticated Users (Allow) - Read
                          THESHIRE\Domain Admins (Allow) - Read, Write, Enroll
                          THESHIRE\Domain Users (Allow) - Enroll
                          THESHIRE\Enterprise Admins (Allow) - Read, Write, Enroll
                          THESHIRE\localadmin (Allow) - Read, Write
Misconfigurations       : ESC3
```

Mitigations

There are a few options. First, right click the affected certificate template in the Certificate Templates Console (certtmpl.msc) and click "Properties"

1. Remove the ability for low-privileged users from enrolling in this template via "Security" and removing the appropriate **Enroll** privilege.
    - This is likely the best fix, as this sensitive EKU should not be available to low-privileged users!
2. Enable **"CA Certificate Manager Approval"** in "Issuance Requirements".
    - This puts requests for this template in the "Pending Requests" queue that must be manually approved by a certificate manager.

You can also implement "Enrollment Agent Restrictions" via the Certification Authority console (certsrv.msc). On the affected CA, right click the CA name and click "Properties" -> "Enrollment Agents". There is more information on this approach here.

ESC4 - Vulnerable Certificate Template Access Control

Details

Certificate templates are securable objects in Active Directory, meaning they have a security descriptor that specifies which Active Directory principals have specific permissions over the template. Templates that have vulnerable access control grant unintended principals the ability to modify settings in the template. With modification rights, an attacker can set vulnerable EKUs (ESC1-ESC3), flip settings like CT_FLAG_ENROLLEE_SUPPLIES_SUBJECT (ESC1), and/or remove "Issuance Requirements" like manager approval or authorized signatures.

Example

```
 [!] Potentially vulnerable Certificate Templates:

CA                      : dc.theshire.local\theshire-DC-CA
Name                    : ESC4Template
SchemaVersion           : 2
OID                     : ESC4 Template (1.3.6.1.4.1.311.21.8.10395027.10224472.4213181.15714845.1171465.9.1768
VulnerableTemplateACL   : True
LowPrivCanEnroll        : True
```

```
EnrolleeSuppliesSubject : False
EnhancedKeyUsage        : Client Authentication (1.3.6.1.5.5.7.3.2)|Secure Email (1.3.6.1.5.5.7.3.4)|Encrypting
HasAuthenticationEku    : True
HasDangerousEku         : False
EnrollmentAgentTemplate : False
CAManagerApproval       : False
IssuanceRequirements    : [Issuance Requirements]
                            Authorized signature count: 0
                            Reenrollment requires: same criteria as for enrollment.
ValidityPeriod          : 1 years
RenewalPeriod           : 6 weeks
Owner                   : THESHIRE\localadmin
DACL                    : NT AUTHORITY\Authenticated Users (Allow) - Read, Write
                          THESHIRE\Domain Admins (Allow) - Read, Write, Enroll
                          THESHIRE\Domain Users (Allow) - Read, Enroll
                          THESHIRE\Enterprise Admins (Allow) - Read, Write, Enroll
                          THESHIRE\localadmin (Allow) - Read, Write
Misconfigurations       : ESC4
```

Mitigations

Right click the affected certificate template in the Certificate Templates Console (certtmpl.msc) and click "Properties".

Go to "Security" and remove the vulnerable access control entry.

ESC5 - Vulnerable PKI AD Object Access Control

Details

A number of objects outside of certificate templates and the certificate authority itself can have a security impact on the entire AD CS system.

These possibilities include (but are not limited to):

- CA server's AD computer object (i.e., compromise through RBCD)
- The CA server's RPC/DCOM server
- PKI-related AD objects. Any descendant AD object or container in the container CN=Public Key Services,CN=Services,CN=Configuration,DC=,DC= (e.g., the Certificate Templates container, Certification Authorities container, the NTAuthCertificates object, etc.)

Due to the broad scope of this specific misconfiguration, we do not currently check for ESC5 by default in this toolkit.

Access paths into the CA server itself can be found in current BloodHound collection.

The CA server's RPC/DCOM server security require manual analysis.

The following commands outputs a list of users and the control/edit right the user has over a PKI-related AD object.

```
 $Controllers = Get-AuditPKIADObjectControllers
Format-PKIAdObjectControllers $Controllers
```

Ensure all principals in the results absolutely require the listed rights. Often times non-tier 0 accounts (be it low privileged users/groups or lower-privileged non-AD server admins) have control of PKI-related AD objects.

Example

```
 THESHIRE\Cert Publishers (S-1-5-21-3022474190-4230777124-3051344698-517)
    GenericAll         CN=THESHIRE-DC-CA,CN=Certification Authorities,CN=Public Key Services,CN=Services,CN=Conf
    GenericAll         CN=AIA,CN=Public Key Services,CN=Services,CN=Configuration,DC=THESHIRE,DC=LOCAL
    GenericAll         CN=DC,CN=CDP,CN=Public Key Services,CN=Services,CN=Configuration,DC=THESHIRE,DC=LOCAL
    GenericAll         CN=THESHIRE-DC-CA,CN=DC,CN=CDP,CN=Public Key Services,CN=Services,CN=Configuration,DC=THE

THESHIRE\DC$ (S-1-5-21-3022474190-4230777124-3051344698-1000)
    WriteOwner         CN=THESHIRE-DC-CA,CN=Enrollment Services,CN=Public Key Services,CN=Services,CN=Configurat
    GenericAll         CN=THESHIRE-DC-CA,CN=AIA,CN=Public Key Services,CN=Services,CN=Configuration,DC=THESHIRE,
    GenericAll         CN=THESHIRE-DC-CA,CN=DC,CN=CDP,CN=Public Key Services,CN=Services,CN=Configuration,DC=THE
    GenericAll         CN=THESHIRE-DC-CA,CN=KRA,CN=Public Key Services,CN=Services,CN=Configuration,DC=THESHIRE,

THESHIRE\Domain Computers (S-1-5-21-3022474190-4230777124-3051344698-515)
    WriteDacl          CN=MisconfiguredTemplate,CN=Certificate Templates,CN=Public Key Services,CN=Services,CN=C

THESHIRE\Domain Users (S-1-5-21-3022474190-4230777124-3051344698-513)
    WriteAllProperties CN=MisconfiguredTemplate,CN=Certificate Templates,CN=Public Key Services,CN=Services,CN=C

THESHIRE\john-sa (S-1-5-21-3022474190-4230777124-3051344698-1602)
    GenericAll         CN=MisconfiguredTemplate,CN=Certificate Templates,CN=Public Key Services,CN=Services,CN=C

NT AUTHORITY\Authenticated Users (S-1-5-11)
    Owner              CN=MisconfiguredTemplate,CN=Certificate Templates,CN=Public Key Services,CN=Services,CN=C
    WriteOwner         CN=MisconfiguredTemplate,CN=Certificate Templates,CN=Public Key Services,CN=Services,CN=C
```

Mitigations

Remove any vulnerable access control entries through Active Directory Users and Computers (*dsa.msc*) or ADSIEdit (*adsiedit.msc*) for configuration objects.

ESC6 - EDITF_ATTRIBUTESUBJECTALTNAME2

Details

If the **EDITF_ATTRIBUTESUBJECTALTNAME2** flag is flipped in the configuration for a Certificate Authority, *ANY* certificate request can specify arbitrary Subject Alternative Names (SANs). This means that ANY template configured for domain authentication that also allows unprivileged users to enroll (e.g., the default **User** template) can be abused to obtain a certificate that allows us to authenticate as a domain admin (or any other active user/machine).

THIS SETTING SHOULD ABSOLUTELY NOT BE SET IN YOUR ENVIRONMENT.

Example

```
 === Certificate Authority ===


ComputerName            : dc.theshire.local
CAName                  : theshire-DC-CA
ConfigString            : dc.theshire.local\theshire-DC-CA
IsRoot                  : True
AllowsUserSuppliedSans  : True
VulnerableACL           : False
EnrollmentPrincipals    : THESHIRE\Domain Users
                          THESHIRE\Domain Computers
                          THESHIRE\certmanager
                          THESHIRE\certadmin
                          THESHIRE\Nested3
EnrollmentEndpoints     :
NTLMEnrollmentEndpoints :
DACL                    : BUILTIN\Administrators (Allow) - ManageCA, ManageCertificates
                          THESHIRE\Domain Admins (Allow) - ManageCA, ManageCertificates
                          THESHIRE\Domain Users (Allow) - Read, Enroll
                          THESHIRE\Domain Computers (Allow) - Enroll
                          THESHIRE\Enterprise Admins (Allow) - ManageCA, ManageCertificates
                          THESHIRE\certmanager (Allow) - ManageCertificates, Enroll
                          THESHIRE\certadmin (Allow) - ManageCA, Enroll
                          THESHIRE\Nested3 (Allow) - ManageCertificates, Enroll
Misconfigurations       : ESC6

[!] The above CA is misconfigured!

...(snip)...

[!] EDITF_ATTRIBUTESUBJECTALTNAME2 set on this CA, the following templates may be vulnerable:

CA                      : dc.theshire.local\theshire-DC-CA
Name                    : User
SchemaVersion           : 1
OID                     : 1.3.6.1.4.1.311.21.8.10395027.10224472.4213181.15714845.1171465.9.1.1
```

```
VulnerableTemplateACL   : False
LowPrivCanEnroll        : True
EnrolleeSuppliesSubject : False
EnhancedKeyUsage        : Encrypting File System (1.3.6.1.4.1.311.10.3.4)|Secure Email (1.3.6.1.5.5.7.3.4)|Clier
HasAuthenticationEku    : True
HasDangerousEku         : False
EnrollmentAgentTemplate : False
CAManagerApproval       : False
IssuanceRequirements    : [Issuance Requirements]
                            Authorized signature count: 0
                            Reenrollment requires: same criteria as for enrollment.
ValidityPeriod          : 1 years
RenewalPeriod           : 6 weeks
Owner                   : THESHIRE\Enterprise Admins
DACL                    : NT AUTHORITY\Authenticated Users (Allow) - Read
                          THESHIRE\Domain Admins (Allow) - Read, Write, Enroll
                          THESHIRE\Domain Users (Allow) - Read, Enroll
                          THESHIRE\Enterprise Admins (Allow) - Read, Write, Enroll
Misconfigurations       :
```

Mitigations

Immediately remove this flag and restart the affected certificate authority from a PowerShell prompt with elevated rights against the CA server:

```
 PS C:\> certutil -config "CA_HOST\CA_NAME" -setreg policy\EditFlags -EDITF_ATTRIBUTESUBJECTALTNAME2

PS C:\> Get-Service -ComputerName CA_HOST certsvc | Restart-Service -Force
```

ESC7 - Vulnerable Certificate Authority Access Control

Details

Outside of certificate templates, a certificate authority itself has a set of permissions that secure various CA actions. These permissions can be accessed from certsrv.msc, right clicking a CA, selecting properties, and switching to the Security tab.

There are two rights that are security sensitive and dangerous if unintended principals possess them:

- **ManageCA** (aka "CA Administrator") - allows for administrative CA actions, including (remotely) flipping the EDITF_ATTRIBUTESUBJECTALTNAME2 bit, resulting in ESC6.
- **ManageCertificates** (aka "Certificate Manager/Officer") - allows the principal to approve pending certificate requests, negating the "Manager Approval" Issuance Requirement/protection

Example

```
 === Certificate Authority ===


ComputerName            : dc.theshire.local
CAName                  : theshire-DC-CA
ConfigString            : dc.theshire.local\theshire-DC-CA
IsRoot                  : True
AllowsUserSuppliedSans  : False
VulnerableACL           : True
EnrollmentPrincipals    : THESHIRE\Domain Users
                          THESHIRE\Domain Computers
                          THESHIRE\certmanager
                          THESHIRE\certadmin
                          THESHIRE\Nested3
EnrollmentEndpoints     :
NTLMEnrollmentEndpoints :
DACL                    : BUILTIN\Administrators (Allow) - ManageCA, ManageCertificates
                          THESHIRE\Domain Admins (Allow) - ManageCA, ManageCertificates
                          THESHIRE\Domain Users (Allow) - ManageCA, Read, Enroll
                          THESHIRE\Domain Computers (Allow) - Enroll
                          THESHIRE\Enterprise Admins (Allow) - ManageCA, ManageCertificates
                          THESHIRE\certmanager (Allow) - ManageCertificates, Enroll
                          THESHIRE\certadmin (Allow) - ManageCA, Enroll
                          THESHIRE\Nested3 (Allow) - ManageCertificates, Enroll
Misconfigurations       : ESC7

[!] The above CA is misconfigured!
```

Mitigations

Open up the Certification Authority console (certsrv.msc) on the affected CA, right click the CA name and click "Properties".

Go to "Security" and remove the vulnerable access control entry.

ESC8 - NTLM Relay to AD CS HTTP Endpoints

NOTE: this particular check in PSPKIAudit only checks if NTLM is present for any published enrollment endpoints. *It does NOT check if Extended Protection for Authentication is present for these NTLM-enabled endoints, so false positives may occur!*

Important

NTLM authentication is disabled for accounts in the Protected Users group. This check may fail if running PSPKIAudit while logged in as a Protected User.

Details

AD CS supports several HTTP-based enrollment methods via additional AD CS server roles that administrators can install. These HTTP-based certificate enrollment interfaces are all vulnerable NTLM relay attacks.

Using NTLM relay, an attacker on a compromised machine can impersonate any inbound-NTLM-authenticating AD account. While impersonating the victim account, an attacker could access these web interfaces and request a client authentication certificate based on the User or Machine certificate templates.

Example

```
 === Certificate Authority ===


ComputerName            : dc.theshire.local
CAName                  : theshire-DC-CA
ConfigString            : dc.theshire.local\theshire-DC-CA
IsRoot                  : True
AllowsUserSuppliedSans  : False
VulnerableACL           : False
EnrollmentPrincipals    : THESHIRE\Domain Users
                          THESHIRE\Domain Computers
                          THESHIRE\certmanager
                          THESHIRE\certadmin
                          THESHIRE\Nested3
EnrollmentEndpoints     : http://dc.theshire.local/certsrv/
NTLMEnrollmentEndpoints : http://dc.theshire.local/certsrv/
DACL                    : BUILTIN\Administrators (Allow) - ManageCA, ManageCertificates
                          THESHIRE\Domain Admins (Allow) - ManageCA, ManageCertificates
                          THESHIRE\Domain Users (Allow) - Read, Enroll
                          THESHIRE\Domain Computers (Allow) - Enroll
                          THESHIRE\Enterprise Admins (Allow) - ManageCA, ManageCertificates
                          THESHIRE\certmanager (Allow) - ManageCertificates, Enroll
                          THESHIRE\certadmin (Allow) - ManageCA, Enroll
                          THESHIRE\Nested3 (Allow) - ManageCertificates, Enroll
Misconfigurations       : ESC8

[!] The above CA is misconfigured!
```

Mitigations

Either remove the HTTP(S) enrollment endpoints, disable NTLM for the endopints, or enable Extended Protection for Authentication. See **Harden AD CS HTTP Endpoints – PREVENT8** in the whitepaper for more deatils.

Misc - Explicit Mappings

Another possible mitigation for some situations is to enforce explicit mappings for certificates. This disables the use of alternate SANs in certificates when authenticating to Active Directory.

For Kerberos, setting the **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Kdc ! UseSubjectAltName** key to 00000000 forces an explicit mapping. There are more details in KB4043463.

Disabling explicit mappings for SChannel is not really documented, but based on our research settings 0x1 or 0x2 to the **HKEY_LOCAL_MACHINE\CurrentControlSet\Control\SecurityProviders\SCHANNEL ! CertificateMappingMethods** key appears to block SANs, but more testing is needed.

Triaging Existing Issued Certificate Requests

WARNING: this functionality has been minimally tested in large environments!

Note: see "Monitor User/Machine Certificate Enrollments - DETECT1" in the whitepaper for additional information and how to perform these searches with certutil.

If you want to examine existing issued certificate requests, for example to see if any requests specified arbitrary SANs, or were requested for specific templates/by specific principals, the `Get-CertRequest [-CAComputerName COMPUTER.DOMAIN.COM | -CAName X-Y-Z]` function builds on various PSPKI functions to give more contextual information.

Specifically, the raw Certificate Signing Request (CSR) is extracted for every currently issued certificate in the domain, and specific information (i.e., whether a SAN was specified, the requestor name/machine/process, etc.) is constructed from the request to enrich the CSR object.

The following flags can be useful:

Flag	Description
-HasSAN	Only return issued certificates that has a Subject Alternative Name specified in the request.
-Requester DOMAIN\USER	Only return issued certificate requests for the specific requesting user.
-Template TEMPLATE_NAME	Only return return issued certificate requests for the specified template name.

To export ALL issued certificate requests to csv, use `Get-CertRequest | Export-CSV -NoTypeInformation requests.csv`.

Here is an example result entry that shows a situation where a Subject Alternative Name (SAN) was specified with Certify:

```
 CA                       : dc.theshire.local\theshire-DC-CA
RequestID                 : 4602
RequesterName             : THESHIRE\cody
RequesterMachineName      : dev.theshire.local
RequesterProcessName      : Certify.exe
SubjectAltNamesExtension  :
SubjectAltNamesAttrib     : Administrator
SerialNumber              : 55000011faef0fab5ffd7f75b30000000011fa
CertificateTemplate       : ESC1 Template
                            (1.3.6.1.4.1.311.21.8.10395027.10224472.4213181.15714845.1171465.9.10657968.9897558)
RequestDate               : 6/3/2021 5:54:51 PM
StartDate                 : 6/3/2021 5:44:51 PM
EndDate                   : 6/3/2022 5:44:51 PM

CA                        : dc.theshire.local\theshire-DC-CA
RequestID                 : 4603
RequesterName             : THESHIRE\cody
RequesterMachineName      : dev.theshire.local
RequesterProcessName      : Certify.exe
SubjectAltNamesExtension  : Administrator
SubjectAltNamesAttrib     :
SerialNumber              : 55000011fb021b79cf7276c2de0000000011fb
CertificateTemplate       : ESC1 Template
                            (1.3.6.1.4.1.311.21.8.10395027.10224472.4213181.15714845.1171465.9.10657968.9897558)
RequestDate               : 6/3/2021 5:55:10 PM
StartDate                 : 6/3/2021 5:45:10 PM
EndDate                   : 6/3/2022 5:45:10 PM
```

The `SubjectAltNamesExtension` property means that the x509 SubjectAlternativeNames extension was used to specify the SAN, which happens for templates with the `CT_FLAG_ENROLLEE_SUPPLIES_SUBJECT` flag. The `SubjectAltNamesAttrib` property means that x509 name/value pairs were used, which happens when specifying a SAN when the `EDITF_ATTRIBUTESUBJECTALTNAME2` CA flag is set.

Existing issued certificates can be revoked using PSPKI's Revoke-Certificate function:

```
PS C:\> Get-CertificationAuthority <CAName> | Get-IssuedRequest -RequestID <X> | Revoke-Certificate -Reason "KeyCompromise"
```

Applicable values for -Reason are "KeyCompromise", "CACompromise", and "Unspecified".
