AD Smartcard Logon Configuration Cheat Sheet

Purpose of this document is to provide the required information for ensuring conceptual understanding of the technical configuration pre-requisites for smartcard logon in AD and providing insights on troubleshooting common issues to eliminate potential misconfigurations.

General Information Regarding Windows Smartcard Logon

Section Key Concepts:

- Smartcard logon to a Windows environment relies on a digital signature produced by the user activating the private key on their smartcard
- A Domain Controller verifies the user certificate and digital signature in order to facilitate a Kerberos ticket issuance for downstream access
    - DC authentication processes include:
        - verifying the user certificate is issued from a trusted CA,
        - verifying the user certificate is not revoked,
        - extracting an identifier from the certificate to correlate to an AD user account, and
        - digital signature validation (proof of possession of the private key)
- Custom registry configurations within the “Kerberos Distribution Center” (KDC) may be necessary to facilitate successful account mapping

Windows Domain Controllers (DCs) leverage Kerberos protocols and ticketing system to establish sessions and grant access to a user/device. There are several possible authentication mechanisms on Windows, but with a smart card, the Public Key Cryptography in Initial Authentication in Kerberos (PKINIT) extension is used to allow a private key on a smart card to digitally sign a timestamp as part of a pre-authentication process. The Domain Controller validates the signature and user public authentication certificate as part of the authentication request prior to granting a Kerberos ticket.

Note that path discovery and validation completed by the DC may require access to external URLs containing AIA certificate bundles and signed CRLs, or OCSP responses Network or firewall rules that prevent access to these resources could potentially impact authentication. Network administrators may need to be engaged using the following guidance to prevent this unintended unintended failure of revocation checking:

https://www.idmanagement.gov/implement/scl-windows/#step-1---network-ports-and-protocols

Required Trust Store Configurations

Section Key Concepts:

- Smartcard logon to an Active Directory Domain Service (ADDS) is actually a mutual authentication process, where the Domain Controller (DC) not only verifies the client certificate, but the client also verifies the DC via its public key server certificate.
- Both the DC and client systems need to ensure their respective trust stores contain the appropriate root CA certificates (and preferably any intermediate CA certificates) to ensure path discovery and validation is completed successfully and efficiently.
- AD has two “trust stores” that complete slightly different functions but both need to contain the trust chain for the client certificate.
    - The Enterprise Trust store managed via Group Policy Object (GPO) tells the both client machines and domain controller what CAs, and thus their child certificates, can be trusted overarchingly for all transactions
    - The NTauth store specifically lists data about those parent CAs that can be used ONLY for domain logon.

Domain controllers have two “trust stores” containing the CA certificates that can be leveraged for network authentication one of which can also help to push out to domain joined systems for general trust scenarios (outside of smartcard logon). Both of these stores need to be configured to contain the appropriate CA certificates to facilitate smartcard logon.

The first store, specifically the NTAuth store, is not a trust store in the traditional sense (e.g., a directory containing PEM or DER encoded signed certificate files), but is actually a data-only object that lists the CAs that it can trust for subsequent end-entity user and device authentication within the domain. The NTAuth object stores ASN.1 encoded certificate information as a byte-array in a multi-value attribute rather than as the certificate file, as a result, the NTauth store acts more like a relatively simple whitelist identifying root and intermediate CAs that can be trusted for domain login. AD administrators can use the certutil function provided an input of a X.509 certificate file to extract and publish this parsed data about certificates to the directory for caching in domain joined devices.

Specifically, most Federal agencies will want to distribute the Federal Common Policy CA G2 within the NTauth object to ensure their user PIV’s will be recognized for smartcard authentication to their environment. Administrators can use the following series of commands to add the FCPCAG2 root to the NTauth store:

- Publication = certutil -dspublish -f [Path]\fcpcag2.crt, where the path is the location of the root certificate file
- Push = gpupdate /force, this pushes the updated NTauth objects to all devices in the domain or forrest
- Confirm = certutil -viewstore -enterprise, this will display a list of trusted root CA certificates in the NTauth store

More detailed instructions for publishing to the NTauth store using the certutil command can be found here:
https://www.idmanagement.gov/implement/trust-fcpca/#use-microsoft-certutil

Separately, DC group policy objects can be modified, resulting in updates to the “Enterprise Trust Store.” This process is slightly more traditional from the perspective of a “trust store” management function and should be very familiar for anyone who has used the Microsoft Certificate Manager (certmgr.msc or mmc with the certificate snap-in). Once configured, this store containing signed CA certificate files, can be pushed out to all other domain joined workstations and compatible devices. Note that the PKI hierarchy used to sign the DC certificates themselves will need to be pushed to client machines to facilitate smartcard logon, in addition to the PIV trust chains.

Effectively, a DC administrator can use the following functions to view and modify the Enterprise trust store using the following navigation options:

Server manager → Tools → Group Policy Management → Domains (Select the relevant domain) → Group Policy Objects → add a new group policy (with an intuitive name like “trust store” or similar)

Once the new group policy is created you can navigate as follows:

Computer Configuration → Expand Policies → Expand Security Settings → Expand Public Key Policies.

Here you should see a list of folders similar to the standard Windows certificate manager to include some other more extraneous folders regarding other encryption options. The critical folders for enterprise trust store management are as follows:

- Trusted Root Certification Authorities
- Intermediate Certification Authorities
- Untrusted Certificates

An AD administrator can import certificate files (.cer, .crt, .p7b) to each folder as needed to trust PIV certificate chains, and potentially any device certification authorities that are needed by the client to authenticate the DC.

For reference you can download PIV relevant CA certificates from the following consolidated p7b:

https://www.idmanagement.gov/implement/tools/CACertificatesValidatingToFederalCommonPolicyG2.p7b

Within the above consolidated certificate bundle you can find and export any relevant PIV issuing CAs identified in the following list for eventual import to the Enterprise Trust Store:

https://www.idmanagement.gov/fpki/notifications/#active-issuing-ca-certificate-details

Modifications to either the DC or user trust stores can be verified several ways, but the most direct mechanism is to run the local machine certificate manager and view the individual certificate folders.

Run → certlm.msc → ensure the local machine store is selected → open both root and intermediate folders and verify each CA is listed in its respective folder

Domain Controller ADCS Certificate Profile Configurations

Section Key Concepts:

- The device certificate used to identify and authenticate the Domain Controller to users' machines within your domain needs to assert very specific information within its certificate fields to ensure smart card logon works as expected
    - Many federal organizations maintain their own Locally Trusted Active Directory Certificate Services (ADCS) public key infrastructures for credentialing their DCs
    - Some PIV issuing PKIs already have the appropriate certificate templates for Domain Controllers, you may want to reach out to your PKI provider (distinct from a PIV Registration Authority) for details if you do not run your own ADCS PKI and you do not credential your DCs using an internal device PKI
- Changes to Windows server in 2024 and 2025 regarding strong certificate identifiers have facilitated the need for inclusion of a proprietary SID field in ADCS auto-enrolled device certificates, which is recommended for credentialing devices like DCs

As mentioned in the previous section, smart card log on requires Mutual authentication between the client and the domain controller. Regardless of the issuing authority, the public certificate used to credential the domain controller must assert very specific information within its fields in order to ensure smart card log on works as expected.

Those critical values to be included in the DC certificate profile include:

- Key Usage extension - must assert
    - Digital Signature and
    - Key Encipherment
- Enhanced Key Usage (EKU) extension - must assert both
    - Client Authentication - 1.3.6.1.5.5.7.3.2 and
    - Server Authentication - 1.3.6.1.5.5.7.3.1
- Subject Alternative Name extension - must assert
    - the DC's GUID

- The SID field has its own field identifier as follows and will assert a value that is generated within AD specific to that device
    - SID OID - 1.3.6.1.4.1.311.25.2

As of September 2025, it is also recommended to include a security identifier (SID) in the domain controller certificate to facilitate strong identifier enforcement rules within Windows servers. The SID is a Microsoft proprietary identifier and is not necessary for inclusion into other Enterprise level certificates such as PIV related certificates. ADCS administrators can use the following link to assist in populating the SID extension in certificate signing requests for their ADCS domain controllers and other joined devices.

https://github.com/GSA/ficam-scripts-public/tree/master/_SIDcsrNPE

Some ADCS administrators may have multiple active domain controller certificates available within their server's personal store. If modifying or updating DC certificate profiles while in the process of enabling smart card logon, please make sure to validate the active domain controller certificate being presented during logon. You may also want to remove any inactive keys and certificates that assert older profiles that may prevent smart card logon.

Mutual Authentication Checks (certutil with dcinfo and scinfo)

Section Key Concepts:

- Available Microsoft utilities, individually run on a user machine and a DC, can assist with error logging and identification of common smart card logon issues
- Some of the logs can be lengthy and detailed due to full certificate path validation; however, for each 'leaf' on the PKI 'tree' there only needs to be one successful revocation check

System administrators can run specific commands from the client machine and the DC respectively to ensure both are configured correctly and they can each validate the other's credentials.

In order to verify that the appropriate device certificate identifying the DC is being presented to the client machine, you will want to open a command prompt either on the client machine or DC and use the following command (the > file path is optional if you only want to view the results in the prompt terminal):

- certutil -dcinfo [> File Path]

Running this command on the client machine should return information that will uniquely identify the DC server certificate being presented to the client. If this command returns an unexpected or outdated server certificate you will want to remove any unnecessary certificates from the personal store of the domain controller directly using the mmc console with certificate snap-in, and then recheck using the dcinfo command.

Alternatively on both the client machine and DC, you will want to ensure that a representative PIV itself can be validated. The following command can be used to interrogate the card and validate the authentication certificate on the device that is using the card (be it the client machine, or on the DC directly if available).

- Certutil -scinfo [> File Path]

Unique Identifier Mapping Suggestions

Section Key Concepts:

- In order for smartcard logon to work, after path discovery and validation is completed on the user's certificate, AD needs to extract identifying information from the PIV authentication certificate to correlate to the AD user account identifier
- PIV certificates contain several fields that can be used or concatenated into AD identifiers
- In October 2025 Microsoft completed final retirement of compatibility mode for support of what were considered "weak identifiers" many of which were used for PIV AD user account mapping
    - This change to identifiers requires either migration to a strong altsecid mapping, or enablement of a "policy tuple" which is a group policy object for securely leveraging what Microsoft categorizes as less strong identifiers (e.g., IssuerSubject or UPN)
- AD implementers will need to either manually register PIV authentication certificates or request them from issuers (e.g., USAccess) in order to appropriately extract unique identifiers for mapping to established AD accounts

There are generally only two high level options for User certificate to AD account mapping including:

- User Principal Name (UPN) - asserted in the Subject Alternative Name (SAN) extension of most PIV authentication certificates; however, considered a "weak" identifier and requires implementation of a GPO "policy tuple" to work
- altSecurityIdentities (AltSecID) - includes several options to link either one or several concatenated fields from the user public X.509 certificate

A full list of potential AltSecID mappings can be found at the following link: https://www.idmanagement.gov/implement/scl-windows/#option-1-link-the-piv-authentication-certificate

Note that two of the AltSecID mappings that have been **deprecated** and are no longer available include the **Subject** Relative Distinguished Name (RDN which is usually referred to as the “subject name”) and the **RFC822** value in the SAN (this is generally an email address which may or may not correlate with many UPN values).

Other, more unique, mappings from the PIV certificate are considered “strong” and allowable. Similar to the UPN mapping, the AltSecID ‘Issuer and Subject’ mapping is also allowed using the GPO ‘policy tuple’ solution. The other supported mappings listed at the previous hyperlink do not require a policy tuple to work.

In order to add these AltSecId values to the AD account, the PIV authentication certificate does need to be available prior to logon. Administrators can either create a custom registration process to acquire these certificates, or they can work with their PIV or PKI issuers to generate the needed certificate reports and then run a script to extract and populate the appropriate values in the AD user records.

Currently the AltSecId **Issuer and Subject** mapping is specifically recommended as it should be considered unique for the lifetime of the user affiliation with the organization. This option does require the policy tuple GPO modification which can be referenced in the next section. Other acceptable AltSecId fields will change upon rekey of PIV certificates or reissuance of a PIV card.

Administrators can use the following powershell script to potentially update a AD user directory with concatenated IssuerSubject AltSecIds, provided they have the required public PIV authentication certificates of their users:

https://github.com/GSA/ficam-scripts-public/tree/master/_altSecId

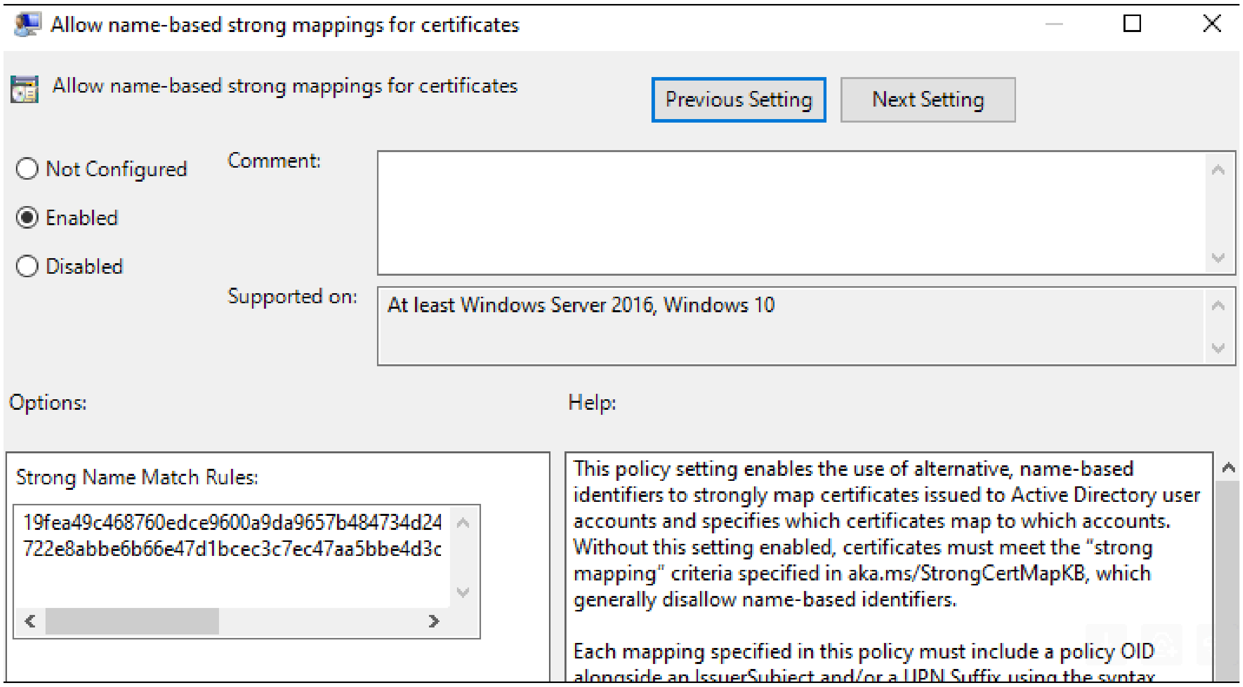
Policy Tuple/Registry Key Modification Recommendations

Section Key Concepts:

- Updates to Windows Server 2019 and later facilitated changes to require DC registry configurations to work with some certificate mappings from external PKI certificates (e.g., PIV)
- ‘Policy Tuples’ defined in GPOs objects are now required if you would like to continue to use either UPN or the AltSecId for IssuerSubject

Policy tuples have been implemented to accommodate some weak identifier mappings in Windows Server as of September 10th, 2024. Tuple mappings are defined via GPO, specifically under Administrative Settings / System / KDC. Within this object administrators can modify an entry titled “Allow name-based strong mappings for certificates” wherein three things must be defined as requisites to allow for weak certificate mappings (e.g., UPN or IssuerSubject AltSecId) to include:

1. Trusted issuing CA - identified by its certificate thumbprint, each tuple will be unique to each issuing CA thumbprint, two common CAs most implementers will want to account for are:
    a. Entrust Managed Services SSP CA (current) - 19fea49c468760edce9600a9da9657b484734d24
    b. Entrust Managed Services SSP CA (maintenance) - 722e8abbe6b66e47d1bcec3c7ec47aa5bbe4d3c5
2. Trusted certificate policy OID - ensures that the client certificate is issued according to a certain policy or policies (generally these are PIV authentication certificate specific), some of the relevant PIV OIDs implementers may want include:
    a. Standard PIV authentication certificate - 2.16.840.1.101.3.2.1.3.13
    b. Derived PIV authentication certificate (software) - 2.16.840.1.101.3.2.1.3.40
    c. Common PIV-I authentication certificate - 2.16.840.1.101.3.2.1.3.45
    d. Common Hardware certificate (like for an alternative token) - 2.16.840.1.101.3.2.1.3.7
3. Name matching - defines what field(s) to extract from the certificate that meets the previous two conditions for correlation to AD user accounts, this will generally be UPN suffix and/or altsecid IssuerSubject as follows:
    a. UPNsuffix=[orgnaization].gov
    b. IssuerSubject



Allow name-based strong mappings for certificates

Allow name-based strong mappings for certificates

○ Not Configured
● Enabled
○ Disabled

Comment:

Supported on: At least Windows Server 2016, Windows 10

Options:

Strong Name Match Rules:

19fea49c468760edce9600a9da9657b484734d24
722e8abbe6b66e47d1bcec3c7ec47aa5bbe4d3c

Help:

This policy setting enables the use of alternative, name-based identifiers to strongly map certificates issued to Active Directory user accounts and specifies which certificates map to which accounts. Without this setting enabled, certificates must meet the “strong mapping” criteria specified in aka.ms/StrongCertMapKB, which generally disallow name-based identifiers.

Each mapping specified in this policy must include a policy OID alongside an IssuerSubject and/or a UPN Suffix using the syntax

The following is an example policy tuple identifying a PIV authentication certificate asserting the HHS.gov suffix as expected in a UPN identifier and issued from the HHS PIV issuing CA.

19fea49c468760edce9600a9da9657b484734d24; 2.16.840.1.101.3.2.1.3.13;
UPNsuffix=hhs.gov

Note that each policy tuple is unique to each issuing CA, whereas you can include multiple OIDs or identifier rules for each tuple separated by commas.

If you are receiving several Error 39 codes in your DC system logs, it most likely indicates that policy tuples may not be properly formatted or the AltSecId population is not functioning as intended.