certutil

Article • 10/19/2023 • 20 contributors

Feedback

In this article

Caution

`Certutil` isn't recommended to be used in any production code and doesn't provide any guarantees of live site support or application compatibilities. It's a tool utilized by developers and IT administrators to view certificate content information on devices.

Certutil.exe is a command-line program installed as part of Certificate Services. You can use certutil.exe to display certification authority (CA) configuration information, configure Certificate Services, and back up and restore CA components. The program also verifies certificates, key pairs, and certificate chains.

If `certutil` is run on a certification authority without other parameters, it displays the current certification authority configuration. If `certutil` is run on a non-certification authority without other parameters, the command defaults to running the `certutil -dump` command. Not all versions of certutil provide all of the parameters and options that this document describes. You can see the choices that

Download PDF

your version of certutil provides by running `certutil -?` or `certutil <parameter> -?`.

> **Tip**
>
> To see complete help for all certutil verbs and options, including ones that are hidden from the `-?` argument, run `certutil -v -uSAGE`. The `uSAGE` switch is case-sensitive.

Parameters

-dump

Dumps the configuration information or files.

```
certutil [options] [-dump]
certutil [options] [-dump] File
```

Options:

```
[-f] [-user] [-Silent] [-split] [-p Password] [-t Timeout]
```

-dumpPFX

Dumps the PFX structure.

```
certutil [options] [-dumpPFX] File
```

Options:

```
[-f] [-Silent] [-split] [-p Password] [-csp Provider]
```

-asn

Parses and displays the contents of a file using Abstract Syntax Notation (ASN.1) syntax. File types include .CER, .DER and PKCS #7 formatted files.

```
certutil [options] -asn File [type]
```

- `[type]`: numeric CRYPT_STRING_* decoding type

-decodehex

Decodes a hexadecimal-encoded file.

```
certutil [options] -decodehex InFile OutFile [type]
```

- `[type]`: numeric CRYPT_STRING_* decoding type

Options:

```
[-f]
```

-encodehex

Encodes a file in hexadecimal.

```
certutil [options] -encodehex InFile OutFile [type]
```

- `[type]`: numeric CRYPT_STRING_* encoding type

Options:

```
[-f] [-nocr] [-nocrlf] [-UnicodeText]
```

-decode

Decodes a Base64-encoded file.

```
certutil [options] -decode InFile OutFile
```

Options:

```
[-f]
```

-encode

Encodes a file to Base64.

```
certutil [options] -encode InFile OutFile
```

Options:

```
[-f] [-unicodetext]
```

-deny

Denies a pending request.

```
certutil [options] -deny RequestId
```

Options:

```
[-config Machine\CAName]
```

-resubmit

Resubmits a pending request.

```
certutil [options] -resubmit RequestId
```

Options:

```
[-config Machine\CAName]
```

-setattributes

Sets attributes for a pending certificate request.

```
certutil [options] -setattributes RequestId AttributeString
```

Where:

- **RequestId** is the numeric Request ID for the pending request.
- **AttributeString** is the request attribute name and value pairs.

Options:

```
[-config Machine\CAName]
```

Remarks

- Names and values must be colon separated, while multiple names and value pairs must be newline separated. For example: `CertificateTemplate:User\nEMail:User@Domain.com` where the `\n` sequence is converted to a newline separator.

-setextension

Set an extension for a pending certificate request.

```
certutil [options] -setextension RequestId ExtensionName Flags {Long |
```

Where:

- **requestID** is the numeric Request ID for the pending request.
- **ExtensionName** is the ObjectId string for the extension.
- **Flags** sets the priority of the extension. `0` is recommended, while `1` sets the extension to critical, `2` disables the extension, and `3` does both.

Options:

```
[-config Machine\CAName]
```

Remarks

- If the last parameter is numeric, it's taken as a **Long**.
- If the last parameter can be parsed as a date, it's taken as a **Date**.
- If the last parameter starts with `\@`, the rest of the token is taken as the filename with binary data or an ascii-text hex dump.
- If the last parameter is anything else, it's taken as a String.

-revoke

Revokes a certificate.

```
certutil [options] -revoke SerialNumber [Reason]
```

Where:

- **SerialNumber** is a comma-separated list of certificate serial numbers to revoke.
- **Reason** is the numeric or symbolic representation of the revocation reason, including:
  - **0. CRL_REASON_UNSPECIFIED** - Unspecified (default)
  - **1. CRL_REASON_KEY_COMPROMISE** - Key compromise
  - **2. CRL_REASON_CA_COMPROMISE** - Certificate Authority compromise

- **3. CRL_REASON_AFFILIATION_CHANGED** - Affiliation changed
- **4. CRL_REASON_SUPERSEDED** - Superseded
- **5. CRL_REASON_CESSATION_OF_OPERATION** - Cessation of operation
- **6. CRL_REASON_CERTIFICATE_HOLD** - Certificate hold
- **8. CRL_REASON_REMOVE_FROM_CRL** - Remove from CRL
- **9: CRL_REASON_PRIVILEGE_WITHDRAWN** - Privilege withdrawn
- **10: CRL_REASON_AA_COMPROMISE** - AA compromise
- **-1. Unrevoke** - Unrevokes

Options:

```
[-config Machine\CAName]
```

-isvalid

Displays the disposition of the current certificate.

```
certutil [options] -isvalid SerialNumber | CertHash
```

Options:

```
[-config Machine\CAName]
```

-getconfig

Gets the default configuration string.

```
certutil [options] -getconfig
```

Options:

```
[-idispatch] [-config Machine\CAName]
```

-getconfig2

Gets the default configuration string via ICertGetConfig.

```
certutil [options] -getconfig2
```

Options:

```
[-idispatch]
```

-getconfig3

Gets configuration via ICertConfig.

```
certutil [options] -getconfig3
```

Options:

```
[-idispatch]
```

-ping

Attempts to contact the Active Directory Certificate Services Request interface.

```
certutil [options] -ping [MaxSecondsToWait | CAMachineList]
```

Where:

- **CAMachineList** is a comma-separated list of CA machine names. For a single machine, use a terminating comma. This option also displays the site cost for each CA machine.

Options:

```
[-config Machine\CAName] [-Anonymous] [-Kerberos] [-ClientCertificate C
```

-pingadmin

Attempts to contact the Active Directory Certificate Services Admin interface.

```
certutil [options] -pingadmin
```

Options:

```
[-config Machine\CAName]
```

-CAInfo

Displays information about the certification authority.

```
certutil [options] -CAInfo [InfoName [Index | ErrorCode]]
```

Where:

- **InfoName** indicates the CA property to display, based on the following infoname argument syntax:
  - ***** - Displays all properties
  - **ads** - Advanced Server
  - **aia [Index]** - AIA URLs
  - **cdp [Index]** - CDP URLs
  - **cert [Index]** - CA cert
  - **certchain [Index]** - CA cert chain
  - **certcount** - CA cert count
  - **certcrlchain [Index]** - CA cert chain with CRLs
  - **certstate [Index]** - CA cert
  - **certstatuscode [Index]** - CA cert verify status
  - **certversion [Index]** - CA cert version
  - **CRL [Index]** - Base CRL

  - **crlstate [Index]** - CRL
  - **crlstatus [Index]** - CRL Publish Status
  - **cross- [Index]** - Backward cross cert
  - **cross+ [Index]** - Forward cross cert
  - **crossstate- [Index]** - Backward cross cert
  - **crossstate+ [Index]** - Forward cross cert
  - **deltacrl [Index]** - Delta CRL
  - **deltacrlstatus [Index]** - Delta CRL Publish Status
  - **dns** - DNS Name
  - **dsname** - Sanitized CA short name (DS name)
  - **error1 ErrorCode** - Error message text
  - **error2 ErrorCode** - Error message text and error code
  - **exit [Index]** - Exit module description
  - **exitcount** - Exit module count
  - **file** - File version
  - **info** - CA info
  - **kra [Index]** - KRA cert
  - **kracount** - KRA cert count
  - **krastate [Index]** - KRA cert
  - **kraused** - KRA cert used count
  - **localename** - CA locale name
  - **name** - CA name
  - **ocsp [Index]** - OCSP URLs
  - **parent** - Parent CA
  - **policy** - Policy module description
  - **product** - Product version
  - **propidmax** - Maximum CA PropId
  - **role** - Role Separation
  - **sanitizedname** - Sanitized CA name
  - **sharedfolder** - Shared folder
  - **subjecttemplateoids** - Subject Template OIDs
  - **templates** - Templates
  - **type** - CA type
  - **xchg [Index]** - CA exchange cert
  - **xchgchain [Index]** - CA exchange cert chain
  - **xchgcount** - CA exchange cert count
  - **xchgcrlchain [Index]** - CA exchange cert chain with CRLs
- **index** is the optional zero-based property index.
- **errorcode** is the numeric error code.

Options:

```
[-f] [-split] [-config Machine\CAName]
```

-CAPropInfo

Displays CA Property Type information.

```
certutil [options] -CAInfo [InfoName [Index | ErrorCode]]
```

Options:

```
[-idispatch] [-v1] [-admin] [-config Machine\CAName]
```

-ca.cert

Retrieves the certificate for the certification authority.

```
certutil [options] -ca.cert OutCACertFile [Index]
```

Where:

- **OutCACertFile** is the output file.
- **Index** is the CA certificate renewal index (defaults to most recent).

Options:

```
[-f] [-split] [-config Machine\CAName]
```

-ca.chain

Retrieves the certificate chain for the certification authority.

```
certutil [options] -ca.chain OutCACertChainFile [Index]
```

Where:

- **OutCACertChainFile** is the output file.
- **Index** is the CA certificate renewal index (defaults to most recent).

Options:

```

[-f] [-split] [-config Machine\CAName]
```

-GetCRL

Gets a certificate revocation list (CRL).

```

certutil [options] -GetCRL OutFile [Index] [delta]
```

Where:

- **Index** is the CRL index or key index (defaults to CRL for most recent key).
- **delta** is the delta CRL (default is base CRL).

Options:

```

[-f] [-split] [-config Machine\CAName]
```

-CRL

Publishes new certificate revocation lists (CRLs) or delta CRLs.

```

certutil [options] -CRL [dd:hh | republish] [delta]
```

Where:

- **dd:hh** is the new CRL validity period in days and hours.
- **republish** republishes the most recent CRLs.

- **delta** publishes the delta CRLs only (default is base and delta CRLs).

Options:

```
[-split] [-config Machine\CAName]
```

-shutdown

Shuts down the Active Directory Certificate Services.

```
certutil [options] -shutdown
```

Options:

```
[-config Machine\CAName]
```

-installCert

Installs a certification authority certificate.

```
certutil [options] -installCert [CACertFile]
```

Options:

```
[-f] [-silent] [-config Machine\CAName]
```

-renewCert

Renews a certification authority certificate.

```
```

```
certutil [options] -renewCert [ReuseKeys] [Machine\ParentCAName]
```

Options:

```

[-f] [-silent] [-config Machine\CAName]
```

- Use `-f` to ignore an outstanding renewal request, and to generate a new request.

-schema

Dumps the schema for the certificate.

```

certutil [options] -schema [Ext | Attrib | CRL]
```

Where:

- The command defaults to the Request and Certificate table.
- **Ext** is the extension table.
- **Attribute** is the attribute table.
- **CRL** is the CRL table.

Options:

```

[-split] [-config Machine\CAName]
```

-view

Dumps the certificate view.

```

certutil [options] -view [Queue | Log | LogFail | Revoked | Ext | Attri
```

Where:

- **Queue** dumps a specific request queue.

- **Log** dumps the issued or revoked certificates, plus any failed requests.
- **LogFail** dumps the failed requests.
- **Revoked** dumps the revoked certificates.
- **Ext** dumps the extension table.
- **Attrib** dumps the attribute table.
- **CRL** dumps the CRL table.
- **csv** provides the output using comma-separated values.

Options:

```
[-silent] [-split] [-config Machine\CAName] [-restrict RestrictionList]
```

Remarks

- To display the **StatusCode** column for all entries, type `-out StatusCode`
- To display all columns for the last entry, type: `-restrict RequestId==$`
- To display the **RequestId** and **Disposition** for three requests, type: `-restrict requestID>=37,requestID<40 -out requestID,disposition`
- To display Row IDs **Row IDs** and **CRL numbers** for all Base CRLs, type: `-restrict crlminbase=0 -out crlrowID,crlnumber crl`
- To display Base CRL number 3, type: `-v -restrict crlminbase=0,crlnumber=3 -out crlrawcrl crl`
- To display the entire CRL table, type: `CRL`
- Use `Date[+|-dd:hh]` for date restrictions.
- Use `now+dd:hh` for a date relative to the current time.
- Templates contain Extended Key Usages (EKUs), which are object identifiers (OIDs) that describe how the certificate is used. Certificates don't always include template common names or display names, but they always contain the template EKUs. You can extract the EKUs for a specific certificate template from Active Directory and then restrict views based on that extension.

-db

Dumps the raw database.

```
certutil [options] -db
```

Options:

```
[-config Machine\CAName] [-restrict RestrictionList] [-out ColumnList]
```

-deleterow

Deletes a row from the server database.

```
certutil [options] -deleterow RowId | Date [Request | Cert | Ext | Attr
```

Where:

- **Request** deletes the failed and pending requests, based on submission date.
- **Cert** deletes the expired and revoked certificates, based on expiration date.
- **Ext** deletes the extension table.
- **Attrib** deletes the attribute table.
- **CRL** deletes the CRL table.

Options:

```
[-f] [-config Machine\CAName]
```

Examples

- To delete failed and pending requests submitted by January 22, 2001, type: `1/22/2001 request`
- To delete all certificates that expired by January 22, 2001, type: `1/22/2001 cert`
- To delete the certificate row, attributes, and extensions for RequestID 37, type: `37`
- To delete CRLs that expired by January 22, 2001, type: `1/22/2001 crl`

> ⓘ **Note**
>
> **Date** expects the format `mm/dd/yyyy` rather then `dd/mm/yyyy`, for example `1/22/2001` rather than `22/1/2001` for January 22, 2001. If your server isn't

configured with US regional settings, using the **Date** argument might produce unexpected results.

-backup

Backs up the Active Directory Certificate Services.

```
certutil [options] -backup BackupDirectory [Incremental] [KeepLog]
```

Where:

- **BackupDirectory** is the directory to store the backed up data.
- **Incremental** performs an incremental backup only (default is full backup).
- **KeepLog** preserves the database log files (default is to truncate log files).

Options:

```
[-f] [-config Machine\CAName] [-p Password] [-ProtectTo SAMNameAndSIDLi
```

-backupDB

Backs up the Active Directory Certificate Services database.

```
certutil [options] -backupdb BackupDirectory [Incremental] [KeepLog]
```

Where:

- **BackupDirectory** is the directory to store the backed up database files.
- **Incremental** performs an incremental backup only (default is full backup).
- **KeepLog** preserves the database log files (default is to truncate log files).

Options:

```
[-f] [-config Machine\CAName]
```

-backupkey

Backs up the Active Directory Certificate Services certificate and private key.

```
certutil [options] -backupkey BackupDirectory
```

Where:

- **BackupDirectory** is the directory to store the backed up PFX file.

Options:

```
[-f] [-config Machine\CAName] [-p password] [-ProtectTo SAMNameAndSIDLi
```

-restore

Restores the Active Directory Certificate Services.

```
certutil [options] -restore BackupDirectory
```

Where:

- **BackupDirectory** is the directory containing the data to be restored.

Options:

```
[-f] [-config Machine\CAName] [-p password]
```

-restoredb

Restores the Active Directory Certificate Services database.

```
certutil [options] -restoredb BackupDirectory
```

Where:

- **BackupDirectory** is the directory containing the database files to be restored.

Options:

```
[-f] [-config Machine\CAName]
```

-restorekey

Restores the Active Directory Certificate Services certificate and private key.

```
certutil [options] -restorekey BackupDirectory | PFXFile
```

Where:

- **BackupDirectory** is the directory containing PFX file to be restored.
- **PFXFile** is the PFX file to be restored.

Options:

```
[-f] [-config Machine\CAName] [-p password]
```

-exportPFX

Exports the certificates and private keys. For more information, see the `-store` parameter in this article.

```
certutil [options] -exportPFX [CertificateStoreName] CertId PFXFile [Mo
```

Where:

- **CertificateStoreName** is the name of the certificate store.
- **CertId** is the certificate or CRL match token.
- **PFXFile** is the PFX file to be exported.

- **Modifiers** are the comma-separated list, which can include one or more of the following:
  - **CryptoAlgorithm=** specifies the cryptographic algorithm to use for encrypting the PFX file, such as `TripleDES-Sha1` or `Aes256-Sha256`.
  - **EncryptCert** - Encrypts the private key associated with the certificate with a password.
  - **ExportParameters** -Exports the private key parameters in addition to the certificate and private key.
  - **ExtendedProperties** - Includes all extended properties associated with the certificate in the output file.
  - **NoEncryptCert** - Exports the private key without encrypting it.
  - **NoChain** - Doesn't import the certificate chain.
  - **NoRoot** - Doesn't import the root certificate.

-importPFX

Imports the certificates and private keys. For more information, see the `-store` parameter in this article.

```
certutil [options] -importPFX [CertificateStoreName] PFXFile [Modifiers
```

Where:

- **CertificateStoreName** is the name of the certificate store.
- **PFXFile** is the PFX file to be imported.
- **Modifiers** are the comma-separated list, which can include one or more of the following:
  - **AT_KEYEXCHANGE** - Changes the keyspec to key exchange.
  - **AT_SIGNATURE** - Changes the keyspec to signature.
  - **ExportEncrypted** - Exports the private key associated with the certificate with password encryption.
  - **FriendlyName=** - Specifies a friendly name for the imported certificate.
  - **KeyDescription=** - Specifies a description for the private key associated with the imported certificate.
  - **KeyFriendlyName=** - Specifies a friendly name for the private key associated with the imported certificate.
  - **NoCert** - Doesn't import the certificate.
  - **NoChain** - Doesn't import the certificate chain.
  - **NoExport** - Makes the private key non-exportable.
  - **NoProtect** - Doesn't password protect keys by using a password.

- **NoRoot** - Doesn't import the root certificate.
- **Pkcs8** - Uses PKCS8 format for the private key in the PFX file.
- **Protect** - Protects keys by using a password.
- **ProtectHigh** - Specifies that a high-security password must be associated with the private key.
- **VSM** - Stores the private key associated with the imported certificate in the Virtual Smart Card (VSC) container.

Options:

```
[-f] [-Enterprise] [-user] [-GroupPolicy] [-Silent] [-p Password] [-csp
```

Remarks

- Defaults to personal machine store.

-dynamicfilelist

Displays a dynamic file list.

```
certutil [options] -dynamicfilelist
```

Options:

```
[-config Machine\CAName]
```

-databaselocations

Displays database locations.

```
certutil [options] -databaselocations
```

Options:

```
[-config Machine\CAName]
```

-hashfile

Generates and displays a cryptographic hash over a file.

```
certutil [options] -hashfile InFile [HashAlgorithm]
```

-store

Dumps the certificate store.

```
certutil [options] -store [CertificateStoreName [CertId [OutputFile]]]
```

Where:

- **CertificateStoreName** is the certificate store name. For example:
  - `My, CA (default), Root,`
  - `ldap:///CN=Certification Authorities,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?cACertificate?one?objectClass=certificationAuthority (View Root Certificates)`
  - `ldap:///CN=CAName,CN=Certification Authorities,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?cACertificate?base?objectClass=certificationAuthority (Modify Root Certificates)`
  - `ldap:///CN=CAName,CN=MachineName,CN=CDP,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?certificateRevocationList?base?objectClass=cRLDistributionPoint (View CRLs)`
  - `ldap:///CN=NTAuthCertificates,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?cACertificate?base?objectClass=certificationAuthority (Enterprise CA Certificates)`

  - `ldap: (AD computer object certificates)`
  - `-user ldap: (AD user object certificates)`

- **CertId** is the certificate or CRL match token. This ID can be a:
  - Serial number
  - SHA-1 certificate
  - CRL, CTL or public key hash
  - Numeric cert index (0, 1, and so on)
  - Numeric CRL index (.0, .1, and so on)
  - Numeric CTL index (..0, ..1, and so on)
  - Public key
  - Signature or extension ObjectId
  - Certificate subject Common Name
  - E-mail address
  - UPN or DNS name
  - Key container name or CSP name
  - Template name or ObjectId
  - EKU or Application Policies ObjectId
  - CRL issuer Common Name.

Many of these identifiers might result in multiple matches.

- **OutputFile** is the file used to save the matching certificates.

Options:

```
[-f] [-Enterprise] [-user] [-GroupPolicy] [-Silent] [-split] [-dc DCNam
```

- The `-user` option accesses a user store instead of a machine store.
- The `-enterprise` option accesses a machine enterprise store.
- The `-service` option accesses a machine service store.
- The `-grouppolicy` option accesses a machine group policy store.

For example:

- `-enterprise NTAuth`
- `-enterprise Root 37`
- `-user My 26e0aaaf000000000004`
- `CA .11`

> **Note**

Performance issues are observed when using the `-store` parameter given these two aspects:

1. When the number of certificates in the store exceeds 10.
2. When a **CertId** is specified, it's used to match all the listed types for every certificate. For example, if a **serial number** is provided, it will also attempt to match all other listed types.

If you are concerned about performance issues, PowerShell commands are recommended where it will only match the specified certificate type.

-enumstore

Enumerates the certificate stores.

```
certutil [options] -enumstore [\\MachineName]
```

Where:

- **MachineName** is the remote machine name.

Options:

```
[-enterprise] [-user] [-grouppolicy]
```

-addstore

Adds a certificate to the store. For more information, see the `-store` parameter in this article.

```
certutil [options] -addstore CertificateStoreName InFile
```

Where:

- **CertificateStoreName** is the certificate store name.
- **InFile** is the certificate or CRL file you want to add to the store.

Options:

```
[-f] [-Enterprise] [-user] [-GroupPolicy] [-dc DCName]
```

-delstore

Deletes a certificate from the store. For more information, see the `-store` parameter in this article.

```
certutil [options] -delstore CertificateStoreName certID
```

Where:

- **CertificateStoreName** is the certificate store name.
- **CertId** is the certificate or CRL match token.

Options:

```
[-f] [-Enterprise] [-user] [-GroupPolicy] [-Silent] [-dc DCName]
```

-verifystore

Verifies a certificate in the store. For more information, see the `-store` parameter in this article.

```
certutil [options] -verifystore CertificateStoreName [CertId]
```

Where:

- **CertificateStoreName** is the certificate store name.
- **CertId** is the certificate or CRL match token.

Options:

```
[-Enterprise] [-user] [-GroupPolicy] [-Silent] [-split] [-dc DCName] [-
```

-repairstore

Repairs a key association or update certificate properties or the key security descriptor. For more information, see the `-store` parameter in this article.

```
certutil [options] -repairstore CertificateStoreName CertIdList [Proper
```

Where:

- **CertificateStoreName** is the certificate store name.
- **CertIdList** is the comma-separated list of certificate or CRL match tokens. For more information, see the `-store` CertId description in this article.
- **PropertyInfFile** is the INF file containing external properties, including:

```
[Properties]
    19 = Empty ; Add archived property, OR:
    19 =       ; Remove archived property

    11 = {text}Friendly Name ; Add friendly name property

    127 = {hex} ; Add custom hexadecimal property
        _continue_ = 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0
        _continue_ = 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1

    2 = {text} ; Add Key Provider Information property
      _continue_ = Container=Container Name&
      _continue_ = Provider=Microsoft Strong Cryptographic Provider
      _continue_ = ProviderType=1&
      _continue_ = Flags=0&
      _continue_ = KeySpec=2

    9 = {text} ; Add Enhanced Key Usage property
      _continue_ = 1.3.6.1.5.5.7.3.2,
      _continue_ = 1.3.6.1.5.5.7.3.1,
```

Options:

```

```

```
  [-f] [-Enterprise] [-user] [-GroupPolicy] [-Silent] [-split] [-csp Prov
```

-viewstore

Dumps the certificates store. For more information, see the `-store` parameter in this article.

```

certutil [options] -viewstore [CertificateStoreName [CertId [OutputFile
```

Where:

- **CertificateStoreName** is the certificate store name. For example:
  - `My, CA (default), Root,`
  - `ldap:///CN=Certification Authorities,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?cACertificate?one?objectClass=certificationAuthority (View Root Certificates)`
  - `ldap:///CN=CAName,CN=Certification Authorities,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?cACertificate?base?objectClass=certificationAuthority (Modify Root Certificates)`
  - `ldap:///CN=CAName,CN=MachineName,CN=CDP,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?certificateRevocationList?base?objectClass=cRLDistributionPoint (View CRLs)`
  - `ldap:///CN=NTAuthCertificates,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?cACertificate?base?objectClass=certificationAuthority (Enterprise CA Certificates)`
  - `ldap: (AD computer object certificates)`
  - `-user ldap: (AD user object certificates)`
- **CertId** is the certificate or CRL match token. This can be a:
  - Serial number
  - SHA-1 certificate
  - CRL, CTL or public key hash
  - Numeric cert index (0, 1, and so on)
  - Numeric CRL index (.0, .1, and so on)

- Numeric CTL index (..0, ..1, and so on)
- Public key
- Signature or extension ObjectId
- Certificate subject Common Name
- E-mail address
- UPN or DNS name
- Key container name or CSP name
- Template name or ObjectId
- EKU or Application Policies ObjectId
- CRL issuer Common Name.

Many of these may result in multiple matches.

- **OutputFile** is the file used to save the matching certificates.

Options:

```
[-f] [-Enterprise] [-user] [-GroupPolicy] [-dc DCName]
```

- The `-user` option accesses a user store instead of a machine store.
- The `-enterprise` option accesses a machine enterprise store.
- The `-service` option accesses a machine service store.
- The `-grouppolicy` option accesses a machine group policy store.

For example:

- `-enterprise NTAuth`
- `-enterprise Root 37`
- `-user My 26e0aaaf000000000004`
- `CA .11`

-viewdelstore

Deletes a certificate from the store.

```
certutil [options] -viewdelstore [CertificateStoreName [CertId [OutputF
```

Where:

- **CertificateStoreName** is the certificate store name. For example:

- `My, CA (default), Root,`
- `ldap:///CN=Certification Authorities,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?cACertificate?one?objectClass=certificationAuthority (View Root Certificates)`
- `ldap:///CN=CAName,CN=Certification Authorities,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?cACertificate?base?objectClass=certificationAuthority (Modify Root Certificates)`
- `ldap:///CN=CAName,CN=MachineName,CN=CDP,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?certificateRevocationList?base?objectClass=cRLDistributionPoint (View CRLs)`
- `ldap:///CN=NTAuthCertificates,CN=Public Key Services,CN=Services,CN=Configuration,DC=cpandl,DC=com?cACertificate?base?objectClass=certificationAuthority (Enterprise CA Certificates)`
- `ldap: (AD computer object certificates)`
- `-user ldap: (AD user object certificates)`

- **CertId** is the certificate or CRL match token. This can be a:
  - Serial number
  - SHA-1 certificate
  - CRL, CTL or public key hash
  - Numeric cert index (0, 1, and so on)
  - Numeric CRL index (.0, .1, and so on)
  - Numeric CTL index (..0, ..1, and so on)
  - Public key
  - Signature or extension ObjectId
  - Certificate subject Common Name
  - E-mail address
  - UPN or DNS name
  - Key container name or CSP name
  - Template name or ObjectId
  - EKU or Application Policies ObjectId
  - CRL issuer Common Name. Many of these may result in multiple matches.

- **OutputFile** is the file used to save the matching certificates.

Options:

```
[-f] [-Enterprise] [-user] [-GroupPolicy] [-dc DCName]
```

- The `-user` option accesses a user store instead of a machine store.
- The `-enterprise` option accesses a machine enterprise store.
- The `-service` option accesses a machine service store.
- The `-grouppolicy` option accesses a machine group policy store.

For example:

- `-enterprise NTAuth`
- `-enterprise Root 37`
- `-user My 26e0aaaf000000000004`
- `CA .11`

-UI

Invokes the certutil interface.

```
certutil [options] -UI File [import]
```

-TPMInfo

Displays Trusted Platform Module Information.

```
certutil [options] -TPMInfo
```

Options:

```
[-f] [-Silent] [-split]
```

-attest

Specifies that the certificate request file should be attested.

```
certutil [options] -attest RequestFile
```

Options:

```
[-user] [-Silent] [-split]
```

-getcert

Selects a certificate from a selection UI.

```
certutil [options] [ObjectId | ERA | KRA [CommonName]]
```

Options:

```
[-Silent] [-split]
```

-ds

Displays directory service (DS) distinguished names (DNs).

```
certutil [options] -ds [CommonName]
```

Options:

```
[-f] [-user] [-split] [-dc DCName]
```

-dsDel

Deletes DS DNs.

```
certutil [options] -dsDel [CommonName]
```

Options:

```
[-user] [-split] [-dc DCName]
```

-dsPublish

Publishes a certificate or certificate revocation list (CRL) to Active Directory.

```
certutil [options] -dspublish CertFile [NTAuthCA | RootCA | SubCA | Cro
certutil [options] -dspublish CRLfile [DSCDPContainer [DSCDPCN]]
```

Where:

- **CertFile** is the name of the certificate file to publish.
- **NTAuthCA** publishes the certificate to the DS Enterprise store.
- **RootCA** publishes the certificate to the DS Trusted Root store.
- **SubCA** publishes the CA certificate to the DS CA object.
- **CrossCA** publishes the cross-certificate to the DS CA object.
- **KRA** publishes the certificate to the DS Key Recovery Agent object.
- **User** publishes the certificate to the User DS object.
- **Machine** publishes the certificate to the Machine DS object.
- **CRLfile** is the name of the CRL file to publish.
- **DSCDPContainer** is the DS CDP container CN, usually the CA machine name.
- **DSCDPCN** is the DS CDP object CN based on the sanitized CA short name and key index.

Options:

```
[-f] [-user] [-dc DCName]
```

- Use `-f` to create a new DS object.

-dsCert

Displays DS certificates.

```
certutil [options] -dsCert [FullDSDN] | [CertId [OutFile]]
```

Options:

```
[-Enterprise] [-user] [-config Machine\CAName] [-dc DCName]
```

-dsCRL

Displays DS CRLs.

```
certutil [options] -dsCRL [FullDSDN] | [CRLIndex [OutFile]]
```

Options:

```
[-idispatch] [-Enterprise] [-user] [-config Machine\CAName] [-dc DCName
```

-dsDeltaCRL

Displays DS delta CRLs.

```
certutil [options] -dsDeltaCRL [FullDSDN] | [CRLIndex [OutFile]]
```

Options:

```
[-Enterprise] [-user] [-config Machine\CAName] [-dc DCName]
```

-dsTemplate

Displays DS template attributes.

```
certutil [options] -dsTemplate [Template]
```

Options:

```
[Silent] [-dc DCName]
```

-dsAddTemplate

Adds DS templates.

```
certutil [options] -dsAddTemplate TemplateInfFile
```

Options:

```
[-dc DCName]
```

-ADTemplate

Displays Active Directory templates.

```
certutil [options] -ADTemplate [Template]
```

Options:

```
[-f] [-user] [-ut] [-mt] [-dc DCName]
```

-Template

Displays the certificate enrollment policy templates.

Options:

```
certutil [options] -Template [Template]
```

Options:

```
[-f] [-user] [-Silent] [-PolicyServer URLOrId] [-Anonymous] [-Kerberos]
```

-TemplateCAs

Displays the certification authorities (CAs) for a certificate template.

```
certutil [options] -TemplateCAs Template
```

Options:

```
[-f] [-user] [-dc DCName]
```

-CATemplates

Displays templates for the Certificate Authority.

```
certutil [options] -CATemplates [Template]
```

Options:

```

```

```
[-f] [-user] [-ut] [-mt] [-config Machine\CAName] [-dc DCName]
```

-SetCATemplates

Sets the certificate templates that the Certificate Authority can issue.

```
certutil [options] -SetCATemplates [+ | -] TemplateList
```

Where:

- The `+` sign adds certificate templates to the CA's available template list.
- The `-` sign removes certificate templates from the CA's available template list.

-SetCASites

Manages site names, including setting, verifying, and deleting Certificate Authority site names.

```
certutil [options] -SetCASites [set] [SiteName]
certutil [options] -SetCASites verify [SiteName]
certutil [options] -SetCASites delete
```

Where:

- **SiteName** is allowed only when targeting a single Certificate Authority.

Options:

```
[-f] [-config Machine\CAName] [-dc DCName]
```

Remarks

- The `-config` option targets a single Certificate Authority (default is all CAs).
- The `-f` option can be used to override validation errors for the specified **SiteName** or to delete all CA site names.

> **Note**
>
> For more information about configuring CAs for Active Directory Domain Services (AD DS) site awareness, see AD DS Site Awareness for AD CS and PKI clients.

-enrollmentServerURL

Displays, adds, or deletes enrollment server URLs associated with a CA.

```
certutil [options] -enrollmentServerURL [URL AuthenticationType [Priori
certutil [options] -enrollmentserverURL URL delete
```

Where:

- **AuthenticationType** specifies one of the following client authentication methods while adding a URL:
  - **Kerberos** - Use Kerberos SSL credentials.
  - **UserName** - Use a named account for SSL credentials.
  - **ClientCertificate** - Use X.509 Certificate SSL credentials.
  - **Anonymous** - Use anonymous SSL credentials.
- **delete** deletes the specified URL associated with the CA.
- **Priority** defaults to `1` if not specified when adding a URL.
- **Modifiers** is a comma-separated list, which includes one or more of the following:
  - **AllowRenewalsOnly** only renewal requests can be submitted to this CA via this URL.
  - **AllowKeyBasedRenewal** allows use of a certificate that has no associated account in the AD. This applies only with **ClientCertificate** and **AllowRenewalsOnly** mode.

Options:

```
[-config Machine\CAName] [-dc DCName]
```

-ADCA

Displays the Active Directory Certificate Authorities.

```
certutil [options] -ADCA [CAName]
```

Options:

```
[-f] [-split] [-dc DCName]
```

-CA

Displays the enrollment policy Certificate Authorities.

```
certutil [options] -CA [CAName | TemplateName]
```

Options:

```
[-f] [-user] [-Silent] [-split] [-PolicyServer URLOrId] [-Anonymous] [-
```

-Policy

Displays the enrollment policy.

```
certutil [options] -Policy
```

Options:

```
[-f] [-user] [-Silent] [-split] [-PolicyServer URLOrId] [-Anonymous] [-
```

-PolicyCache

Displays or deletes enrollment policy cache entries.

```
certutil [options] -PolicyCache [delete]
```

Where:

- **delete** deletes the policy server cache entries.
- **-f** deletes all cache entries

Options:

```
[-f] [-user] [-policyserver URLorID]
```

-CredStore

Displays, adds, or deletes Credential Store entries.

```
certutil [options] -CredStore [URL]
certutil [options] -CredStore URL add
certutil [options] -CredStore URL delete
```

Where:

- **URL** is the target URL. You can also use `*` to match all entries or `https://machine*` to match a URL prefix.
- **add** adds a credential store entry. Using this option also requires the use of SSL credentials.
- **delete** deletes credential store entries.
- **-f** overwrites a single entry or deletes multiple entries.

Options:

```
[-f] [-user] [-Silent] [-Anonymous] [-Kerberos] [-ClientCertificate Cli
```

-InstallDefaultTemplates

Installs the default certificate templates.

```
certutil [options] -InstallDefaultTemplates
```

Options:

```
[-dc DCName]
```

-URL

Verifies certificate or CRL URLs.

```
certutil [options] -URL InFile | URL
```

Options:

```
[-f] [-split]
```

-URLCache

Displays or deletes URL cache entries.

```
certutil [options] -URLcache [URL | CRL | * [delete]]
```

Where:

- **URL** is the cached URL.
- **CRL** runs on all cached CRL URLs only.
- ***** operates on all cached URLs.
- **delete** deletes relevant URLs from the current user's local cache.
- **-f** forces fetching a specific URL and updating the cache.

Options:

```
[-f] [-split]
```

-pulse

Pulses an autoenrollment event or NGC task.

```
certutil [options] -pulse [TaskName [SRKThumbprint]]
```

Where:

- **TaskName** is the task to trigger.
  - **Pregen** is the NGC Key pregen task.
  - **AIKEnroll** is the NGC AIK certificate enrollment task. (Defaults to the autoenrollment event).
- **SRKThumbprint** is the thumbprint of the Storage Root Key
- **Modifiers**:
  - Pregen
  - PregenDelay
  - AIKEnroll
  - CryptoPolicy
  - NgcPregenKey
  - DIMSRoam

Options:

```
[-user]
```

-MachineInfo

Displays information about the Active Directory machine object.

```
certutil [options] -MachineInfo DomainName\MachineName$
```

-DCInfo

Displays information about the domain controller. The default displays DC certificates without verification.

```
certutil [options] -DCInfo [Domain] [Verify | DeleteBad | DeleteAll]
```

- **Modifiers**:
  - Verify
  - DeleteBad
  - DeleteAll

Options:

```
[-f] [-user] [-urlfetch] [-dc DCName] [-t Timeout]
```

> **Tip**
>
> The ability to specify an Active Directory Domain Services (AD DS) domain **[Domain]** and to specify a domain controller (**-dc**) was added in Windows Server 2012. To successfully run the command, you must use an account that is a member of **Domain Admins** or **Enterprise Admins**. The behavior modifications of this command are as follows:
>
> - If a domain is not specified and a specific domain controller is not specified, this option returns a list of domain controllers to process from the default domain controller.
> - If a domain is not specified, but a domain controller is specified, a report of the certificates on the specified domain controller is generated.
> - If a domain is specified, but a domain controller is not specified, a list of domain controllers is generated along with reports on the certificates for each domain controller in the list.
> - If the domain and domain controller are specified, a list of domain controllers is generated from the targeted domain controller. A report of the certificates for each domain controller in the list is also generated.
>
> For example, assume there is a domain named CPANDL with a domain controller named CPANDL-DC1. You can run the following command to a

retrieve a list of domain controllers and their certificates from CPANDL-DC1: `certutil -dc cpandl-dc1 -DCInfo cpandl`.

-EntInfo

Displays information about an enterprise Certificate Authority.

```
certutil [options] -EntInfo DomainName\MachineName$
```

Options:

```
[-f] [-user]
```

-TCAInfo

Displays information about the Certificate Authority.

```
certutil [options] -TCAInfo [DomainDN | -]
```

Options:

```
[-f] [-Enterprise] [-user] [-urlfetch] [-dc DCName] [-t Timeout]
```

-SCInfo

Displays information about the smart card.

```
certutil [options] -scinfo [ReaderName [CRYPT_DELETEKEYSET]]
```

Where:

- **CRYPT_DELETEKEYSET** deletes all keys on the smart card.

Options:

```
[-Silent] [-split] [-urlfetch] [-t Timeout]
```

-SCRoots

Manages smart card root certificates.

```
certutil [options] -SCRoots update [+][InputRootFile] [ReaderName]
certutil [options] -SCRoots save @OutputRootFile [ReaderName]
certutil [options] -SCRoots view [InputRootFile | ReaderName]
certutil [options] -SCRoots delete [ReaderName]
```

Options:

```
[-f] [-split] [-p Password]
```

-key

Lists the keys stored in a key container.

```
certutil [options] -key [KeyContainerName | -]
```

Where:

- **KeyContainerName** is the key container name for the key to verify. This option defaults to machine keys. To switch to user keys, use `-user`.
- Using the `-` sign refers to using the default key container.

Options:

```
[-user] [-Silent] [-split] [-csp Provider] [-Location AlternateStorageL
```

-delkey

Deletes the named key container.

```
certutil [options] -delkey KeyContainerName
```

Options:

```
[-user] [-Silent] [-split] [-csp Provider] [-Location AlternateStorageL
```

-DeleteHelloContainer

Deletes the Windows Hello container, removing all associated credentials that are stored on the device, including any WebAuthn and FIDO credentials.

Users need to sign out after using this option for it to complete.

```
certutil [options] -DeleteHelloContainer
```

-verifykeys

Verifies a public or private key set.

```
certutil [options] -verifykeys [KeyContainerName CACertFile]
```

Where:

- **KeyContainerName** is the key container name for the key to verify. This option defaults to machine keys. To switch to user keys, use `-user`.
- **CACertFile** signs or encrypts certificate files.

Options:

```
[-f] [-user] [-Silent] [-config Machine\CAName]
```

Remarks

- If no arguments are specified, each signing CA certificate is verified against its private key.
- This operation can only be performed against a local CA or local keys.

-verify

Verifies a certificate, certificate revocation list (CRL), or certificate chain.

```
certutil [options] -verify CertFile [ApplicationPolicyList | - [Issuanc
certutil [options] -verify CertFile [CACertFile [CrossedCACertFile]]
certutil [options] -verify CRLFile CACertFile [IssuedCertFile]
certutil [options] -verify CRLFile CACertFile [DeltaCRLFile]
```

Where:

- **CertFile** is the name of the certificate to verify.
- **ApplicationPolicyList** is the optional comma-separated list of required Application Policy ObjectIds.
- **IssuancePolicyList** is the optional comma-separated list of required Issuance Policy ObjectIds.
- **CACertFile** is the optional issuing CA certificate to verify against.
- **CrossedCACertFile** is the optional certificate cross-certified by **CertFile**.
- **CRLFile** is the CRL file used to verify the **CACertFile**.
- **IssuedCertFile** is the optional issued certificate covered by the CRLfile.
- **DeltaCRLFile** is the optional delta CRL file.
- **Modifiers**:
  - Strong - Strong signature verification
  - MSRoot - Must chain to a Microsoft root
  - MSTestRoot - Must chain to a Microsoft test root
  - AppRoot - Must chain to a Microsoft application root
  - EV - Enforce Extended Validation Policy

Options:

```
[-f] [-Enterprise] [-user] [-Silent] [-split] [-urlfetch] [-t Timeout]
```

Remarks

- Using **ApplicationPolicyList** restricts chain building to only chains valid for the specified Application Policies.
- Using **IssuancePolicyList** restricts chain building to only chains valid for the specified Issuance Policies.
- Using **CACertFile** verifies the fields in the file against **CertFile** or **CRLfile**.
- If **CACertFile** isn't specified, the full chain is built and verified against **CertFile**.
- If **CACertFile** and **CrossedCACertFile** are both specified, the fields in both files are verified against **CertFile**.
- Using **IssuedCertFile** verifies the fields in the file against **CRLfile**.
- Using **DeltaCRLFile** verifies the fields in the file against **CertFile**.

-verifyCTL

Verifies the AuthRoot or Disallowed Certificates CTL.

```
certutil [options] -verifyCTL CTLobject [CertDir] [CertFile]
```

Where:

- **CTLObject** identifies the CTL to verify, including:
  - **AuthRootWU** reads the AuthRoot CAB and matching certificates from the URL cache. Use `-f` to download from Windows Update instead.
  - **DisallowedWU** reads the Disallowed Certificates CAB and disallowed certificate store file from the URL cache. Use `-f` to download from Windows Update instead.
    - **PinRulesWU** reads the PinRules CAB from the URL cache. Use `-f` to download from Windows Update instead.
  - **AuthRoot** reads the registry-cached AuthRoot CTL. Use with `-f` and an untrusted **CertFile** to force the registry cached **AuthRoot** and Disallowed Certificate CTLs to update.
  - **Disallowed** reads the registry-cached Disallowed Certificates CTL. Use with `-f` and an untrusted **CertFile** to force the registry cached **AuthRoot** and Disallowed Certificate CTLs to update.
    - **PinRules** reads the registry cached PinRules CTL. Using `-f` has the same behavior as with **PinRulesWU**.

    - **CTLFileName** specifies the file or http path to the CTL or CAB file.

- **CertDir** specifies the folder containing certificates matching the CTL entries. Defaults to the same folder or website as the **CTLobject**. Using an http folder path requires a path separator at the end. If you don't specify **AuthRoot** or **Disallowed**, multiple locations are searched for matching certificates, including local certificate stores, crypt32.dll resources and the local URL cache. Use `-f` to download from Windows Update, as needed.

- **CertFile** specifies the certificate(s) to verify. Certificates are matched against CTL entries, displaying the results. This option suppresses most of the default output.

Options:

```
[-f] [-user] [-split]
```

-syncWithWU

Syncs certificates with Windows Update.

```
certutil [options] -syncWithWU DestinationDir
```

Where:

- **DestinationDir** is the specified directory.
- **f** forces an overwrite.
- **Unicode** writes redirected output in Unicode.
- **gmt** displays times as GMT.
- **seconds** displays times with seconds and milliseconds.
- **v** is a verbose operation.
- **PIN** is the Smart Card PIN.
- **WELL_KNOWN_SID_TYPE** is a numeric SID:
    - 22 - Local System
    - 23 - Local Service
    - 24 - Network Service

Remarks

The following files are downloaded by using the automatic update mechanism:

- *authrootstl.cab* contains the CTLs of non-Microsoft root certificates.
- *disallowedcertstl.cab* contains the CTLs of untrusted certificates.
- *disallowedcert.sst* contains the serialized certificate store, including the untrusted certificates.
- *thumbprint.crt* contains the non-Microsoft root certificates.

For example, `certutil -syncWithWU \\server1\PKI\CTLs`.

- If you use a nonexistent local path or folder as the destination folder, you see the error: `The system can't find the file specified. 0x80070002 (WIN32: 2 ERROR_FILE_NOT_FOUND)`
- If you use a nonexistent or unavailable network location as the destination folder, you see the error: `The network name can't be found. 0x80070043 (WIN32: 67 ERROR_BAD_NET_NAME)`
- If your server can't connect over TCP port 80 to Microsoft Automatic Update servers, you receive the following error: `A connection with the server couldn't be established 0x80072efd (INet: 12029 ERROR_INTERNET_CANNOT_CONNECT)`
- If your server is unable to reach the Microsoft Automatic Update servers with the DNS name `ctldl.windowsupdate.com`, you receive the following error: `The server name or address couldn't be resolved 0x80072ee7 (INet: 12007 ERROR_INTERNET_NAME_NOT_RESOLVED).`
- If you don't use the `-f` switch, and any of the CTL files already exist in the directory, you receive a file exists error: `certutil: -syncWithWU command FAILED: 0x800700b7 (WIN32/HTTP: 183 ERROR_ALREADY_EXISTS) Certutil: Can't create a file when that file already exists.`
- If there's a change in the trusted root certificates, you see: `Warning! Encountered the following no longer trusted roots: <folder path>\ <thumbprint>.crt. Use "-f" option to force the delete of the above ".crt" files. Was "authrootstl.cab" updated? If yes, consider deferring the delete until all clients have been updated.`

Options:

```
[-f] [-Unicode] [-gmt] [-seconds] [-v] [-privatekey] [-pin PIN] [-sid W
```

-generateSSTFromWU

Generates a store file that is synced with Windows Update.

```
certutil [options] -generateSSTFromWU SSTFile
```

Where:

- **SSTFile** is the `.sst` file to be generated that contains the Third Party Roots downloaded from Windows Update.

Options:

```
[-f] [-split]
```

-generatePinRulesCTL

Generates a Certificate Trust List (CTL) file that contains a list of pinning rules.

```
certutil [options] -generatePinRulesCTL XMLFile CTLFile [SSTFile [Query
```

Where:

- **XMLFile** is the input XML file to be parsed.
- **CTLFile** is the output CTL file to be generated.
- **SSTFile** is the optional *.sst* file to be created that contains all of the certificates used for pinning.
- **QueryFilesPrefix** are optional *Domains.csv* and *Keys.csv* files to be created for database query.
  - The **QueryFilesPrefix** string is prepended to each created file.
  - The **Domains.csv** file contains rule name, domain rows.
  - The **Keys.csv** file contains rule name, key SHA256 thumbprint rows.

Options:

```
[-f]
```

-downloadOcsp

Downloads the OCSP responses and writes to the directory.

```
certutil [options] -downloadOcsp CertificateDir OcspDir [ThreadCount] [
```

Where:

- **CertificateDir** is the directory of a certificate, store and PFX files.
- **OcspDir** is the directory to write OCSP responses.
- **ThreadCount** is the optional maximum number of threads for concurrent downloading. Default is **10**.
- **Modifiers** are comma separated list of one or more of the following:
  - **DownloadOnce** - Downloads once and exits.
  - **ReadOcsp** - Reads from OcspDir instead of writing.

-generateHpkpHeader

Generates the HPKP header using certificates in a specified file or directory.

```
certutil [options] -generateHpkpHeader CertFileOrDir MaxAge [ReportUri]
```

Where:

- **CertFileOrDir** is the file or directory of certificates, which is the source of pin-sha256.
- **MaxAge** is the max-age value in seconds.
- **ReportUri** is the optional report-uri.
- **Modifiers** are comma separated list of one or more of the following:
  - **includeSubDomains** - Appends the includeSubDomains.

-flushCache

Flushes the specified caches in selected process, such as, lsass.exe.

```
certutil [options] -flushCache ProcessId CacheMask [Modifiers]
```

Where:

- **ProcessId** is the numeric ID of a process to flush. Set to **0** to flush all processes where flush is enabled.
- **CacheMask** is the bit mask of caches to be flushed either numeric or the following bits:
  - **0**: ShowOnly
  - **0x01**: CERT_WNF_FLUSH_CACHE_REVOCATION
  - **0x02**: CERT_WNF_FLUSH_CACHE_OFFLINE_URL
  - **0x04**: CERT_WNF_FLUSH_CACHE_MACHINE_CHAIN_ENGINE
  - **0x08**: CERT_WNF_FLUSH_CACHE_USER_CHAIN_ENGINES
  - **0x10**: CERT_WNF_FLUSH_CACHE_SERIAL_CHAIN_CERTS
  - **0x20**: CERT_WNF_FLUSH_CACHE_SSL_TIME_CERTS
  - **0x40**: CERT_WNF_FLUSH_CACHE_OCSP_STAPLING
- **Modifiers** are comma separated list of one or more of the following:
  - **Show** - Shows the caches being flushed. Certutil must be explicitly terminated.

-addEccCurve

Adds an ECC Curve.

```
certutil [options] -addEccCurve [CurveClass:]CurveName CurveParameters
```

Where:

- **CurveClass** is the ECC Curve Class type:
  - WEIERSTRASS (Default)
  - MONTGOMERY
  - TWISTED_EDWARDS
- **CurveName** is the ECC Curve name.
- **CurveParameters** are one of the following:
  - A certificate filename containing ASN encoded parameters.
  - A file containing ASN encoded parameters.

- **CurveOID** is the ECC Curve OID and is one of the following:
  - A certificate filename containing an ASN encoded OID.
  - An explicit ECC Curve OID.
- **CurveType** is the Schannel ECC NamedCurve point (numeric).

Options:

```
[-f]
```

-deleteEccCurve

Deletes the ECC Curve.

```
certutil [options] -deleteEccCurve CurveName | CurveOID
```

Where:

- **CurveName** is the ECC Curve name.
- **CurveOID** is the ECC Curve OID.

Options:

```
[-f]
```

-displayEccCurve

Displays the ECC Curve.

```
certutil [options] -displayEccCurve [CurveName | CurveOID]
```

Where:

- **CurveName** is the ECC Curve name.
- **CurveOID** is the ECC Curve OID.

Options:

```
[-f]
```

-csplist

Lists the cryptographic service providers (CSPs) installed on this machine for cryptographic operations.

```
certutil [options] -csplist [Algorithm]
```

Options:

```
[-user] [-Silent] [-csp Provider]
```

-csptest

Tests the CSPs installed on this machine.

```
certutil [options] -csptest [Algorithm]
```

Options:

```
[-user] [-Silent] [-csp Provider]
```

-CNGConfig

Displays CNG cryptographic configuration on this machine.

```
certutil [options] -CNGConfig
```

Options:

```
[-Silent]
```

-sign

Re-signs a certificate revocation list (CRL) or certificate.

```
certutil [options] -sign InFileList | SerialNumber | CRL OutFileList [S
certutil [options] -sign InFileList | SerialNumber | CRL OutFileList [#
certutil [options] -sign InFileList OutFileList [Subject:CN=...] [Issue
```

Where:

- **InFileList** is the comma-separated list of certificate or CRL files to modify and re-sign.
- **SerialNumber** is the serial number of the certificate to create. The validity period and other options can't be present.
- **CRL** creates an empty CRL. The validity period and other options can't be present.
- **OutFileList** is the comma-separated list of modified certificate or CRL output files. The number of files must match infilelist.
- **StartDate+dd:hh** is the new validity period for the certificate or CRL files, including:
  - optional date plus
  - optional days and hours validity period If multiple fields are used, use a (+) or (-) separator. Use `now[+dd:hh]` to start at the current time. Use `now-dd:hh+dd:hh` to start at a fixed offset from the current time and a fixed validity period. Use `never` to have no expiration date (for CRLs only).
- **SerialNumberList** is the comma-separated serial number list of the files to add or remove.
- **ObjectIdList** is the comma-separated extension ObjectId list of the files to remove.

- **@ExtensionFile** is the INF file that contains the extensions to update or remove. For example:

```
[Extensions]
    2.5.29.31 = ; Remove CRL Distribution Points extension
    2.5.29.15 = {hex} ; Update Key Usage extension
    _continue_=03 02 01 86
```

- **HashAlgorithm** is the name of the hash algorithm. This must only be the text preceded by the `#` sign.
- **AlternateSignatureAlgorithm** is the alternate signature algorithm specifier.

Options:

```
[-nullsign] [-f] [-user] [-Silent] [-Cert CertId] [-csp Provider]
```

Remarks

- Using the minus sign (-) removes serial numbers and extensions.
- Using the plus sign (+) adds serial numbers to a CRL.
- You can use a list to remove both serial numbers and **ObjectIds** from a CRL at the same time.
- Using the minus sign before **AlternateSignatureAlgorithm** allows you to use the legacy signature format.
- Using the plus sign allows you to use the alternate signature format.
- If you don't specify **AlternateSignatureAlgorithm**, the signature format in the certificate or CRL is used.

-vroot

Creates or deletes web virtual roots and file shares.

```
certutil [options] -vroot [delete]
```

-vocsproot

Creates or deletes web virtual roots for an OCSP web proxy.

```
certutil [options] -vocsproot [delete]
```

-addEnrollmentServer

Adds an Enrollment Server application and application pool if necessary for the specified Certificate Authority. This command doesn't install binaries or packages.

```
certutil [options] -addEnrollmentServer Kerberos | UserName | ClientCer
```

Where:

- **addEnrollmentServer** requires you to use an authentication method for the client connection to the Certificate Enrollment Server, including:
  - **Kerberos** uses Kerberos SSL credentials.
  - **UserName** uses named account for SSL credentials.
  - **ClientCertificate** uses X.509 Certificate SSL credentials.
- **Modifiers**:
  - **AllowRenewalsOnly** allows only renewal request submissions to the Certificate Authority through the URL.
  - **AllowKeyBasedRenewal** allows use of a certificate with no associated account in Active Directory. This applies when used with **ClientCertificate** and **AllowRenewalsOnly** mode.

Options:

```
[-config Machine\CAName]
```

-deleteEnrollmentServer

Deletes an Enrollment Server application and application pool if necessary for the specified Certificate Authority. This command doesn't install binaries or packages.

```
```

```
certutil [options] -deleteEnrollmentServer Kerberos | UserName | Client
```

Where:

- **deleteEnrollmentServer** requires you to use an authentication method for the client connection to the Certificate Enrollment Server, including:
  - **Kerberos** uses Kerberos SSL credentials.
  - **UserName** uses named account for SSL credentials.
  - **ClientCertificate** uses X.509 Certificate SSL credentials.

Options:

```
[-config Machine\CAName]
```

-addPolicyServer

Add a Policy Server application and application pool, if necessary. This command doesn't install binaries or packages.

```
certutil [options] -addPolicyServer Kerberos | UserName | ClientCertifi
```

Where:

- **addPolicyServer** requires you to use an authentication method for the client connection to the Certificate Policy Server, including:
  - **Kerberos** uses Kerberos SSL credentials.
  - **UserName** uses named account for SSL credentials.
  - **ClientCertificate** uses X.509 Certificate SSL credentials.
- **KeyBasedRenewal** allows use of policies returned to the client containing keybasedrenewal templates. This option applies only for **UserName** and **ClientCertificate** authentication.

-deletePolicyServer

Deletes a Policy Server application and application pool, if necessary. This command doesn't remove binaries or packages.

```
```

```
certutil [options] -deletePolicyServer Kerberos | UserName | ClientCert
```

Where:

- **deletePolicyServer** requires you to use an authentication method for the client connection to the Certificate Policy Server, including:
  - **Kerberos** uses Kerberos SSL credentials.
  - **UserName** uses named account for SSL credentials.
  - **ClientCertificate** uses X.509 Certificate SSL credentials.
- **KeyBasedRenewal** allows use of a KeyBasedRenewal policy server.

-Class

Displays COM registry information.

```
certutil [options] -Class [ClassId | ProgId | DllName | *]
```

Options:

```
[-f]
```

-7f

Checks certificate for 0x7f length encodings.

```
certutil [options] -7f CertFile
```

-oid

Displays the object identifier or sets a display name.

```
certutil [options] -oid ObjectId [DisplayName | delete [LanguageId [typ
certutil [options] -oid GroupId
certutil [options] -oid AlgId | AlgorithmName [GroupId]
```

Where:

- **ObjectId** is the ID to be displayed or to add to the display name.
- **GroupId** is the GroupID number (decimal) that ObjectIds enumerate.
- **AlgId** is the hexadecimal ID that objectID looks up.
- **AlgorithmName** is the algorithm name that objectID looks up.
- **DisplayName** displays the name to store in DS.
- **Delete** deletes the display name.
- **LanguageId** is the language ID value (defaults to current: 1033).
- **Type** is the type of DS object to create, including:
  - `1` - Template (default)
  - `2` - Issuance Policy
  - `3` - Application Policy
- `-f` creates a DS object.

Options:

```
[-f]
```

-error

Displays the message text associated with an error code.

```
certutil [options] -error ErrorCode
```

-getsmtpinfo

Gets Simple Mail Transfer Protocol (SMTP) information.

```
certutil [options] -getsmtpinfo
```

-setsmtpinfo

Sets SMTP information.

```
certutil [options] -setsmtpinfo LogonName
```

Options:

```
[-config Machine\CAName] [-p Password]
```

-getreg

Displays a registry value.

```
certutil [options] -getreg [{ca | restore | policy | exit | template |
```

Where:

- **ca** uses a Certificate Authority's registry key.
- **restore** uses Certificate Authority's restore registry key.
- **policy** uses the policy module's registry key.
- **exit** uses the first exit module's registry key.
- **template** uses the template registry key (use `-user` for user templates).
- **enroll** uses the enrollment registry key (use `-user` for user context).
- **chain** uses the chain configuration registry key.
- **PolicyServers** uses the Policy Servers registry key.
- **ProgId** uses the policy or exit module's ProgID (registry subkey name).
- **RegistryValueName** uses the registry value name (use `Name*` to prefix match).
- **value** uses the new numeric, string or date registry value or filename. If a numeric value starts with `+` or `-`, the bits specified in the new value are set or cleared in the existing registry value.

Options:

```
[-f] [-Enterprise] [-user] [-GroupPolicy] [-config Machine\CAName]
```

Remarks

- If a string value starts with `+` or `-`, and the existing value is a `REG_MULTI_SZ` value, the string is added to or removed from the existing registry value. To force creation of a `REG_MULTI_SZ` value, add `\n` to the end of the string value.
- If the value starts with `\@`, the rest of the value is the name of the file containing the hexadecimal text representation of a binary value.
- If it doesn't refer to a valid file, it's instead parsed as `[Date][+|-][dd:hh]` which is an optional date plus or minus optional days and hours.
- If both are specified, use a plus sign (+) or minus sign (-) separator. Use `now+dd:hh` for a date relative to the current time.
- Use `i64` as a suffix to create a REG_QWORD value.
- Use `chain\chaincacheresyncfiletime @now` to effectively flush cached CRLs.
- Registry aliases:
  - Config
  - CA
  - Policy - PolicyModules
  - Exit - ExitModules
  - Restore - RestoreInProgress
  - Template - Software\Microsoft\Cryptography\CertificateTemplateCache
  - Enroll - Software\Microsoft\Cryptography\AutoEnrollment (Software\Policies\Microsoft\Cryptography\AutoEnrollment)
  - MSCEP - Software\Microsoft\Cryptography\MSCEP
  - Chain - Software\Microsoft\Cryptography\OID\EncodingType 0\CertDllCreateCertificateChainEngine\Config
  - PolicyServers - Software\Microsoft\Cryptography\PolicyServers (Software\Policies\Microsoft\Cryptography\PolicyServers)
  - Crypt32 - System\CurrentControlSet\Services\crypt32
  - NGC - System\CurrentControlSet\Control\Cryptography\Ngc
  - AutoUpdate - Software\Microsoft\SystemCertificates\AuthRoot\AutoUpdate
  - Passport - Software\Policies\Microsoft\PassportForWork
  - MDM - Software\Microsoft\Policies\PassportForWork

-setreg

Sets a registry value.

```
certutil [options] -setreg [{ca | restore | policy | exit | template |
```

Where:

- **ca** uses a Certificate Authority's registry key.
- **restore** uses Certificate Authority's restore registry key.
- **policy** uses the policy module's registry key.
- **exit** uses the first exit module's registry key.
- **template** uses the template registry key (use `-user` for user templates).
- **enroll** uses the enrollment registry key (use `-user` for user context).
- **chain** uses the chain configuration registry key.
- **PolicyServers** uses the Policy Servers registry key.
- **ProgId** uses the policy or exit module's ProgID (registry subkey name).
- **RegistryValueName** uses the registry value name (use `Name*` to prefix match).
- **Value** uses the new numeric, string or date registry value or filename. If a numeric value starts with `+` or `-`, the bits specified in the new value are set or cleared in the existing registry value.

Options:

```
[-f] [-Enterprise] [-user] [-GroupPolicy] [-config Machine\CAName]
```

Remarks

- If a string value starts with `+` or `-`, and the existing value is a `REG_MULTI_SZ` value, the string is added to or removed from the existing registry value. To force creation of a `REG_MULTI_SZ` value, add `\n` to the end of the string value.
- If the value starts with `\@`, the rest of the value is the name of the file containing the hexadecimal text representation of a binary value.
- If it doesn't refer to a valid file, it's instead parsed as `[Date][+|-][dd:hh]` which is an optional date plus or minus optional days and hours.
- If both are specified, use a plus sign (+) or minus sign (-) separator. Use `now+dd:hh` for a date relative to the current time.
- Use `i64` as a suffix to create a REG_QWORD value.
- Use `chain\chaincacheresyncfiletime @now` to effectively flush cached CRLs.

-delreg

Deletes a registry value.

```
certutil [options] -delreg [{ca | restore | policy | exit | template |
```

Where:

- **ca** uses a Certificate Authority's registry key.
- **restore** uses Certificate Authority's restore registry key.
- **policy** uses the policy module's registry key.
- **exit** uses the first exit module's registry key.
- **template** uses the template registry key (use `-user` for user templates).
- **enroll** uses the enrollment registry key (use `-user` for user context).
- **chain** uses the chain configuration registry key.
- **PolicyServers** uses the Policy Servers registry key.
- **ProgId** uses the policy or exit module's ProgID (registry subkey name).
- **RegistryValueName** uses the registry value name (use `Name*` to prefix match).
- **Value** uses the new numeric, string or date registry value or filename. If a numeric value starts with `+` or `-`, the bits specified in the new value are set or cleared in the existing registry value.

Options:

```
[-f] [-Enterprise] [-user] [-GroupPolicy] [-config Machine\CAName]
```

Remarks

- If a string value starts with `+` or `-`, and the existing value is a `REG_MULTI_SZ` value, the string is added to or removed from the existing registry value. To force creation of a `REG_MULTI_SZ` value, add `\n` to the end of the string value.
- If the value starts with `\@`, the rest of the value is the name of the file containing the hexadecimal text representation of a binary value.
- If it doesn't refer to a valid file, it's instead parsed as `[Date][+|-][dd:hh]` which is an optional date plus or minus optional days and hours.
- If both are specified, use a plus sign (+) or minus sign (-) separator. Use `now+dd:hh` for a date relative to the current time.
- Use `i64` as a suffix to create a REG_QWORD value.
- Use `chain\chaincacheresyncfiletime @now` to effectively flush cached CRLs.
- Registry aliases:
  - Config
  - CA
  - Policy - PolicyModules
  - Exit - ExitModules
  - Restore - RestoreInProgress

- Template - Software\Microsoft\Cryptography\CertificateTemplateCache
- Enroll - Software\Microsoft\Cryptography\AutoEnrollment (Software\Policies\Microsoft\Cryptography\AutoEnrollment)
- MSCEP - Software\Microsoft\Cryptography\MSCEP
- Chain - Software\Microsoft\Cryptography\OID\EncodingType 0\CertDllCreateCertificateChainEngine\Config
- PolicyServers - Software\Microsoft\Cryptography\PolicyServers (Software\Policies\Microsoft\Cryptography\PolicyServers)
- Crypt32 - System\CurrentControlSet\Services\crypt32
- NGC - System\CurrentControlSet\Control\Cryptography\Ngc
- AutoUpdate - Software\Microsoft\SystemCertificates\AuthRoot\AutoUpdate
- Passport - Software\Policies\Microsoft\PassportForWork
- MDM - Software\Microsoft\Policies\PassportForWork

-importKMS

Imports user keys and certificates into the server database for key archival.

```
certutil [options] -importKMS UserKeyAndCertFile [CertId]
```

Where:

- **UserKeyAndCertFile** is a data file with user private keys and certificates that are to be archived. This file can be:
  - An Exchange Key Management Server (KMS) export file.
  - A PFX file.
- **CertId** is a KMS export file decryption certificate match token. For more information, see the `-store` parameter in this article.
- `-f` imports certificates not issued by the Certificate Authority.

Options:

```
[-f] [-Silent] [-split] [-config Machine\CAName] [-p Password] [-symkey
```

-ImportCert

Imports a certificate file into the database.

```
certutil [options] -ImportCert Certfile [ExistingRow]
```

Where:

- **ExistingRow** imports the certificate in place of a pending request for the same key.
- `-f` imports certificates not issued by the Certificate Authority.

Options:

```
[-f] [-config Machine\CAName]
```

Remarks

The Certificate Authority may also need to be configured to support foreign certificates by running `certutil -setreg ca\KRAFlags +KRAF_ENABLEFOREIGN`.

-GetKey

Retrieves an archived private key recovery blob, generates a recovery script, or recovers archived keys.

```
certutil [options] -GetKey SearchToken [RecoveryBlobOutFile]
certutil [options] -GetKey SearchToken script OutputScriptFile
certutil [options] -GetKey SearchToken retrieve | recover OutputFileBas
```

Where:

- **script** generates a script to retrieve and recover keys (default behavior if multiple matching recovery candidates are found, or if the output file isn't specified).
- **retrieve** retrieves one or more Key Recovery Blobs (default behavior if exactly one matching recovery candidate is found, and if the output file is specified). Using this option truncates any extension and appends the certificate-specific string and the `.rec` extension for each key recovery blob. Each file contains a certificate chain and an associated private key, still encrypted to one or more Key Recovery Agent certificates.

- **recover** retrieves and recovers private keys in one step (requires Key Recovery Agent certificates and private keys). Using this option truncates any extension and appends the `.p12` extension. Each file contains the recovered certificate chains and associated private keys, stored as a PFX file.
- **SearchToken** selects the keys and certificates to be recovered, including:
  - Certificate Common Name
  - Certificate Serial Number
  - Certificate SHA-1 hash (thumbprint)
  - Certificate KeyId SHA-1 hash (Subject Key Identifier)
  - Requester Name (domain\user)
  - UPN (user@domain)
- **RecoveryBlobOutFile** outputs a file with a certificate chain and an associated private key, still encrypted to one or more Key Recovery Agent certificates.
- **OutputScriptFile** outputs a file with a batch script to retrieve and recover private keys.
- **OutputFileBaseName** outputs a file base name.

Options:

```
[-f] [-UnicodeText] [-Silent] [-config Machine\CAName] [-p Password] [-
```

Remarks

- For **retrieve**, any extension is truncated and a certificate-specific string and the `.rec` extensions are appended for each key recovery blob. Each file contains a certificate chain and an associated private key, still encrypted to one or more Key Recovery Agent certificates.
- For **recover**, any extension is truncated and the `.p12` extension is appended. Contains the recovered certificate chains and associated private keys, stored as a PFX file.

-RecoverKey

Recovers an archived private key.

```
certutil [options] -RecoverKey RecoveryBlobInFile [PFXOutFile [Recipien
```

Options:

```
[-f] [-user] [-Silent] [-split] [-p Password] [-ProtectTo SAMNameAndSID
```

-mergePFX

Merges PFX files.

```
certutil [options] -MergePFX PFXInFileList PFXOutFile [Modifiers]
```

Where:

- **PFXInFileList** is a comma-separated list of PFX input files.
- **PFXOutFile** is the name of the PFX output file.
- **Modifiers** are comma separated lists of one or more of the following:
  - **ExtendedProperties** includes any extended properties.
  - **NoEncryptCert** specifies to not encrypt the certificates.
  - **EncryptCert** specifies to encrypt the certificates.

Options:

```
[-f] [-user] [-split] [-p password] [-ProtectTo SAMNameAndSIDList] [-cs
```

Remarks

- The password specified on the command line must be a comma-separated password list.
- If more than one password is specified, the last password is used for the output file. If only one password is provided or if the last password is `*`, the user is prompted for the output file password.

-convertEPF

Converts a PFX file into an EPF file.

```
```

```
certutil [options] -ConvertEPF PFXInFileList EPFOutFile [cast | cast-]
```

Where:

- **PFXInFileList** is a comma-separated list of PFX input files.
- **EPFOutFile** is the name of the PFX output file.
- **EPF** is the name of the EPF output file.
- **cast** uses CAST 64 encryption.
- **cast-** uses CAST 64 encryption (export).
- **V3CACertId** is the V3 CA certificate match token. For more information, see the `-store` parameter in this article.
- **Salt** is the EPF output file salt string.

Options:

```
[-f] [-Silent] [-split] [-dc DCName] [-p Password] [-csp Provider]
```

Remarks

- The password specified on the command line must be a comma-separated password list.
- If more than one password is specified, the last password is used for the output file. If only one password is provided or if the last password is `*`, the user is prompted for the output file password.

-add-chain

Adds a certificate chain.

```
certutil [options] -add-chain LogId certificate OutFile
```

Options:

```
[-f]
```

-add-pre-chain

Adds a pre-certificate chain.

```
certutil [options] -add-pre-chain LogId pre-certificate OutFile
```

Options:

```
[-f]
```

-get-sth

Gets a signed tree head.

```
certutil [options] -get-sth [LogId]
```

Options:

```
[-f]
```

-get-sth-consistency

Gets signed tree head changes.

```
certutil [options] -get-sth-consistency LogId TreeSize1 TreeSize2
```

Options:

```
[-f]
```

-get-proof-by-hash

Gets proof of a hash from a timestamp server.

```
certutil [options] -get-proof-by-hash LogId Hash [TreeSize]
```

Options:

```
[-f]
```

-get-entries

Retrieves entries from an event log.

```
certutil [options] -get-entries LogId FirstIndex LastIndex
```

Options:

```
[-f]
```

-get-roots

Retrieves the root certificates from the certificate store.

```
certutil [options] -get-roots LogId
```

Options:

```
[-f]
```

-get-entry-and-proof

Retrieves an event log entry and its cryptographic proof.

```
certutil [options] -get-entry-and-proof LogId Index [TreeSize]
```

Options:

```
[-f]
```

-VerifyCT

Verifies a certificate against the Certificate Transparency log.

```
certutil [options] -VerifyCT Certificate SCT [precert]
```

Options:

```
[-f]
```

-?

Displays the list of parameters.

```
certutil -?
certutil <name_of_parameter> -?
certutil -? -v
```

Where:

- **-?** displays the list of parameters
- **-<name_of_parameter> -?** displays help content for the specified parameter.
- **-? -v** displays a verbose list of parameters and options.

Options

This section defines all of the options you're able to specify, based on the command. Each parameter includes information about which options are valid for use.

Expand table

Option	Description
-admin	Use ICertAdmin2 for CA properties.
-anonymous	Use anonymous SSL credentials.
-cert CertId	Signing certificate.
-clientcertificate clientCertId	Use X.509 Certificate SSL credentials. For selection UI, use `-clientcertificate`.
-config Machine\CAName	Certificate Authority and computer name string.
-csp provider	Provider: **KSP** - Microsoft Software Key Storage Provider **TPM** - Microsoft Platform Crypto Provider **NGC** - Microsoft Passport Key Storage Provider **SC** - Microsoft Smart Card Key Storage Provider
-dc DCName	Target a specific Domain Controller.
-enterprise	Use the local machine enterprise registry certificate store.
-f	Force overwrite.
-generateSSTFromWU SSTFile	Generate SST by using the automatic update mechanism.
-gmt	Display times using GMT.
-GroupPolicy	Use the group policy certificate store.
-idispatch	Use IDispatch instead of COM native methods.
-kerberos	Use Kerberos SSL credentials.
-location alternatestoragelocation	`(-loc)` AlternateStorageLocation.
-mt	Display machine templates.
-nocr	Encode text without CR characters.
-nocrlf	Encode text without CR-LF characters.

-nullsign	Use the hash of the data as a signature.
-oldpfx	Use old PFX encryption.
-out columnlist	Comma-separated column list.
-p password	Password
-pin PIN	Smart card PIN.
-policyserver URLorID	Policy Server URL or ID. For selection U/I, use `-policyserver`. For all Policy Servers, use `-policyserver *`
-privatekey	Display password and private key data.
-protect	Protect keys with password.
-protectto SAMnameandSIDlist	Comma-separated SAM name/SID list.
-restrict restrictionlist	Comma-separated Restriction List. Each restriction consists of a column name, a relational operator, and a constant integer, string, or date. One column name may be preceded by a plus or minus sign to indicate the sort order. For example: `requestID = 47`, `+requestername >= a, requestername`, or `-requestername > DOMAIN, Disposition = 21`.
-reverse	Reverse Log and Queue columns.
-seconds	Display times using seconds and milliseconds.
-service	Use service certificate store.
-sid	Numeric SID: **22** - Local System **23** - Local Service **24** - Network Service
-silent	Use the `silent` flag to acquire crypt context.
-split	Split embedded ASN.1 elements, and save to files.
-sslpolicy servername	SSL Policy matching ServerName.
-symkeyalg symmetrickeyalgorithm[,keylength]	Name of the Symmetric Key Algorithm with optional key length. For example: `AES,128` or `3DES`.
-syncWithWU DestinationDir	Sync with Windows Update.
-t timeout	URL fetch timeout in milliseconds.
-Unicode	Write redirected output in Unicode.

-UnicodeText	Write output file in Unicode.
-urlfetch	Retrieve and verify AIA Certs and CDP CRLs.
-user	Use the HKEY_CURRENT_USER keys or certificate store.
-username username	Use named account for SSL credentials. For selection UI, use `-username`.
-ut	Display user templates.
-v	Provide more detailed (verbose) information.
-v1	Use V1 interfaces.

Hash algorithms: MD2 MD4 MD5 SHA1 SHA256 SHA384 SHA512.

Related links

For more examples of how to use this command, see the following articles:

- Active Directory Certificate Services (AD CS)
- Certutil tasks for managing certificates
- Configure trusted roots and disallowed certificates in Windows

Feedback

Was this page helpful? Yes No