Functional Documentation for EIDAuthenticate

Version 1.7

Prepared by: "Vincent Le Toux"

Date: 28/04/2016

Table of Contents

Revision History

This section records the change history of this document.

Name	Date	Reason For Changes	Version
Vincent Le Toux	01/12/2012	Creation	1.0
Frédéric Bourgeois	23/02/2013	Update	1.1
Nathan J. Lichtenstein	19/04/2013	Review	1.2
Vincent Le Toux	02/07/2013	Description of the Development lifecycle	1.3
Vincent Le Toux	08/08/2013	Update to EIDAuthenticate 1.0	1.4
Vincent Le Toux	25/10/2013	Update to EIDAuthenticate 1.1	1.5
Vincent Le Toux	25/01/2014	Update to EIDAuthenticate 1.2	1.6
Vincent Le Toux	28/04/2016	Add the offline CRL check feature	1.7

Description

EIDAuthenticate is a software tool used to perform smart card logon on a standalone computer not joined to an active directory domain. The software was designed to support most smart cards and to be compatible with existing smart card components.

It offers the same level as the smart card logon mechanism offered by Microsoft :

- enforces the use of smart cards
- improves authentication for logon purposes safety
- locks the screen if the smart card is removed
- Forces logoff on smart card removal
- Minimum constraints were considered about certificate of smart card. (smart card logon EKU, signature only smart card, ...)

System Specifications

Operating systems supported are:

- Windows XP
- Windows Server 2003
- Windows Vista
- Windows 7
- Windows 8
- Windows Server 2008
- Windows Server 2012

The software can't be installed on a Windows XP or a Windows 2003 joined to an Active Directory Domain.

Hardware

The software requires that smart cards to use have a working CSP (Cryptographic service provider). The CSP can be released directly by the manufacturer or rely on the Microsoft "Base Smart Card provider" (aka using minidriver). On the second case, the PIN remaining attempt is only available to smart cards having a minidriver.

Smart cards having only a PKCS11 interface are not supported.

Requirements for the smart card reader are set by the manufacturer of the smart card, typically a CCID smart card reader. Smart card reader having a PINPAD have an undetermined behavior and have to be studied on a case by case basis.

External System Dependencies

This software requires certain external systems to complete some or all of its tasks. This section lists those dependencies explicitly:

External System	Dependencies on that System
Windows Installer 3.0	None
Smart card minidriver or CSP	Depends on the smart card manufacturer
Smart card reader driver	Depends on the smart card manufacturer

Components

EIDAuthentication includes the following components:

- EIDAuthenticationPackage.dll which is a security package. It is designed to be an extension of the Windows Security Kernel (LSA)
- EIDPasswordChangeNotification.dll which is a password filter library loaded into the LSA process to handle password change
- EIDKernelPackage.sys which is a kernel mode security package.
- EIDCredentialProvider.dll / EIDGina.dll which are libraries designed to handle communication with the end-user. These libraries are loaded into the session manager and take care of opening interactive sessions.
- EIDConfigurationWizard.exe / EIDConfigurationWizardElevated.exe which are programs design to configure the authentication behavior, e.g. configure a smart card
- EIDLogManager.exe which is a program design to capture and save debug traces

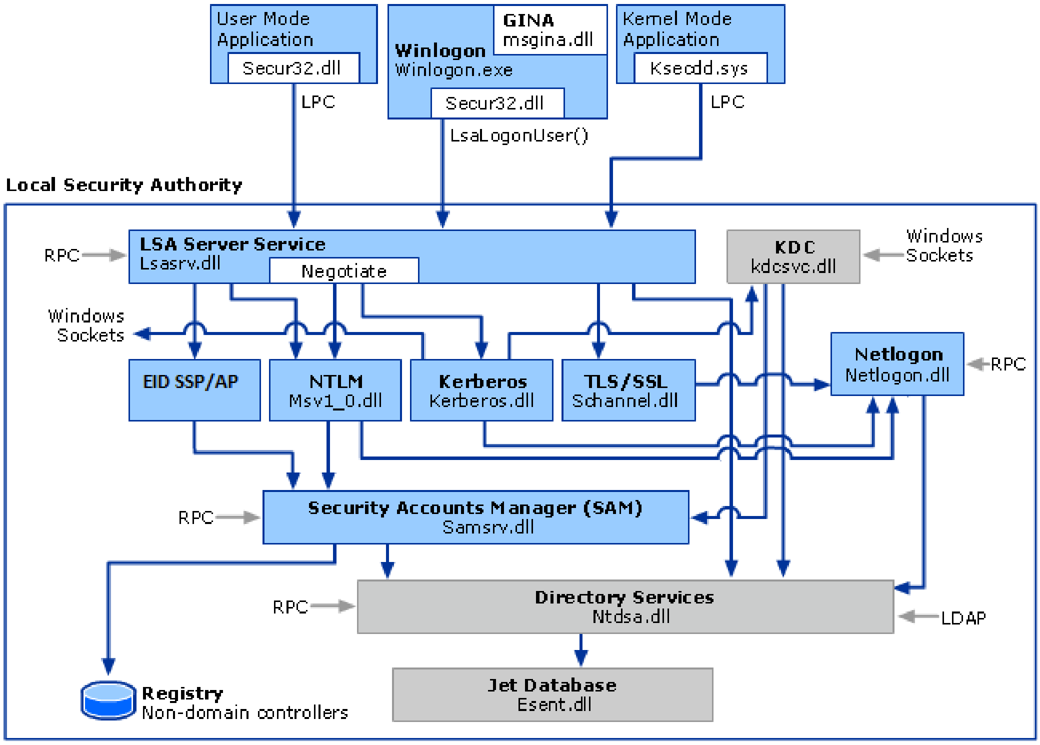
Authentication package

This component allows the creation of interactive and no interactive sessions. The creation of the interactive sessions are triggered by the Credential Provider or by the GINA package. Non interactive sessions, alias SSPI authentication, are done on a client / server scenario, like remote access.

The other internal authentication mechanism is not removed (Which means network share can be accessed with the password of a user).

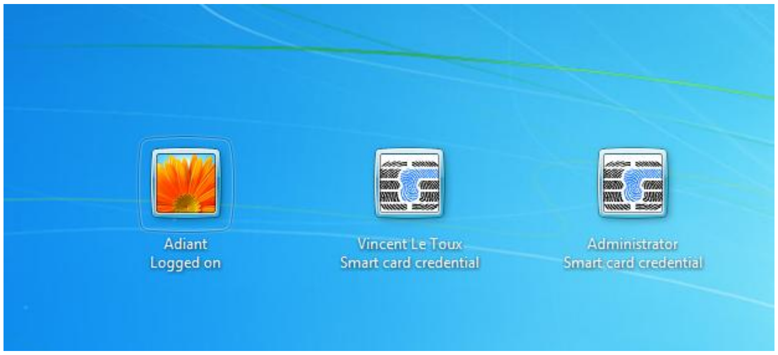
This component is responsible for applying security policy like the "force smart card logon" described later.

The following schema describes the position of the component "EID SSP/AP" in relation with the other security packages.

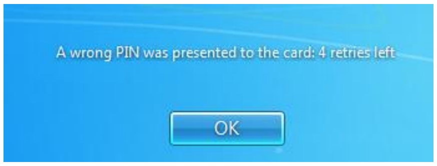


Credential Provider

This component adds to the session manager an interface to logon using the smart card. It detects the insertion of a smart card, asks for the PIN code, displays errors, and handles communication with the authentication package.



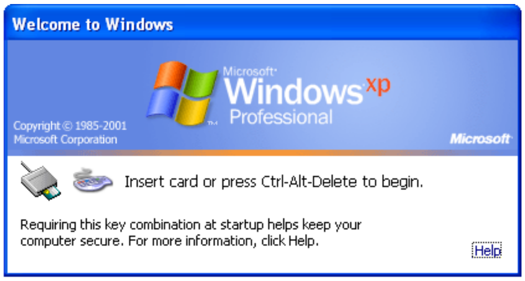
If available, it shows the remaining PIN attempts.



A wrong PIN was presented to the card: 4 retries left

GINA Dll

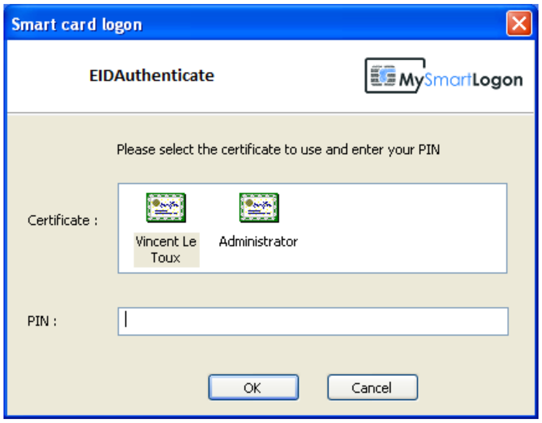
This component replaces the dialogs used to login on Windows XP and Windows 2003. The original component name found in the literature is "Microsoft GINA". GINA stand for "*Graphical Identification and Authentication dynamic-link library*"?



Insert card or press Ctrl-Alt-Delete to begin.

Requiring this key combination at startup helps keep your computer secure. For more information, click Help.

Figure 1 EIDAuthenticate GINA. Look for the addition of the smart card icon



Please select the certificate to use and enter your PIN

Certificate : Vincent Le Toux, Administrator

PIN :

Figure 2 EIDAuthenticate logon screen

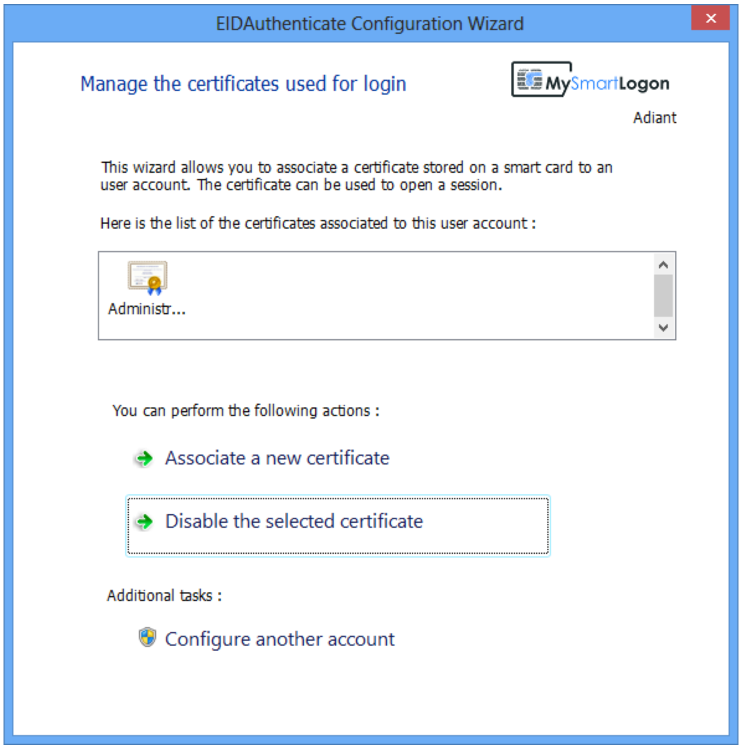
This GINA implements:

- PIN change
- Force logoff on smart card removal

Configuration Wizard

This program is designed to associate a smart card's certificate to a user account so the smart card can be used to login to this account.

This program runs with the current user on the machine to be configured ; an administrator account is not required.



EIDAuthenticate Configuration Wizard

Manage the certificates used for login

MySmartLogon

Adiant

This wizard allows you to associate a certificate stored on a smart card to an user account. The certificate can be used to open a session.

Here is the list of the certificates associated to this user account :

Administr...

You can perform the following actions :

Associate a new certificate

Disable the selected certificate

Additional tasks :

Configure another account

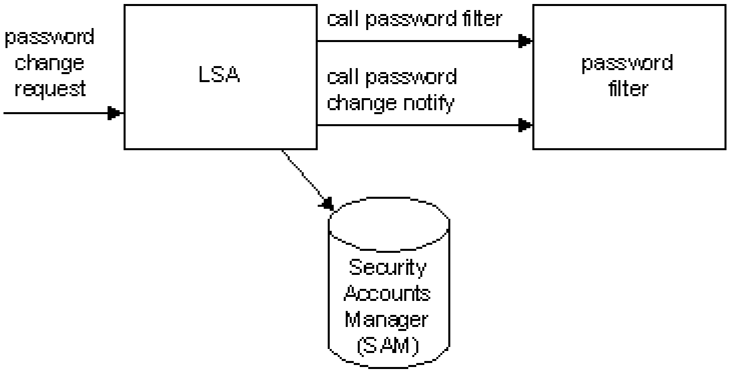
This program can import an existing certificate into the smart card, allows the certificate to be trusted, or is able to create a certificate on a smart card.

Advanced and **not supported** command line switches are :

- /ADVANCED : enable the advanced features like the user only "force policy".
- /LANG : for the use of the language defined in the control panel in the "format" tab.

Password filter

This component is responsible for handling password change notification and allows the internal credential to be updated.



The password filter after having received a notification forward it to the security package.

Development lifecycle

EIDAuthenticate has been rewrote starting with the version 1.0 to comply with most secure development methods.

The following technologies (described in the document "Windows ISV Software Security Defenses") have been activated :

- /GS Stack Buffer Overrun Detection
- /SafeSEH Exception Handling Protection
- Data Execution Prevention (DEP) / No eXecute (NX)
- Address Space Layout Randomization (ASLR)
- Heap Metadata Protection

The development lifecycle includes the following actions :

- Check carefully buffers sent from untrusted environments
- Add SAL notations to function prototypes and internal structures (SAL Reference)
- Run code analysis (PREFAST tool) and solves all warnings

The binaries are tested for compliance using :

- BinScope Binary Analyzer
- ApplicationVerifier

Note : the behavior regarding the "Additional LSA Protection" introduced in Windows 8.1 and Windows 2012 R2 (reference) is undetermined because the RunAsPPL setting requires that all components (including the smart card drivers) are compliant (reference).

Logon security flow

The logon process is split in two distinct processes. The first is the session manager, logonui.exe, on Vista, and Winlogon.exe on XP, the second is the security kernel also named LSA.

The session manager gathers the credentials, display error messages and the security kernel create the new session. It is responsible for checking the credentials.

Session Manager

The session manager is extended by EIDAuthenticate using a Credential Provider on Vista and using a Gina on XP. Many Credential Providers can be added while only one custom Gina can be added. Due to security segregation, the certificate is read twice, one in the session manager and one in the security kernel.

Security kernel

The Security Package is responsible for handling smart card authentication operations (certificate and PIN validation) and creating the session.

Logon process on Vista & later

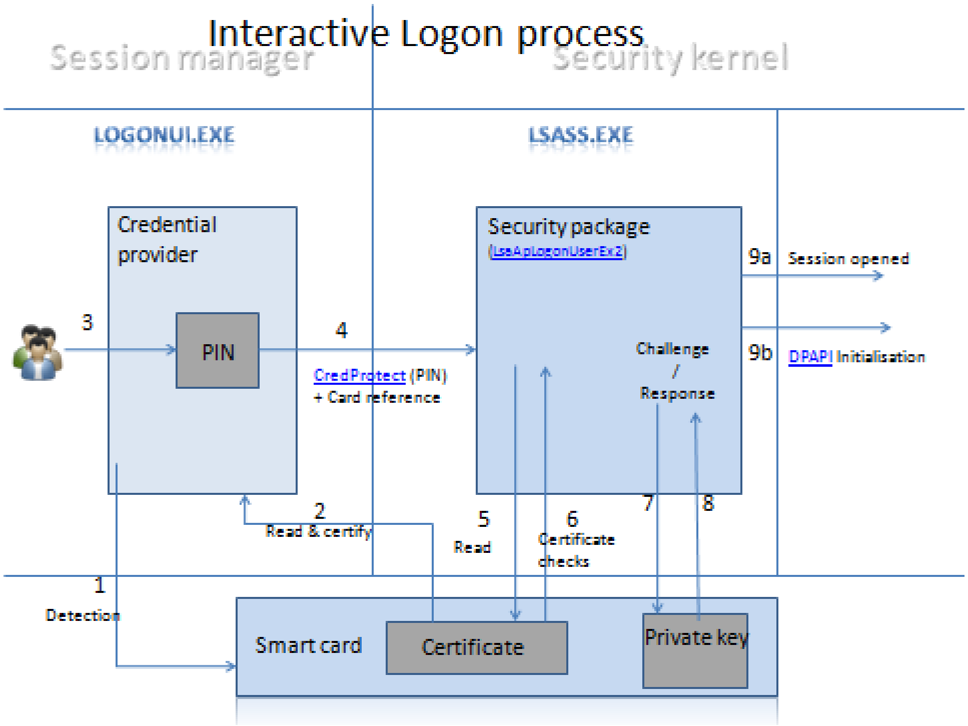


Figure 3: Logon process on Vista & later

Logon process on Windows XP & 2003

Security processes on Windows XP are slightly different because Microsoft doesn’t support the creation of a new GINA dll. The only supported development method is the "hooking development method".

EIDAuthenticate inserts a hook between Gina and Winlogon to switch to new screens flow when a smart card is inserted. It handles authentication operations then switches the control to the original GINA.

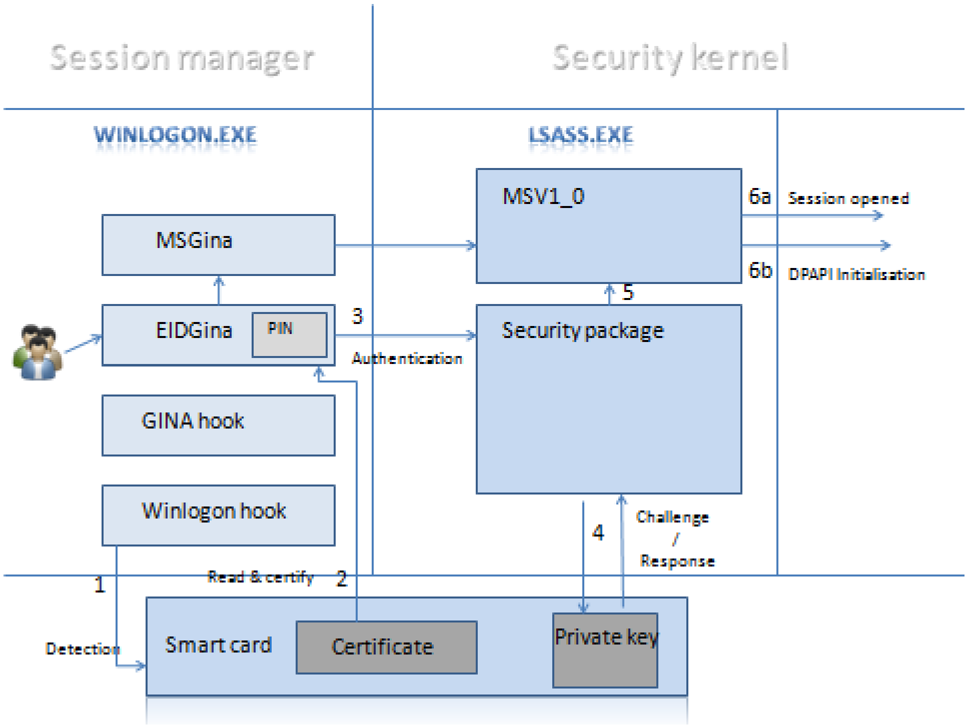


Figure 4 Logon process on Windows XP & 2003

The smart card operations are done in the process Winlogon.exe to benefit from the smart card redirection when terminal server is used.

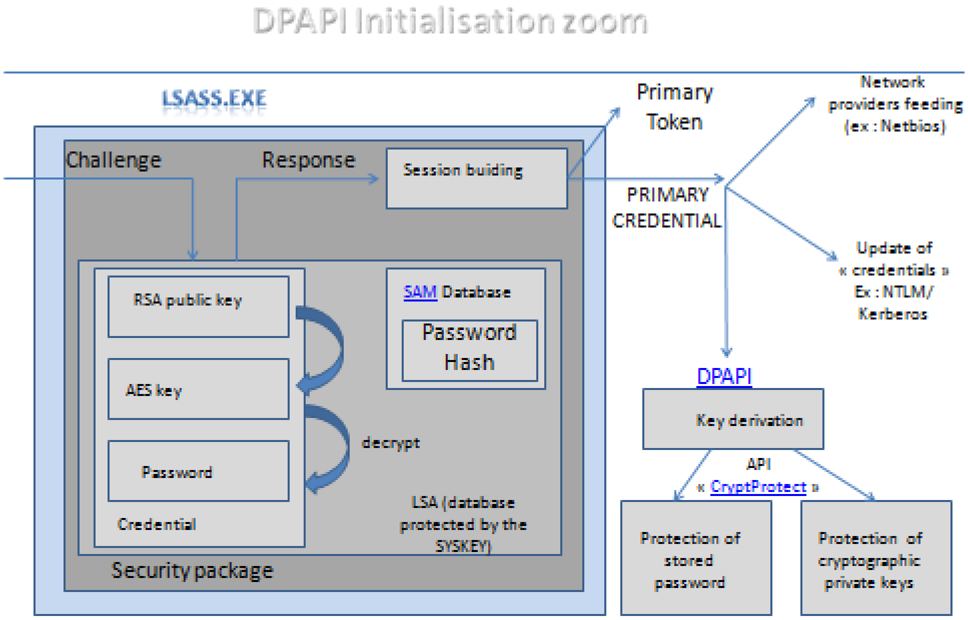
DPAPI

Windows Data Protection also named DPAPI is a mechanism implemented in Windows to protect sensitive data. It is used to protect certificate or the file encryption system EFS. An overview of this API can be found in this document[1].

This API is initialized by a secret based on the user password. The computation algorithm of this secret depends on the version of the operating system but it is typically a SHA1 PBKDF2 derivation with thousands iterations.

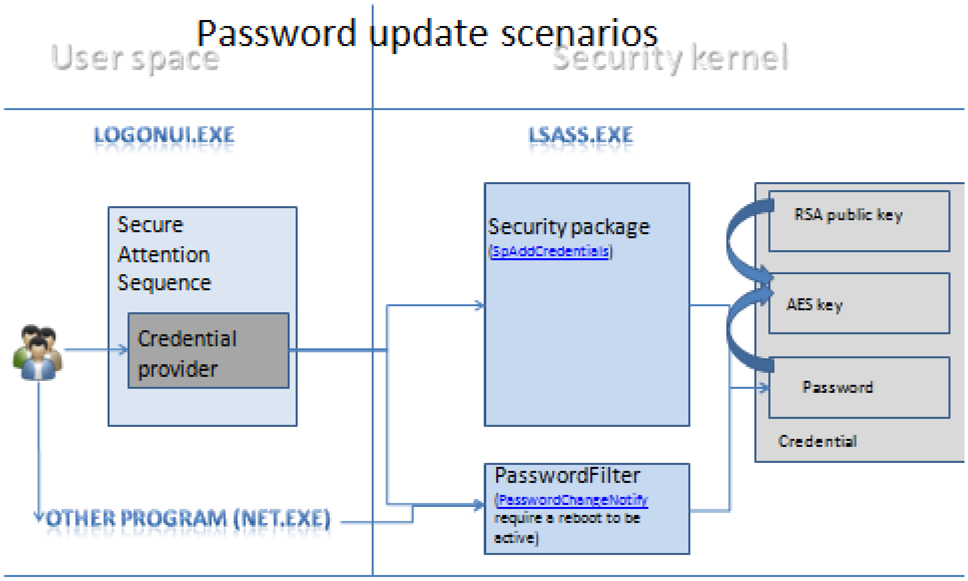
EIDAuthenticate provides the data to perform this initialization.

[1] http://msdn.microsoft.com/en-us/library/ms995355.aspx


DPAPI Initialisation zoom
LSASS.EXE
Challenge
Response
Session building
Primary Token
Network providers feeding (ex : Netbios)
PRIMARY CREDENTIAL
Update of « credentials » Ex : NTLM/ Kerberos
RSA public key
AES key
Password
Credential
SAM Database
Password Hash
decrypt
LSA (database protected by the SYSKEY)
Security package
DPAPI
Key derivation
API « CryptProtect »
Protection of stored password
Protection of cryptographic private keys

If the « credential » created on output doesn't match the NTLM password, the session is opened, but a password change event is triggered. Cryptographic material cannot be accessed anymore.

Password update flow


Password update scenarios
User space
Security kernel
LOGONUI.EXE
LSASS.EXE
Secure Attention Sequence
Credential provider
Security package (SpAddCredentials)
RSA public key
AES key
Password
Credential
OTHER PROGRAM (NET.EXE)
PasswordFilter (PasswordChangeNotify require a reboot to be active)

The password is encrypted by a new AES key, itself encrypted by the public key of the smart card saved when EIDAuthenticate has been configured.
Note : to be in sync when the password is changed offline, the workstation must have been rebooted once because the passwordfilter cannot be dynamically loaded at the setup (at the opposite of a security package)

Because EIDAuthenticate must return the secret for the DPAPI initialization, the secret must be updated when the password is changed. When this happens, two entry points are triggered:

- The first is inside the security package for the login already in use.

- The second is in all password filter dll.

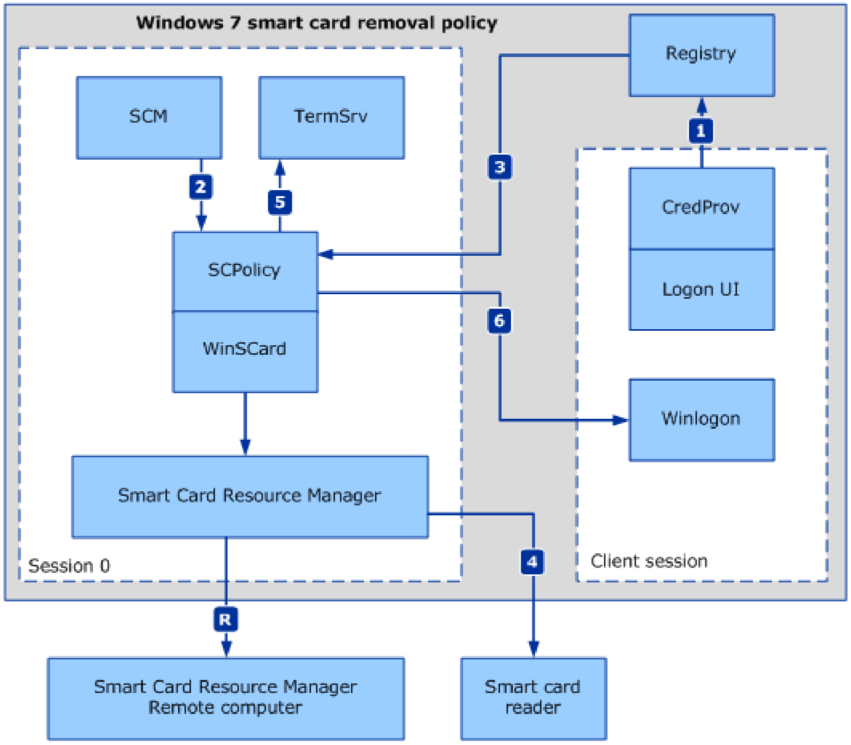
The password filter requires a reboot to be active. Passwords changed administratively, for example using the command "net use", before the first reboot are not handled.

Smart card removal

This component use the service "Smart Card Removal Policy Service" designed by Microsoft.

Here is a citation of the documentation found on technet.

The smart card removal policy is applicable when a user has logged on with a smart card and subsequently removes that smart card from the reader. The action that is performed when the smart card is removed is controlled by using Group Policy. For information about smart card Group Policy settings, see Smart Card Group Policy and Registry Settings.


Windows 7 smart card removal policy
Registry
SCM
TermSrv
1
2
5
3
CredProv
Logon UI
SCPolicy
WinSCard
6
Winlogon
Smart Card Resource Manager
Session 0
4
Client session
R
Smart Card Resource Manager Remote computer
Smart card reader

1. *In Windows Server 2008 R2, Windows Server 2008, Windows 7, and Windows Vista, Winlogon is no longer directly involved in monitoring for smart card removal events. The sequence of steps involved in removal policy begins with the smart card credential provider in the logon UI process. When a user successfully logs on with a smart card, the smart card credential provider captures the reader name. This information is then stored in the registry along with the session identifier where the logon was initiated.*
2. *The smart card resource manager notifies the smart card removal policy service that a logon has occurred.*
3. *ScPolicySvc retrieves the smart card information from the registry that the smart card credential provider stored. This call is redirected if the user is in a remote session. If the smart card is removed, ScPolicySvc is notified.*

4. *ScPolicySvc calls Remote Desktop Services to take the appropriate action if the request is to log the user off or to disconnect the user's session, which might result in data loss. If the setting is configured to lock the computer when the smart card is removed, then ScPolicySvc sends a message to Winlogon to lock the computer.*

The removal policy needs to be active that the next session is opened using the smart card. It can be a logoff/logon or a lock/unlock scenario.

If the service is disabled, the policy is not enforced.

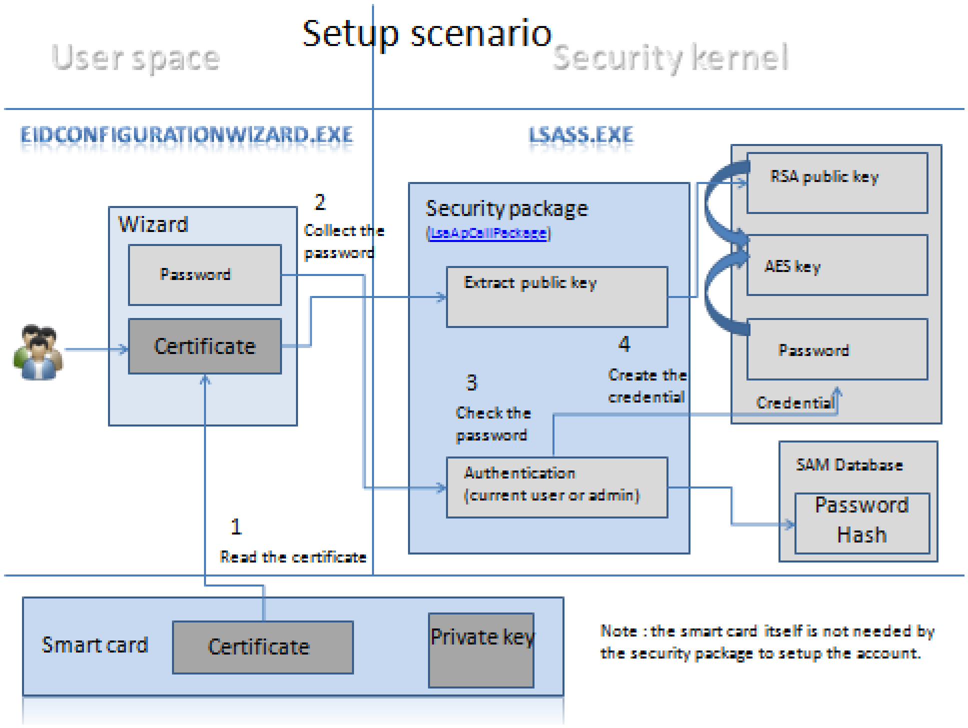
The mmc dialog and the dialog of EIDAuthenticate accessible from the Control Panel enforce the same registry keys. The difference is that EIDAuthenticate start or stop the remove policy service when the configuration is altered.

Security package interface

The configuration wizard interacts with the .security package using the following interface.

Action	**Context**	**Result**
Remove all existing credential	Used in the uninstaller	Remove all data stored for authentication
Remove existing credential	Remove existing data stored for the current user if it exists	Remove stored data
Configure an existing smart card	Extract a certificate from a smart card and create a credential	Create stored data
Create a new certificate	Create a root certificate and/or a user certificate and/or import a certificate on a smart card Create a credential based on this certificate	Create stored data

The following schema details how credentials are created from a certificate.



Data storage

Data is stored in the “Local Security Authority” (LSA) secret database. This database is designed for use with Windows Security Kernel. The database is protected by the SYSKEY and designed to be accessed by the LSASS.exe process.

The software stores the credential in two forms :

- encrypted form (default)

Password is encrypted using a random 256 bits AES key which is encrypted itself by the RSA key contained in the certificate. The container is composed by the encrypted password, the encrypted key and the certificate

- Plain text form

This form is designed for smart card which can’t perform decryption. The Password is stored in plain text with the certificate.

The plain-test form protection used by Signature only smart card can be disabled by injecting a dll in the lsass.exe process address space. This attacks needs an administrator account and to have the DEBUG privilege.

Certificate Validation Process against a CRL or an OCSP

EIDAuthenticate starting from 1.0.2.0 enforces by default certificate validation against a CRL or a OCSP. This behavior wasn't enforced by default before because this software has been designed for use cases where the user has physical access to the computer. It can remove the network plug and bypass this control : the CRL information's is valid for a certain period of time and CRLs are not designed to handle real time revocation. Also this control decreases the user experience because the logon takes more time and computer can be blocked if the CRL / OCSP server is down.

Here is how this control is made. The trust chain-building process made at the logon validates the certification path by checking each certificate in the certification path from the end certificate to the root CA's certificate. The certificates are retrieved from the Intermediate Certification Authorities store, the Trusted Root Certification Authorities store. If CryptoAPI discovers a problem with one of the certificates in the path, or if it cannot find a certificate, the certification path is discarded as a non-trusted certification path. As a consequence the logon fails.

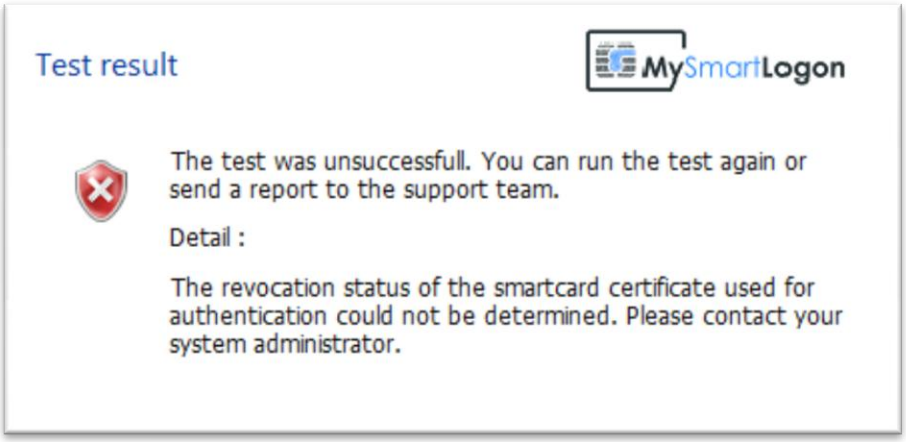
The chain building is disabled if the certificate is found on the computer "trusted certificate" store. This store can be filled by the configuration wizard with the user consent with the certificate being configured if the wizard cannot build the chain. As a consequence, and in this specific case, the CRL configuration stored in the CA certificate will not be found.

CRL / OCSP Policies

To modify the CRL / OCSP checking behavior, add the DWORD registry key *ForceCRLCheck* and set its value to 1 in *HKLM\SOFTWARE\Policies\Microsoft\Windows\SmartCardCredentialProvider*. The default is 0. Set this value to 0 to disable certificate revocation settings.

When this value is set to 1 and if the CRL server in unavailable, the login will still succeed. Set this value to 2 to make the login fail if the CRL server is unavailable and if the CRL data is not cached.

IMPORTANT : ForceCRLCheck is set to 2 and if the CRL / OCSP server can't be contacted and if the revocation information is not cached, the login process will fail :



Test result

The test was unsuccessfull. You can run the test again or send a report to the support team.

Detail :

The revocation status of the smartcard certificate used for authentication could not be determined. Please contact your system administrator.

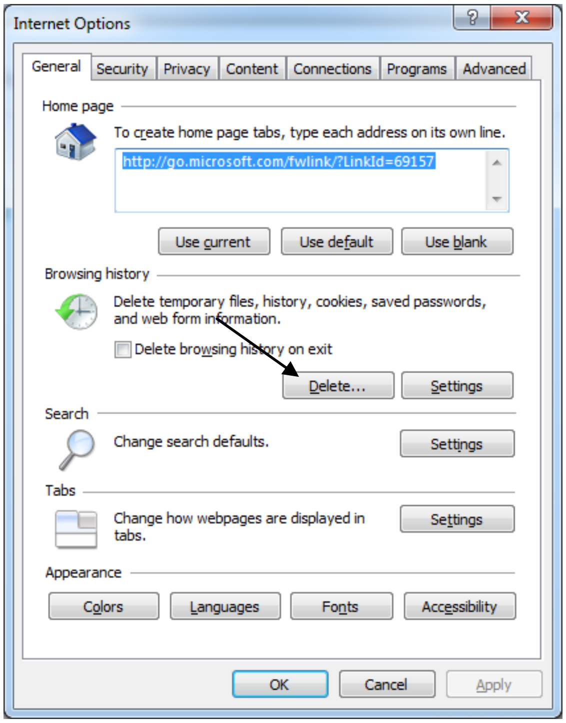
The key *UrlRetrievalTimeout* can be set to define the maximum time in ms before issuing a timeout. Default is 60s.

CRL Caching

We are describing here how to clear the caches that CryptoAPI maintains to test for certificate revocation.

First, unplug the network to disable the revocation processes active.

If the CRL is published via HTTP / HTTPS, you have to clear the browsing history of WinHttp via the Options panel of Internet Explorer. Select *Delete* on the *Browsing History* zone.



Internet Options

Home page

To create home page tabs, type each address on its own line.

http://go.microsoft.com/fwlink/?LinkId=69157

Browsing history

Delete temporary files, history, cookies, saved passwords, and web form information.

□ Delete browsing history on exit

Search

Change search defaults.

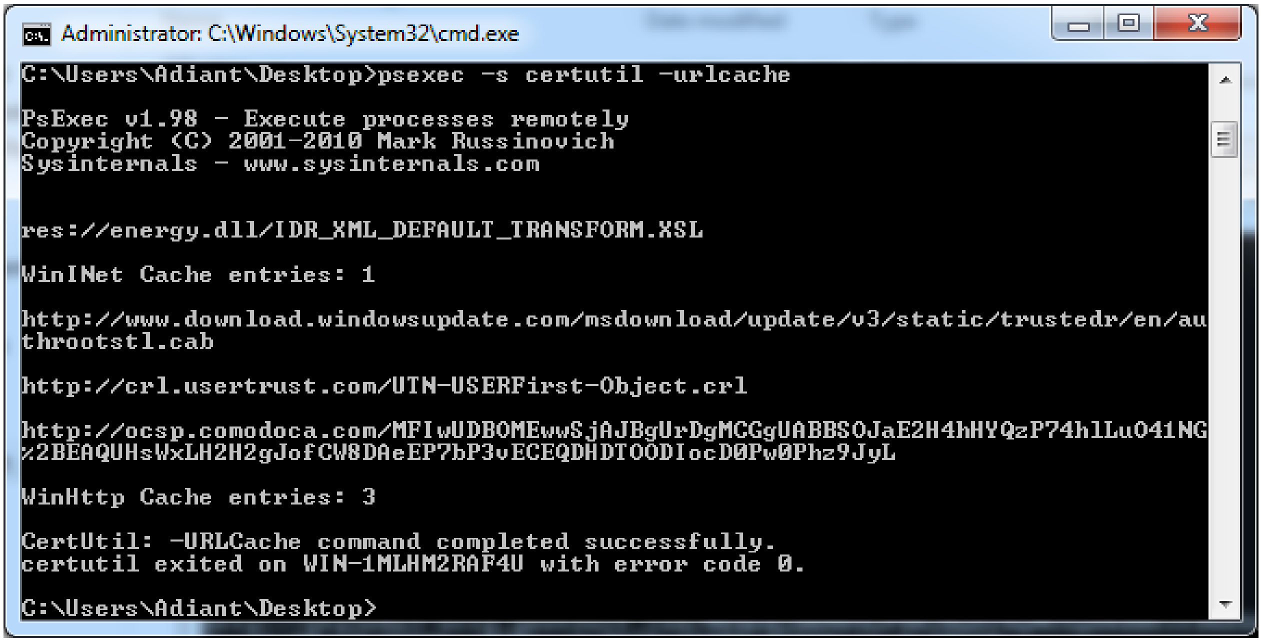
Tabs

Change how webpages are displayed in tabs.

Appearance

Then you have to clear the CryptoAPI disk cache.

Run the command "*psexec -s certutil -urlcache * delete*" using the utility psexec.exe provided by SysInternals to clear the cache information of the system account. You can check the cache status using "*psexec -s certutil -urlcache*". *Psexec* need an elevated prompt to be run.



Administrator: C:\Windows\System32\cmd.exe

```
C:\Users\Adiant\Desktop>psexec -s certutil -urlcache

PsExec v1.98 - Execute processes remotely
Copyright (C) 2001-2010 Mark Russinovich
Sysinternals - www.sysinternals.com


res://energy.dll/IDR_XML_DEFAULT_TRANSFORM.XSL

WinINet Cache entries: 1

http://www.download.windowsupdate.com/msdownload/update/v3/static/trustedr/en/au
throotstl.cab

http://crl.usertrust.com/UTN-USERFirst-Object.crl

http://ocsp.comodoca.com/MFIwUDBOMEwwSjAJBgUrDgMCGgUABBSOJaE2H4hHYQzP74hlLu041NG
%2BEAQUHsWxLH2H2gJofCW8DAeEP7bP3vECEQDHDTOODIocD0Pw0Phz9JyL

WinHttp Cache entries: 3

CertUtil: -URLCache command completed successfully.
certutil exited on WIN-1MLHM2RAF4U with error code 0.

C:\Users\Adiant\Desktop>
```

You can look at the CryptnetUrlCache folder of the SYSTEM account folder (this folder is located in "%WINDIR%\config\systemprofile\AppData\LocalLow\Microsoft\) to monitor this operation.

Then clear the cache of the logon process by running in an elevated prompt :

certutil -setreg chain\ChainCacheResyncFiletime @now

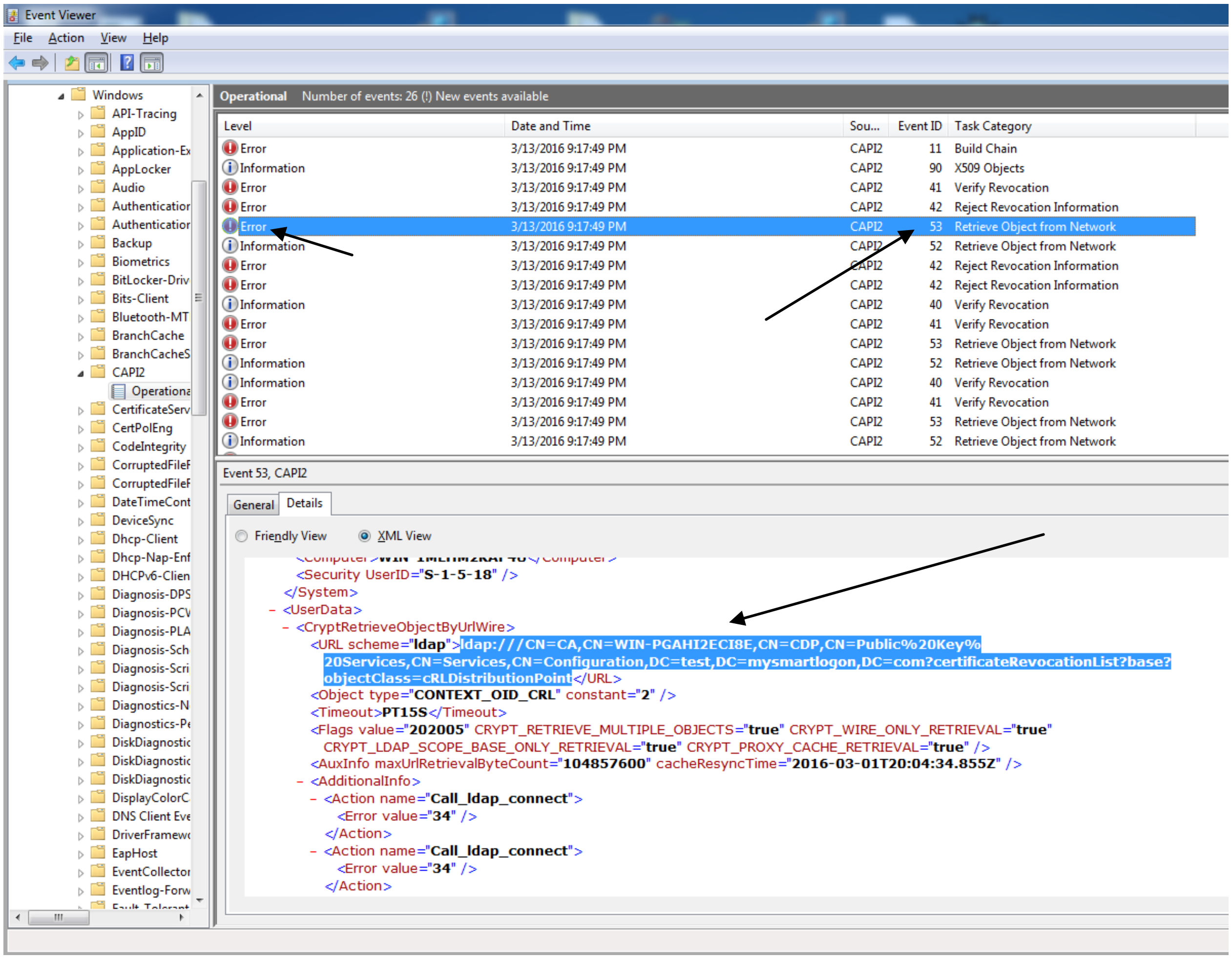
If the network is disconnected, if *ForceCRLCheck* is set to 2 and if you have a certificate with revocation information, you should get a revocation status error message when performing a logon.

More information about CRL OCSCP caching can be found in the article Troubleshooting PKI problems on Windows Vista

Offline mode

CRL files can be downloaded and copied manually to a folder. This folder can be added in the verification procress since EIDAuthenticate 1.2.4 by setting the value of the registry key AdditionalCRLPath in HKLM\SOFTWARE\Policies\Microsoft\Windows\SmartCardCredentialProvider. Its value is a string and is a directory like "c:\CrlPath".

Determining the correct set of CRL to put in the folder for a full offline mode can be difficult because some CRL file are needed for verifying certification authorities or crl signatures. The missing CRL can be found by applying the CryptoAPI event logging as described here: https://technet.microsoft.com/en-us/library/cc749296(v=ws.10).aspx. Select the event "Retrieve Object from Network" to get the URL of the object to add.



Operational Number of events: 26 (!) New events available

Level	Date and Time	Sou...	Event ID	Task Category
Error	3/13/2016 9:17:49 PM	CAPI2	11	Build Chain
Information	3/13/2016 9:17:49 PM	CAPI2	90	X509 Objects
Error	3/13/2016 9:17:49 PM	CAPI2	41	Verify Revocation
Error	3/13/2016 9:17:49 PM	CAPI2	42	Reject Revocation Information
Error	3/13/2016 9:17:49 PM	CAPI2	53	Retrieve Object from Network
Information	3/13/2016 9:17:49 PM	CAPI2	52	Retrieve Object from Network
Error	3/13/2016 9:17:49 PM	CAPI2	42	Reject Revocation Information
Error	3/13/2016 9:17:49 PM	CAPI2	42	Reject Revocation Information
Information	3/13/2016 9:17:49 PM	CAPI2	40	Verify Revocation
Error	3/13/2016 9:17:49 PM	CAPI2	41	Verify Revocation
Error	3/13/2016 9:17:49 PM	CAPI2	53	Retrieve Object from Network
Information	3/13/2016 9:17:49 PM	CAPI2	52	Retrieve Object from Network
Information	3/13/2016 9:17:49 PM	CAPI2	40	Verify Revocation
Error	3/13/2016 9:17:49 PM	CAPI2	41	Verify Revocation
Error	3/13/2016 9:17:49 PM	CAPI2	53	Retrieve Object from Network
Information	3/13/2016 9:17:49 PM	CAPI2	52	Retrieve Object from Network

Event 53, CAPI2

General | Details

○ Friendly View ● XML View

```
    <Security UserID="S-1-5-18" />
  </System>
- <UserData>
  - <CryptRetrieveObjectByUrlWire>
      <URL scheme="ldap">ldap:///CN=CA,CN=WIN-PGAHI2ECI8E,CN=CDP,CN=Public%20Key%
        20Services,CN=Services,CN=Configuration,DC=test,DC=mysmartlogon,DC=com?certificateRevocationList?base?
        objectClass=cRLDistributionPoint</URL>
      <Object type="CONTEXT_OID_CRL" constant="2" />
      <Timeout>PT15S</Timeout>
      <Flags value="202005" CRYPT_RETRIEVE_MULTIPLE_OBJECTS="true" CRYPT_WIRE_ONLY_RETRIEVAL="true"
        CRYPT_LDAP_SCOPE_BASE_ONLY_RETRIEVAL="true" CRYPT_PROXY_CACHE_RETRIEVAL="true" />
      <AuxInfo maxUrlRetrievalByteCount="104857600" cacheResyncTime="2016-03-01T20:04:34.855Z" />
    - <AdditionalInfo>
      - <Action name="Call_ldap_connect">
          <Error value="34" />
        </Action>
      - <Action name="Call_ldap_connect">
          <Error value="34" />
        </Action>
```

Security policies

Login policy

EIDAuthenticate does not require to change a password when it has expired.

It doesn’t allow the logon if the account (not the password) is expired.

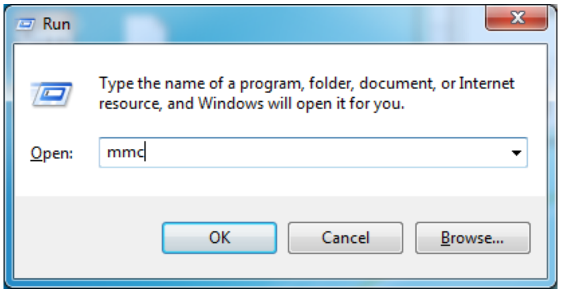
The auto locking of accounts is ignored because the risk of having multiple login attempts is forwarded to the smart card.

Trust policy

EIDAuthenticate is using the computer store (and not the user store) to check the trust of the chain of certificates.

Here is a short procedure to review it :

type : Windows + L then mmc. Press OK.



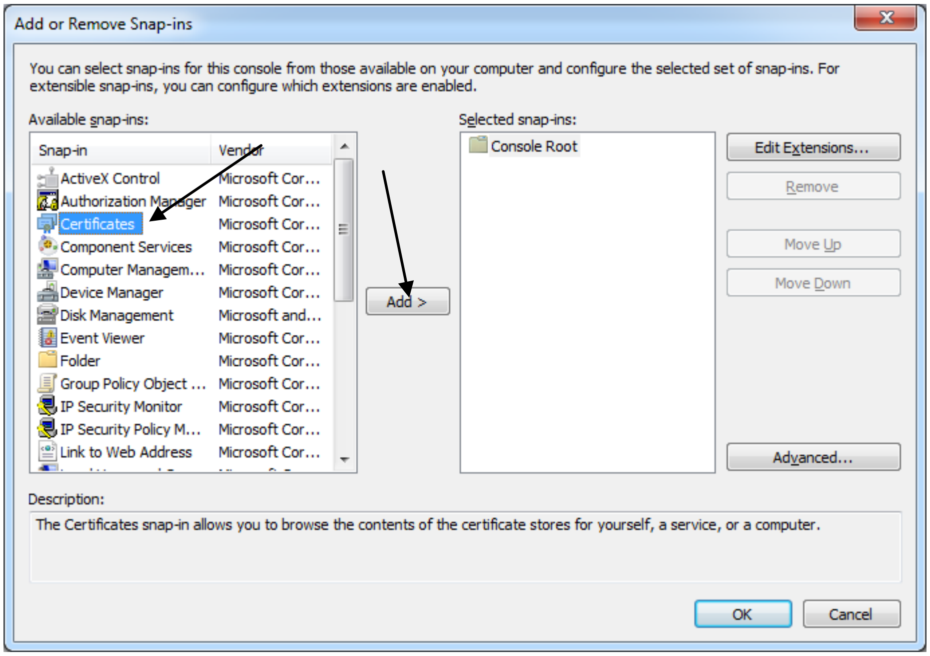
Run

Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: mmc

Menu File -> Add or remove Snap-ins

Select "Certificates" then "Add>".



Add or Remove Snap-ins

You can select snap-ins for this console from those available on your computer and configure the selected set of snap-ins. For extensible snap-ins, you can configure which extensions are enabled.

Available snap-ins:

Snap-in	Vendor
ActiveX Control	Microsoft Cor...
Authorization Manager	Microsoft Cor...
Certificates	Microsoft Cor...
Component Services	Microsoft Cor...
Computer Managem...	Microsoft Cor...
Device Manager	Microsoft Cor...
Disk Management	Microsoft and...
Event Viewer	Microsoft Cor...
Folder	Microsoft Cor...
Group Policy Object ...	Microsoft Cor...
IP Security Monitor	Microsoft Cor...
IP Security Policy M...	Microsoft Cor...
Link to Web Address	Microsoft Cor...

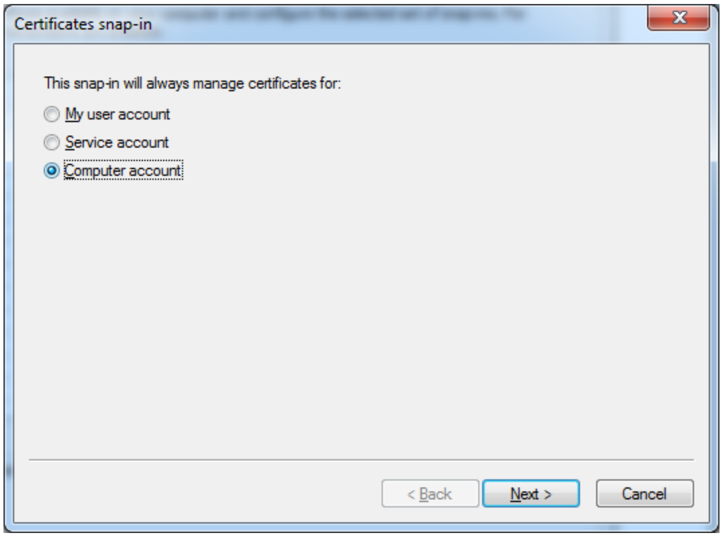
Selected snap-ins:

Console Root

Description:

The Certificates snap-in allows you to browse the contents of the certificate stores for yourself, a service, or a computer.

Select "Computer account" then Next> and complete the wizard.

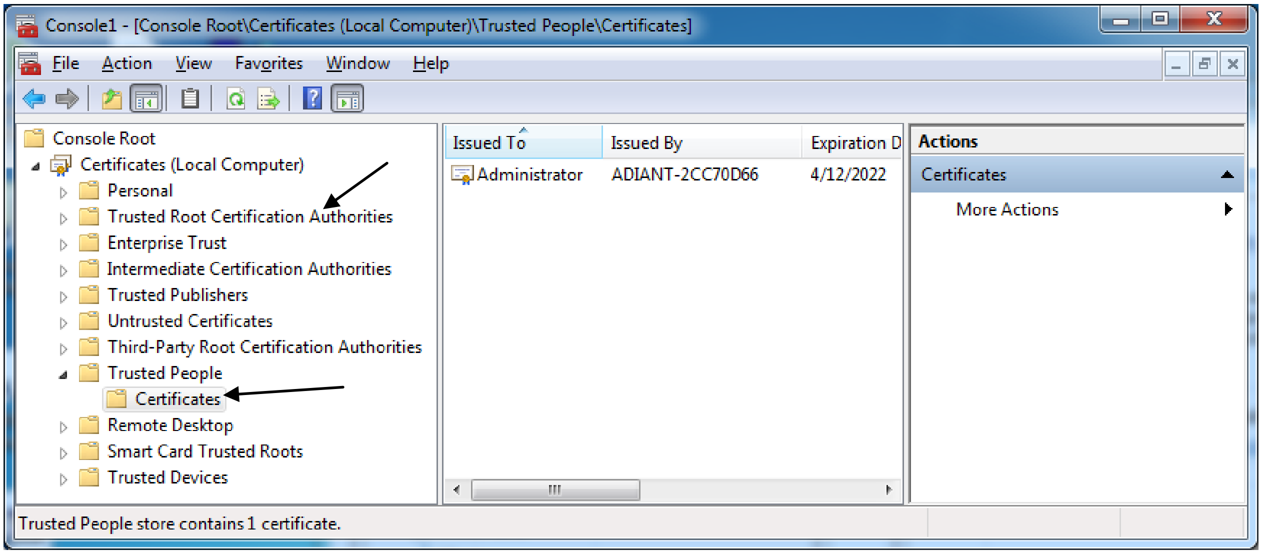


Certificates snap-in

This snap-in will always manage certificates for:

○ My user account
○ Service account
● Computer account

The wizard put the certificates with no chain in the store "Trusted People". For certificates with a user trusted chain, the certificates of the chain are extracted, then put into the "Trusted Root Certification Authorities" and into the "Intermediate Certification Authorities" if there are more than 2 certificates in the chain.



Console1 - [Console Root\Certificates (Local Computer)\Trusted People\Certificates]

Issued To	Issued By	Expiration D
Administrator	ADIANT-2CC70D66	4/12/2022

Trusted People store contains 1 certificate.

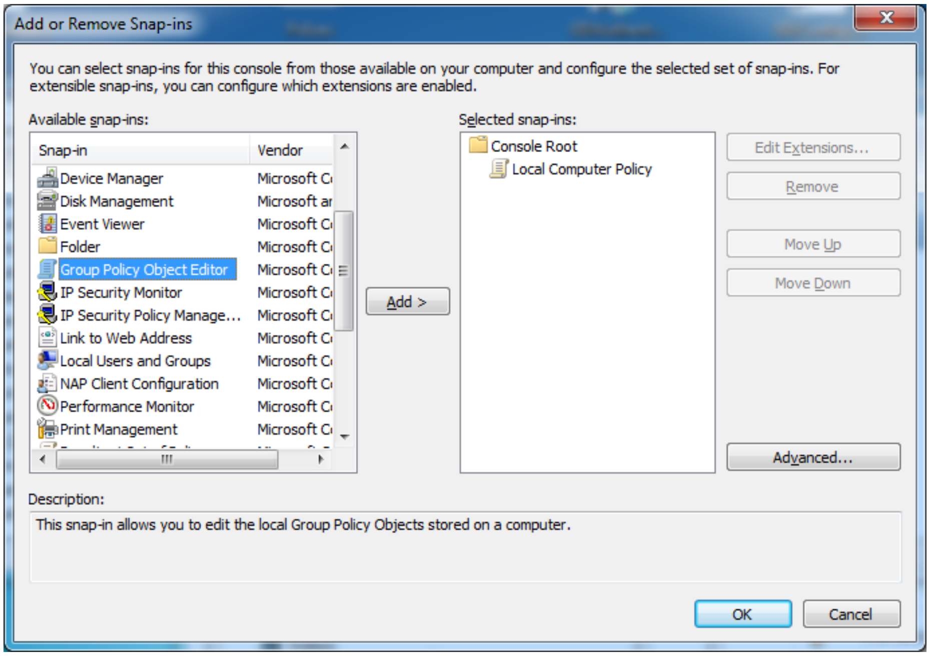
Remove policy overview

The remove policy is handled differently depending on the operating system. On Windows XP, the EIDGina hooks the smart card removal event and lock the workstation. This mechanisms is not multi-user aware.

On Windows Vista and later, a service named “Smart Card Removal Policy service” **must** be started for this policy setting to work. It detects newly created sessions and monitors smart card events. This service is designed by Microsoft and is used as-is.

The policy is applied for the next smart card session opened and not for the current one.

This policy can be set using the built-in program (the configuration Wizard) or can be set manually. The Group Policy settings are located in **Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options** in the mmc.exe "Group Policy Object Editor" snap-in.



Add or Remove Snap-ins

You can select snap-ins for this console from those available on your computer and configure the selected set of snap-ins. For extensible snap-ins, you can configure which extensions are enabled.

Available snap-ins:

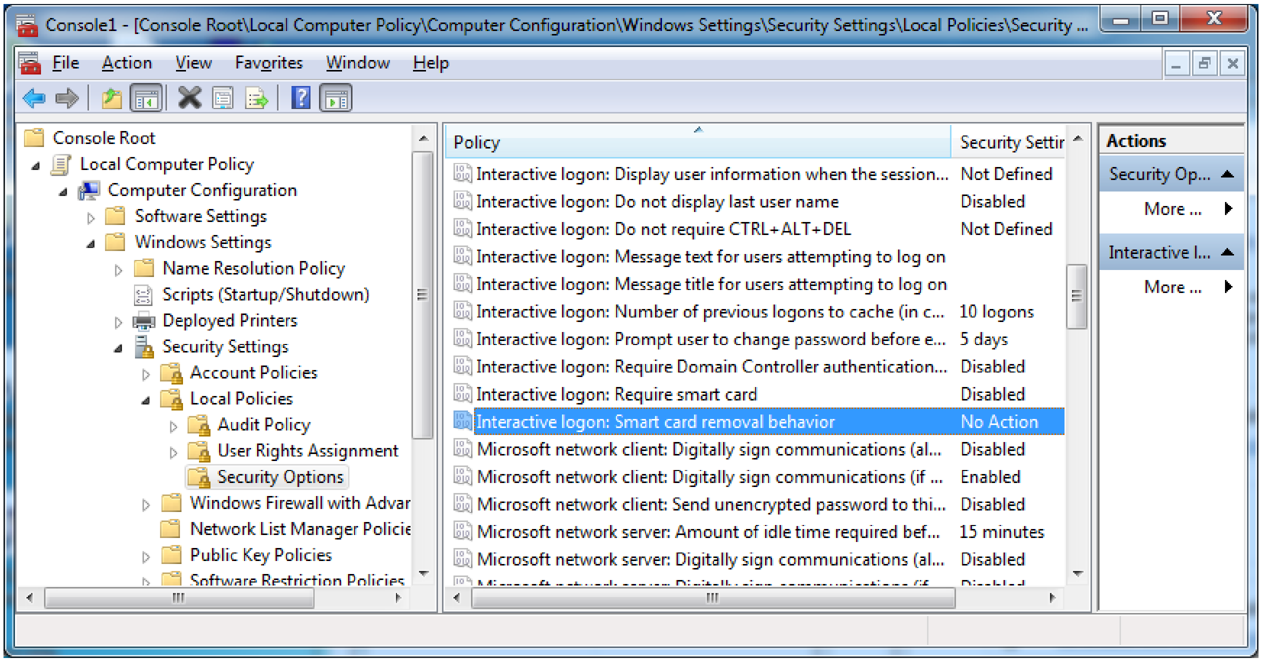
Snap-in	Vendor
Device Manager	Microsoft Co
Disk Management	Microsoft ar
Event Viewer	Microsoft Co
Folder	Microsoft Co
Group Policy Object Editor	Microsoft Co
IP Security Monitor	Microsoft Co
IP Security Policy Manage...	Microsoft Co
Link to Web Address	Microsoft Co
Local Users and Groups	Microsoft Co
NAP Client Configuration	Microsoft Co
Performance Monitor	Microsoft Co
Print Management	Microsoft Co

Selected snap-ins:

Console Root
Local Computer Policy

Description:

This snap-in allows you to edit the local Group Policy Objects stored on a computer.



Console1 - [Console Root\Local Computer Policy\Computer Configuration\Windows Settings\Security Settings\Local Policies\Security ...

Console Root
Local Computer Policy
Computer Configuration
Software Settings
Windows Settings
Name Resolution Policy
Scripts (Startup/Shutdown)
Deployed Printers
Security Settings
Account Policies
Local Policies
Audit Policy
User Rights Assignment
Security Options
Windows Firewall with Advar
Network List Manager Policie
Public Key Policies
Software Restriction Policies

Policy	Security Settir
Interactive logon: Display user information when the session...	Not Defined
Interactive logon: Do not display last user name	Disabled
Interactive logon: Do not require CTRL+ALT+DEL	Not Defined
Interactive logon: Message text for users attempting to log on	
Interactive logon: Message title for users attempting to log on	
Interactive logon: Number of previous logons to cache (in c...	10 logons
Interactive logon: Prompt user to change password before e...	5 days
Interactive logon: Require Domain Controller authentication...	Disabled
Interactive logon: Require smart card	Disabled
Interactive logon: Smart card removal behavior	No Action
Microsoft network client: Digitally sign communications (al...	Disabled
Microsoft network client: Digitally sign communications (if ...	Enabled
Microsoft network client: Send unencrypted password to thi...	Disabled
Microsoft network server: Amount of idle time required bef...	15 minutes
Microsoft network server: Digitally sign communications (al...	Disabled

Group Policy setting	Registry key	Default	Description
Interactive logon: Smart card removal behavior	scremoveoption	This policy setting is not defined, which means that the system treats it as **No Action**.	This setting determines what happens when the smart card for a logged-on user is removed from the smart card reader. The options are: **No Action** **Lock Workstation** The workstation is locked when the smart card is removed, allowing users to leave the area, take their smart card with them, and still maintain a protected session. **Force Logoff** The user is automatically logged off when the smart card is removed. **Disconnect if a Remote Desktop Services session** Removal of the smart card disconnects the session without logging the user off. This allows the user to reinsert the smart card and resume the session later, or at another smart card reader equipped computer, without having to log on again. If the session is local, this policy setting functions identically to the **Lock Workstation** policy setting. Remarque Remote Desktop Services was called Terminal Services in previous versions of Windows Server. For Windows 7 and Windows Vista, the Smart Card Removal Policy service must be started for this policy setting to work.

Force smart card policy

The enforcement of smart card is managed by EIDAuthenticate by setting the flag `PRIMARY_CRED_INTERACTIVE_SMARTCARD_LOGON` when the session is created. The LSA understand that this session is not password based and apply the security policies restricting the usage of the password if they have been set.

The documented policy is a machine wide security policy.

This policy can be set using the built-in program (the configuration Wizard) or can be set manually. The Group Policy settings are located in **Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options**.

Group Policy setting	Registry key	Default	Description
Interactive logon: Require smart card	scforceoption	Disabled	This security policy setting requires users to log on to a computer by using a smart card. **Enabled** Users can only log on to the computer by using a smart card. **Disabled** Users can log on to the computer by using any method.

An undocumented policy exist to apply this restriction to a single account. It consists of setting the flag `USER_SMARTCARD_REQUIRED` to the Account Control field of an account record.

This flag can be set using the built-in configuration program but this option is set as experimental given the fact that side effects have not been collected.

Certificate policies

The following smart card Group Policy settings are located in **Computer Configuration\Administrative Templates\Windows Components\Smart Card**.

The registry keys are in the following locations:

HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\SmartCardCredentialProvider

Group Policy setting	Registry key	Default	Description

Allow certificates with no extended key usage certificate attribute	AllowCertificatesWithNoEKU	Enabled	This policy setting allows certificates without an enhanced key usage (EKU) set to be used for logon. In versions of Windows prior to Windows Vista, smart card certificates that are used for logon require an EKU extension with a smart card logon object identifier. This policy setting can be used to modify that restriction. Enabled Certificates with the following attributes can also be used to log on with a smart card: Certificates with no EKU Certificates with an All Purpose EKU Certificates with a Client Authentication EKU Disabled or Not Configured Only certificates that contain the smart card logon object identifier can be used to log on with a smart card.
Allow signature keys valid for Logon	AllowSignatureOnlyKeys	Enabled	This policy setting lets you allow signature key-based certificates to be enumerated and available for logon. Enabled Any certificates available on the smart card with a signature-only key are listed on the logon screen. Disabled or Not Configured Any certificates available on the smart card with a signature-only key are not listed on the logon screen.

Allow time invalid certificates	AllowTimeInvalidCertificates	Enabled	This policy setting permits those certificates that are expired or not yet valid to be displayed for logon. Under previous versions of Windows, certificates were required to contain a valid time and to not be expired. To be used, the certificate must be accepted by the domain controller. This policy setting only controls which certificates are displayed on the client computer. Enabled Certificates are listed on the logon screen whether they have an invalid time or their time validity has expired. Disabled or Not Configured Certificates that are expired or not yet valid are not listed on the logon screen.
Force CRL or OCSP check	*ForceCRLCheck*	1	This policy setting permits to define how the revocation check is done when the login is performed. If the value is set to 0, no CRL / OCSP check is done. Login works with revoked certificates. If the value is set to 1, a CRL / OCSP check is done. After a check has been done, the revocation data is cached. If no cached data or fresh revocation data is available, the login succeed (default). If the value is set to 2, a CRL / OCSP check is done. After a check has been done, the revocation data is cached. If no cached data or fresh revocation data is available, the login fails. Caching mechanisms are handled by CAPI, the crypto API library of Windows.
CRL or OCSP check timeout	*UrlRetrievalTimeout*	60000	When revocation checks are handled, this settings define the maximum time allowed to retrieve revocation data. When the timeout is reached, the revocation check stop and the login can fail based on the ForceCRLCheck Policy. However, revocation data transfer continues after this delay. Once the revocation data is retrieved, it is cached by CAPI.