MySmartLogon

EIDAuthenticate

User Guide

Smart card logon for standalone Windows computers

Version 2.0

My Smart Logon SAS
www.mysmartlogon.com · support@mysmartlogon.com

September 2026

About this document

This guide explains how to install EIDAuthenticate 2.0, associate a smart card with a Windows account, and use the card to sign in locally, to run programs under another account and to authenticate over the network with WinRM and SMB. It is written for end users and for the administrators who deploy the product.

Item	Value
Product	EIDAuthenticate Enterprise Edition 2.0
Document	User Guide, revision 2.0
Publisher	My Smart Logon SAS
Audience	End users, desktop support, system administrators
Companion document	EIDAuthenticate 2.0 Functional Specification (architecture, security model, policies)
Support	support@mysmartlogon.com

Revision history

Version	Date	Author	Changes
1.0	01/12/2012	Vincent Le Toux	Creation
1.1	23/02/2013	Frédéric Bourgeois	Update
1.2	19/04/2013	Nathan J. Lichtenstein	Review
1.3	22/07/2013	Vincent Le Toux	Update to EIDAuthenticate 1.0
1.4	25/01/2014	Vincent Le Toux	Update to EIDAuthenticate 1.2
1.5	12/01/2019	Vincent Le Toux	Update to EIDAuthenticate 1.6
1.5.1	24/03/2021	Vincent Le Toux	Revision
1.6.2	03/12/2021	Vincent Le Toux	Revision
1.7	08/09/2024	Vincent Le Toux	Windows 11 / Server 2022, kiosk mode, EIDCmd.exe
1.8	15/05/2026	Vincent Le Toux	Revision
2.0	28/09/2026	Vincent Le Toux	Complete rewrite for EIDAuthenticate 2.0: new layout and screenshots, network authentication (WinRM over Negotiate, SMB file shares), EIDRunAs, audit channel, new EIDCmd commands, Windows XP/2003 content removed

Conventions

Convention	Meaning
Bold	A button, a link or a label displayed on screen.
`Monospace`	A command, a file name, a registry value or text to type.

Convention	Meaning
NOTE / TIP	Additional information that makes a task easier.
WARNING / IMPORTANT	An action that can lock you out of a computer or weaken its security.
NEW IN 2.0	A feature introduced by EIDAuthenticate 2.0.

Contents

About this document **2**
Revision history 2
Conventions 2

1 Introduction **5**
1.1 What's new in version 2.0 5

2 System requirements **7**
2.1 Operating systems 7
2.2 Smart cards and readers 7
2.3 Network features 7

3 Installation **8**
3.1 Interactive installation 8
3.2 Silent installation 10
3.3 Upgrade, repair and removal 10

4 Associating a smart card with an account **11**
4.1 Starting the configuration wizard 11
4.2 The main window 11
4.3 Associating a certificate 12
4.4 Creating or importing a certificate on the card 15
4.5 Dissociating a certificate 16
4.6 Configuring another account 16

5 Security policies **18**
5.1 Smart card removal policy 18
5.2 Force smart card policy 19
5.3 Certificate revocation policy 19
5.4 Other settings 20

6 Using the smart card **22**
6.1 Signing in and unlocking 22
6.2 Elevation prompts and applications 22
6.3 Changing the PIN 23

7 Remote and network authentication **24**

7.1 Principles and prerequisites 24
7.2 Remote Desktop 25
7.3 PowerShell remoting and WinRM 25
7.4 Network file shares (SMB) 26

8 Running a program as another account (EIDRunAs) 27
8.1 Usage 27
8.2 Standard users and the EIDSecondaryLogon service 28
8.3 Hardening policy 28

9 Command-line administration 30
9.1 EIDCmd.exe 30
9.2 Configuration wizard switches 31
9.3 Kiosk modes 31

10 Auditing 32
10.1 Where to find the events 32
10.2 Event reference 33
10.3 Privacy and integrity 34

11 Troubleshooting 35
11.1 Common problems 35
11.2 Checking the card with certutil 35
11.3 Collecting traces with EID Log Manager 36
11.4 Capturing traces manually 38
11.5 Troubleshooting the installation 38

12 Appendix - registry reference 39

1 Introduction

EIDAuthenticate lets you sign in to a standalone Windows computer with a smart card and a PIN instead of a password. It brings the security of Active Directory smart card logon to computers that are not members of a domain, with no server, no directory and no changes to the Windows accounts.

The certificate stored on the card is **associated** with a local Windows account. From then on, the card and its PIN are enough to open that account: the private key never leaves the card, the PIN never leaves the computer, and Windows receives exactly the same session it would have created from the password, including access to the user's encrypted data (DPAPI, EFS, saved credentials).

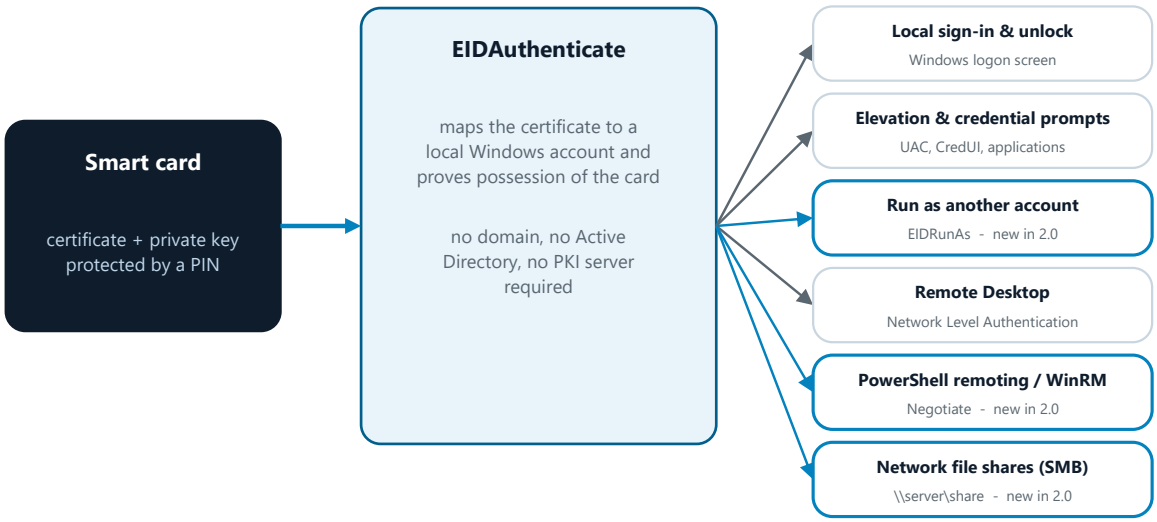


Figure 1 – What a smart card associated with EIDAuthenticate can be used for

1.1 What's new in version 2.0

Area	What changed
Network authentication	The card can now authenticate **PowerShell remoting / WinRM** over the Negotiate protocol (no CredSSP configuration required) and **SMB network file shares** (`\\server\share`), through a new kernel-mode security package. See chapter 7.
Run as another account	New **EIDRunAs** tool: start a program under another local account after authenticating that account's card - also from a standard user session, through the new EIDSecondaryLogon service. See chapter 8.
Audit trail	New **MySmartLogon-EIDAuthenticate/Operational** event channel with documented, stable event IDs: which card and which certificate opened which session, why a logon was refused, who changed an association. See chapter 10.
No reboot	The security packages (user mode and kernel mode) are registered live at installation: the product works immediately, without restarting the computer.

Area	What changed
Stored credential	The account password is no longer kept by the product: only its one-way hashes are stored, encrypted with the card, and re-keyed after every successful logon. Credentials created by 1.x are converted automatically at the next card logon.
EIDCmd	New commands `/LIST`, `/LISTALL`, `/REMOVEALL`, `/REQUIRESMARTCARD` and `/GETUSERNAME`.
User interface	Redesigned configuration wizard and EID Log Manager (live tracing window, persistent tracing, crash dumps), Start menu shortcut, installer available in 10 languages, repair no longer asks for the serial number.
Platforms	x64 and ARM64. Support for Windows XP and Windows Server 2003 (GINA) and for 32-bit Windows is discontinued.

2 System requirements

2.1 Operating systems

Windows version	Architecture	Status
Windows 11, Windows 10	x64, ARM64	Supported
Windows Server 2025, 2022, 2019, 2016	x64	Supported
Windows 8.1, Windows Server 2012 R2	x64	Supported for local sign-in (legacy)
Windows 7 SP1, Windows Server 2008 R2	x64	Supported for local sign-in (legacy)
Windows XP, Vista, Server 2003, all 32-bit editions	-	Not supported by version 2.0 (use 1.x)

EIDAuthenticate authenticates **local accounts**. On a computer joined to an Active Directory domain, domain accounts keep using the smart card logon built into Windows (Kerberos PKINIT); EIDAuthenticate only handles the local accounts of that computer.

2.2 Smart cards and readers

- The card must be usable by Windows: a **minidriver** compatible with the Microsoft Base Smart Card Crypto Provider, or a vendor **CSP**. Most PIV, GIDS, CAC and national eID cards qualify. Cards exposing only a PKCS#11 interface are not supported.
- Any CCID-compliant reader works. Readers with a PIN pad must be evaluated case by case.
- The certificate must carry an RSA key. It does not need to come from a public or corporate PKI: the wizard can create a private certification authority and the card certificate for you (section 4.4).

2.3 Network features

WinRM and SMB authentication (chapter 7) require EIDAuthenticate 2.0 on **both** the client and the server. Remote Desktop with Network Level Authentication also requires the product on both computers.

> **NOTE**
> EIDAuthenticate adds an authentication package to the Local Security Authority (LSA). When LSA protection (`RunAsPPL`) is enabled, Windows only loads packages signed for that purpose. If the package is reported as not loaded after installation, check the LSA protection setting with your support contact before disabling anything.

3 Installation

Installing EIDAuthenticate requires administrator rights. The package is a Windows Installer (MSI) file; the same file is used for an interactive installation and for a silent deployment.

3.1 Interactive installation

1. Run `EIDAuthenticateEnterpriseEdition_x64.msi` (`_arm64` on ARM computers). The setup wizard opens; click **Next**.

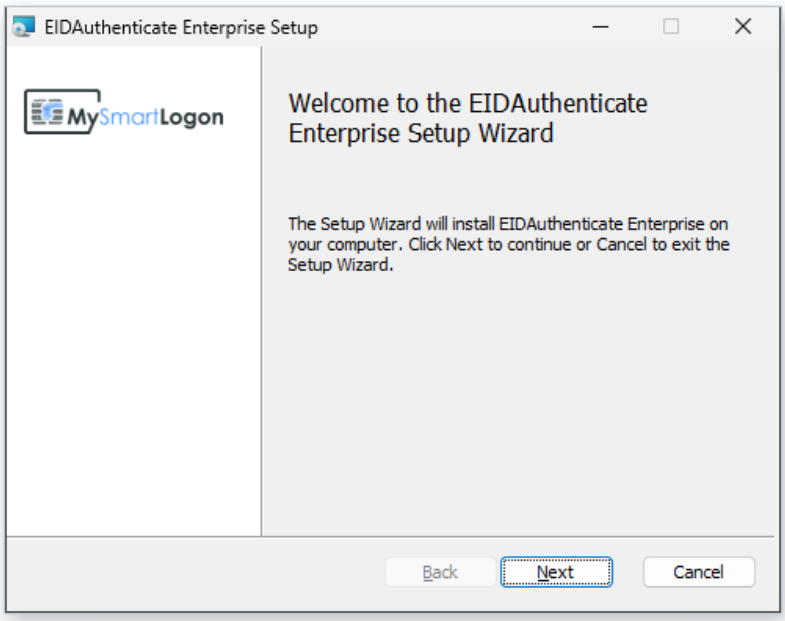


Figure 2 – Welcome page of the setup wizard

2. Read the license agreement, select **I accept the terms in the License Agreement** and click **Next**.

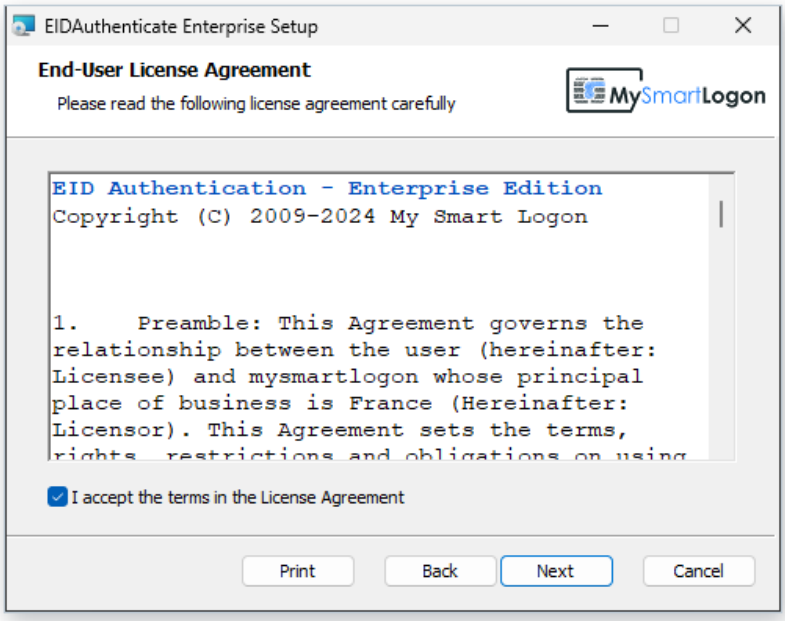


Figure 3 – End-user license agreement

3 Type the serial number delivered with your license: three groups of eight characters, one per box. If you do not have one yet, contact@mysmartlogon.com can issue a trial serial. Click **Next**.

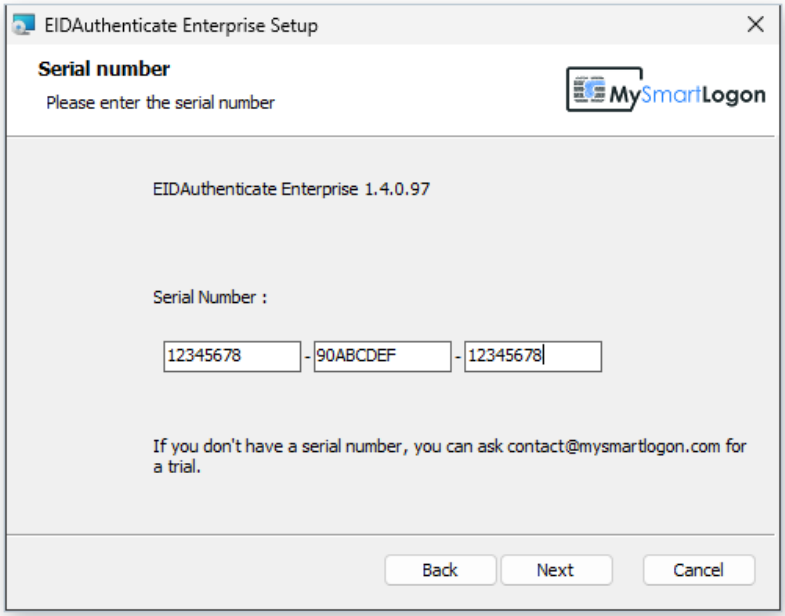


Figure 4 – Serial number page (the value shown is an example)

The serial number is checked immediately. A serial generated for a workstation cannot be used on a server edition of Windows (and vice versa); an invalid serial displays an error and the wizard stays on the same page. A trial serial displays the date at which the trial ends.

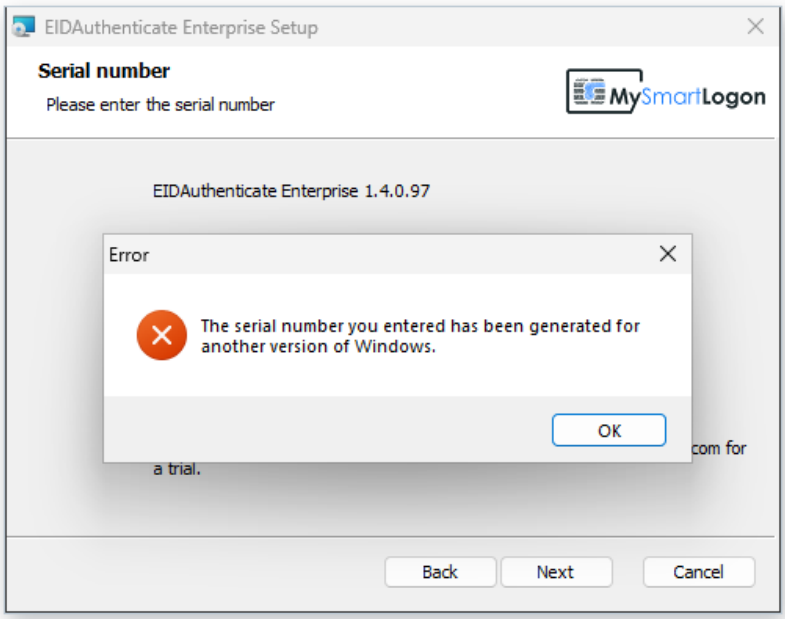


Figure 5 – Error displayed for a serial issued for another edition of Windows

4 Click **Install**, accept the User Account Control prompt, then **Finish** once the installation completes.

> **NEW IN 2.0**
> No restart is needed. EIDAuthenticate 2.0 registers its authentication package into the running LSA and starts its kernel-mode package immediately, so smart card logon, WinRM and SMB authentication work as soon as the installer closes. If live registration is not possible on a given system, the installer asks for a restart.

After installation, the product adds:

- a **Smart Card Logon** entry in Control Panel > System and Security, and a Start menu shortcut **EIDAuthenticate - Smart Card Logon**;
- a **Smart card credential** sign-in option on the Windows logon screen;
- the command-line tools `EIDCmd.exe`, `EIDRunAs.exe` and `EIDLogManager.exe` in `C:\Program Files\EIDAuthenticate`;
- the **EIDSecondaryLogon** service (manual start), used by EIDRunAs;
- the **MySmartLogon-EIDAuthenticate/Operational** event log.

3.2 Silent installation

For a deployment with a software distribution tool or a script, run msiexec from an elevated prompt:

INSTALL

```
msiexec /i EIDAuthenticateEnterpriseEdition_x64.msi /qn /norestart /l*v install.log
SERIALNUMBERPROPERTY=<serial number>
```

UNINSTALL

```
msiexec /x EIDAuthenticateEnterpriseEdition_x64.msi /qn /norestart /l*v uninstall.log
```

REPAIR

```
msiexec /fa EIDAuthenticateEnterpriseEdition_x64.msi /qn /l*v repair.log
```

Exit code 0 means success and 3010 means success with a restart required. Any other code is described in the log file given with `/l*v`.

3.3 Upgrade, repair and removal

- **Upgrade** - install the new MSI over the existing version. Associations, policies and the license are kept. Installing an older version over a newer one is refused.
- **Repair** - from Settings > Apps or with `msiexec /fa`. Version 2.0 no longer asks for the serial number during a repair.
- **Removal** - from Settings > Apps or with `msiexec /x`. Removing the product deletes every stored association; the Windows accounts and their passwords are not changed. A restart completes the removal.

> **TIP**
> Checking the installation: `EIDCmd /LICENSE` shows the serial number status, and the event **1900 - Package initialized** in the audit channel shows the exact version loaded by LSA.

4 Associating a smart card with an account

The configuration wizard associates a certificate stored on a smart card with a Windows account. Any user can configure his or her own account; configuring another account requires administrator rights.

4.1 Starting the configuration wizard

Use one of the following entry points:

- Start menu: **EIDAuthenticate - Smart Card Logon**.
- Control Panel > **System and Security** > **Smart Card Logon**. The same section gives direct access to the removal, force smart card and revocation policies (chapter 5).
- Command line: `"C:\Program Files\EIDAuthenticate\EIDConfigurationWizard.exe"` (see section 9.2 for its switches).

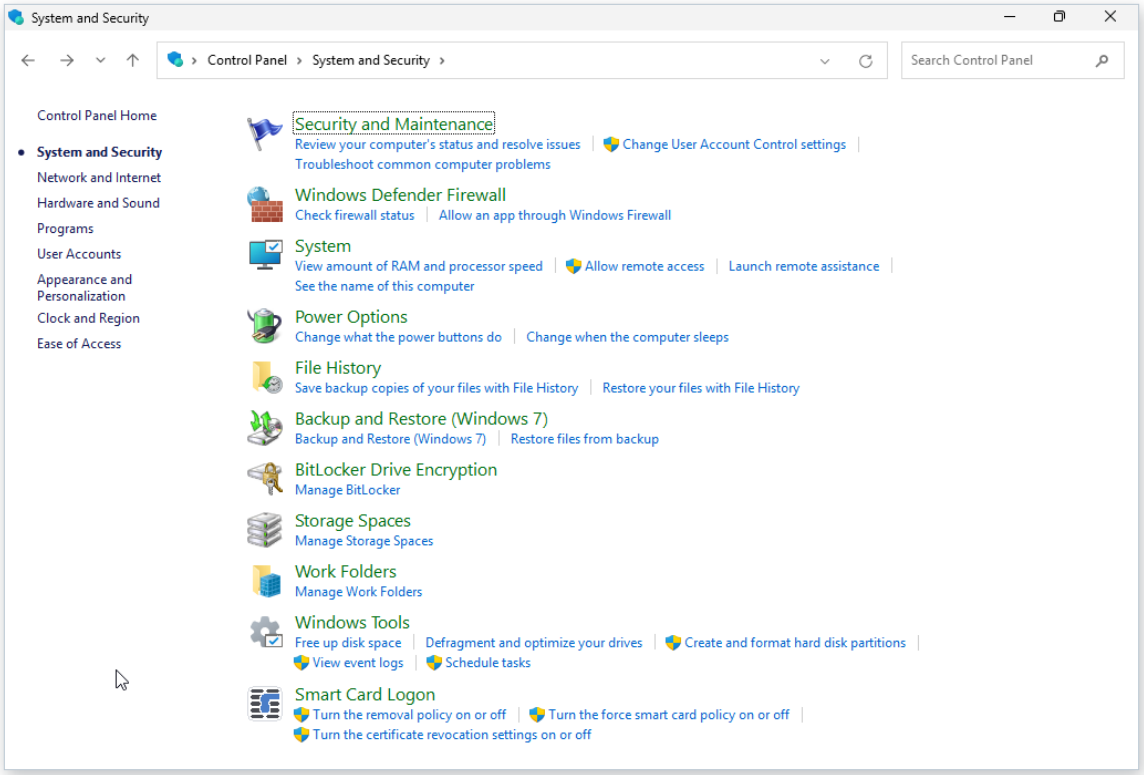


Figure 6 – The Smart Card Logon section of Control Panel

4.2 The main window

The main window lists the certificates already associated with the account named in the upper-right corner, together with the product version. From here you can associate a new certificate, dissociate the selected one, configure another account or change the machine-wide policies.

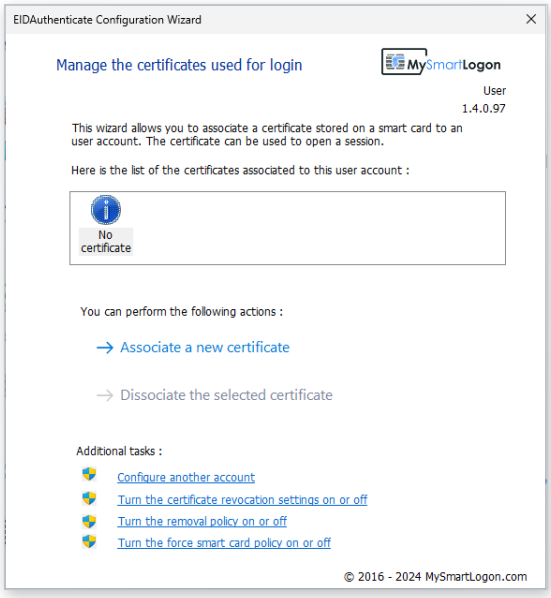


Figure 7 – No certificate associated yet

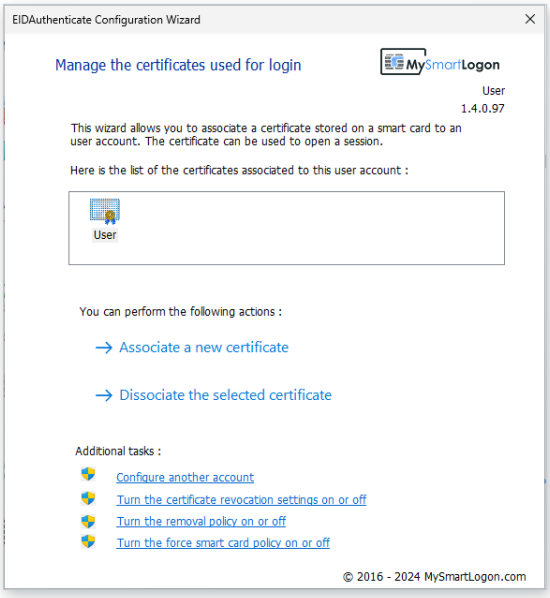


Figure 8 – A certificate associated with the account

4.3 Associating a certificate

1. Insert the smart card and click **Associate a new certificate**. The wizard reads the card and shows the certificates it contains. Double-click a certificate to display its details; click **Refresh** after changing cards.
2. Select the certificate to use and click **Next**. If the card is empty, use **Create or Import a certificate** first (section 4.4).

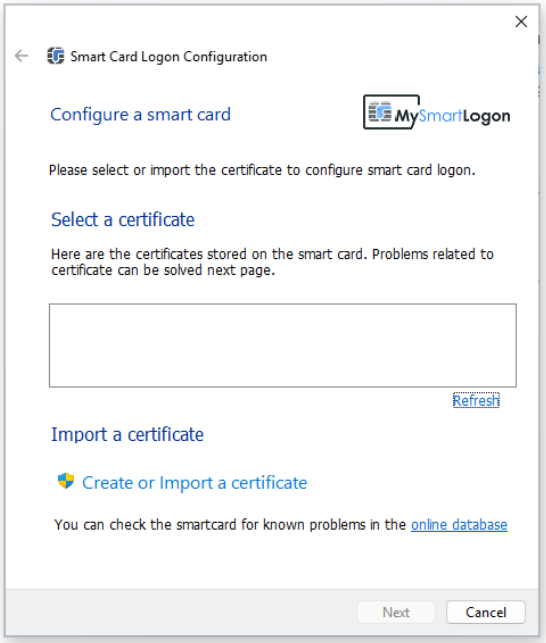


Figure 9 – Empty card: create or import a certificate

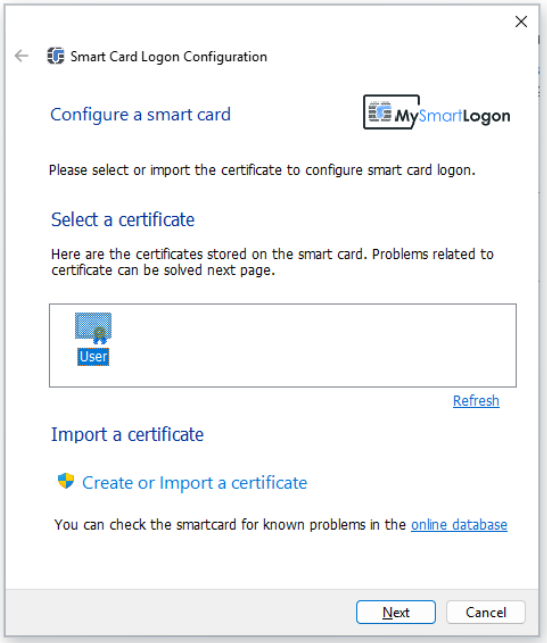


Figure 10 – Selecting the certificate stored on the card

3 The wizard checks the certificate against the current security policy: encryption capability, trust, key usage and time validity. Every check must be green to continue. A problem can be fixed from the link displayed under the check (for example *Do not check the EKU*); changing a policy requires administrator rights.

4 Type the **password of the Windows account** - not the PIN. This proves that you are entitled to configure the account and lets the product open the same session as a password logon. Leave the field empty if the account has no password.

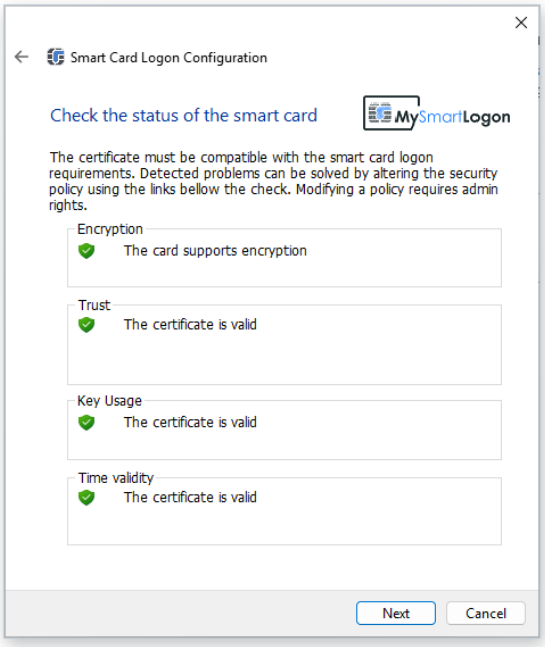


Figure 11 – All checks passed

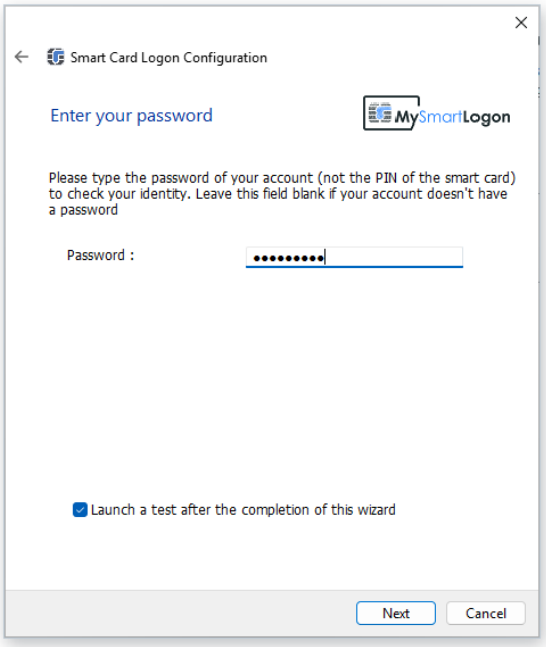


Figure 12 – Account password

A wrong password is reported immediately and the association is not created:

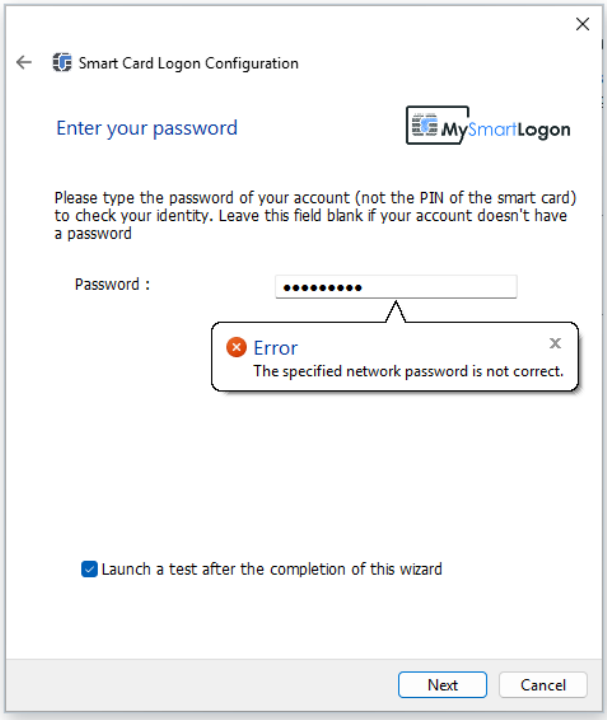


Figure 13 – Wrong account password

5 Keep **Launch a test after the completion of this wizard** selected and click **Next**. Windows Security asks for the PIN, exactly like the logon screen will. Type it and press Enter.

6 The last page gives the result of the test. Click **Finish**. The card can now be used to sign in.

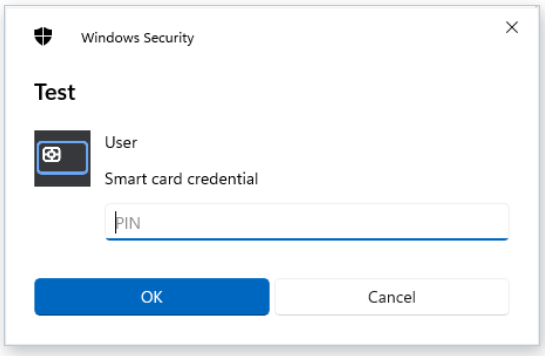


Figure 14 – PIN requested by the test

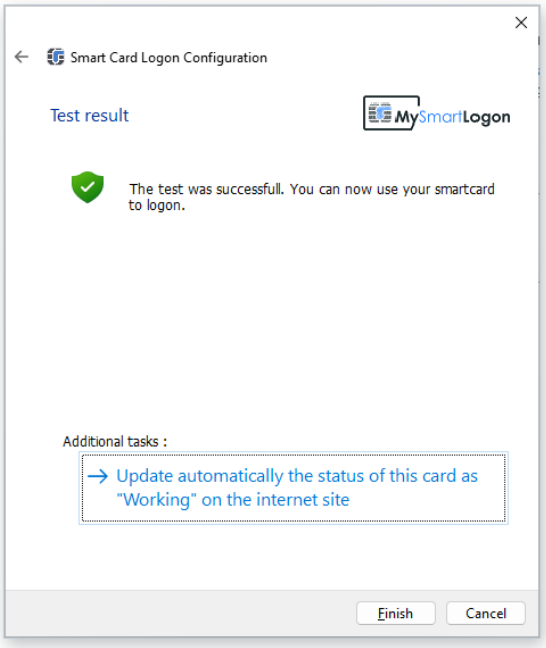


Figure 15 – Successful test

If the test fails, the page shows the reason (here a wrong PIN) and offers to send a report to the support team. The association itself has been saved: fix the cause and run the test again from the main window.

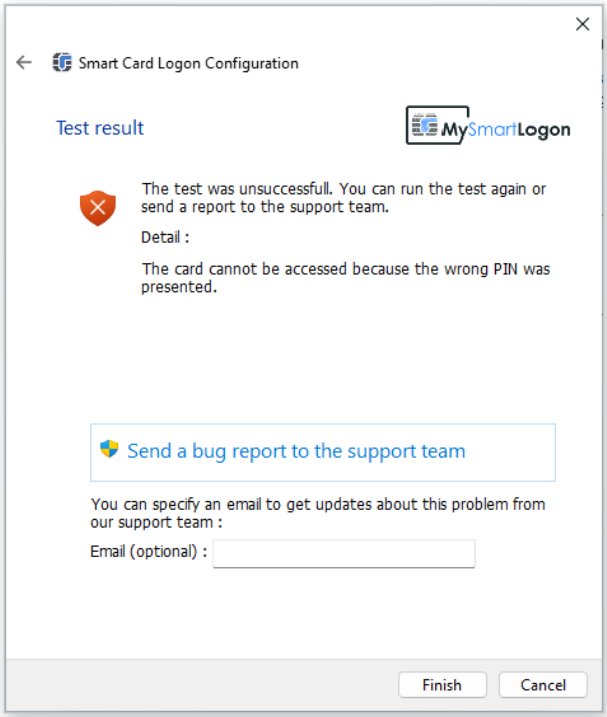


Figure 16 – Failed test with its reason

> **NOTE**
>
> EIDAuthenticate validates the certificate against the **computer** certificate stores (`certlm.msc`), not the user stores. When the chain cannot be built, the wizard offers to place the certificate itself in the computer **Trusted People** store.
>
> If the security policy is relaxed or tightened **after** a certificate has been associated, the result of the next logon depends on the new policy.

4.4 Creating or importing a certificate on the card

Create or Import a certificate opens an elevated page that prepares the card when it does not yet hold a suitable certificate:

- **Create a new certification authority** - the wizard creates a private root certificate on the computer and uses it to issue the card certificate (default the first time).
- **Use this certification authority** - reuse an existing authority whose private key is available (**Select an authority** / **Show the certificate**).
- **Import a p12 file into the smart card** - load an existing certificate and its private key from a PKCS#12 (.pfx/.p12) file protected by the given password.
- **Delete all certificates on the smart card before processing** - empty the card first.
- **Key length** - 1024, 2048 (default), 3072 or 4096 bits for the generated key.

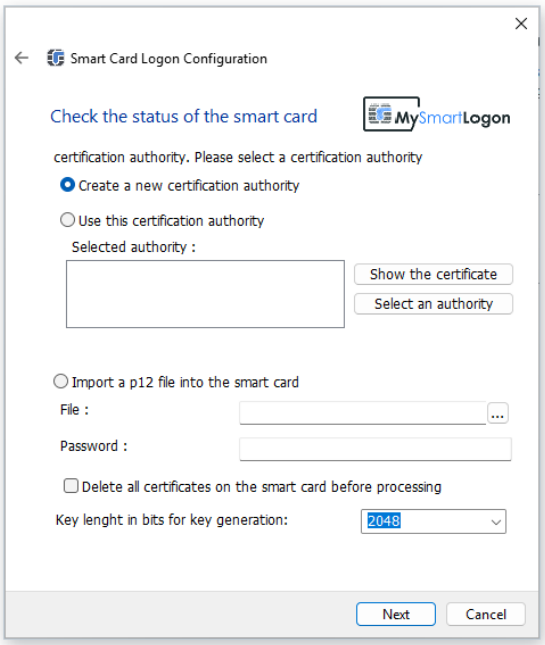


Figure 17 – Create or import a certificate

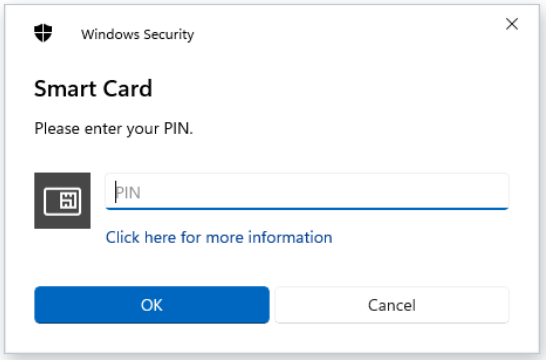


Figure 18 – The card PIN is required to write the key

Writing a key on the card requires its PIN; an error is displayed if the card refuses the operation (for example a card that only accepts keys generated by its issuer).

4.5 Dissociating a certificate

Select the certificate in the main window and click **Dissociate the selected certificate**. Repeat for each certificate: smart card logon is disabled for the account once no certificate remains. The command-line equivalent is `EIDCmd /DELETE` (chapter 9).

> **WARNING**
> A certificate can be associated with **one account only**. Associating it with a second account silently removes it from the first one.

4.6 Configuring another account

1. Click **Configure another account** in the main window.
2. Type the account name in the standard **Select User** dialog and click **OK**, then accept the elevation prompt.
3. The wizard reopens for that account: its name is displayed in the upper-right corner. Associate a certificate as described above; the password asked is the password of that account.

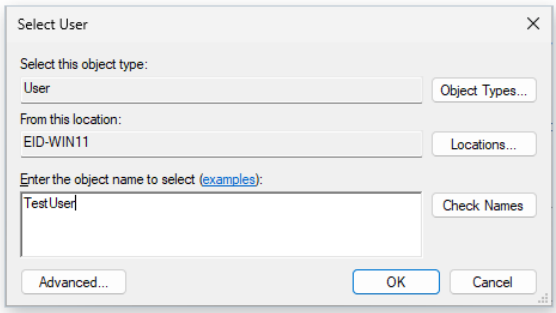


Figure 19 – Choosing the account

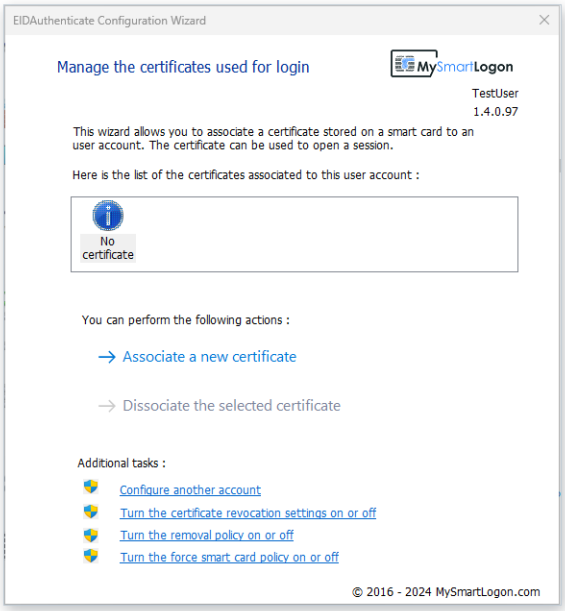


Figure 20 – Wizard opened for the account TestUser

NOTE

A disabled account (such as the built-in Administrator) can be configured, but the logon fails until the account is enabled.

5 Security policies

Three machine-wide policies can be changed from the wizard or from Control Panel. They use the same registry values as the corresponding Windows policies, so they can also be deployed by Group Policy or a script. Changing them requires administrator rights.

5.1 Smart card removal policy

Defines what happens when the card used to open the session is removed from the reader.

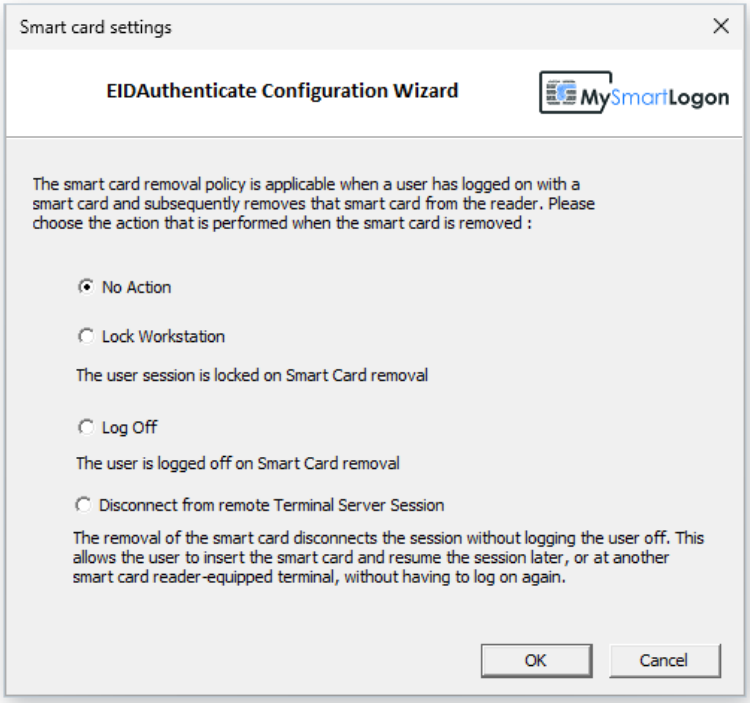


Figure 21 – Smart card removal policy

Option	Effect
No Action	Nothing happens (default).
Lock Workstation	The session is locked; the user unlocks it with the card.
Log Off	The user is signed out. Unsaved work is lost.
Disconnect from remote Terminal Server Session	A remote session is disconnected and can be resumed later, from this or another computer, without signing in again. A local session is locked.

The policy relies on the Windows **Smart Card Removal Policy** service, which the wizard starts and sets to automatic when a policy other than *No Action* is chosen. The policy applies to sessions opened after the change: sign out and sign in again to activate it.

5.2 Force smart card policy

Requires every user of the computer to sign in with a smart card (Windows policy *Interactive logon: Require smart card*).

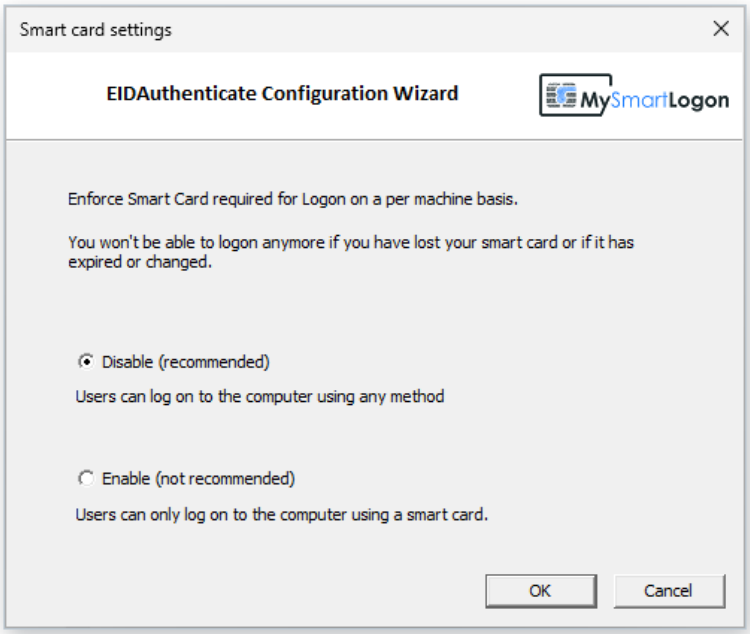


Figure 22 – Force smart card policy

> **IMPORTANT**
> Once this policy is enabled, a lost, blocked, expired or replaced card prevents the owner from signing in. Make sure at least one administrator account has a working card before enabling it.

When the policy is active and a password logon is attempted, Windows refuses it and asks for a smart card.

5.2.1 Emergency procedure

The logon screen displays a **Disable Force Smart Card Logon Policy** link while the policy is active. It lets an administrator authenticate with a password and turn the policy off, and it offers the Windows password reset wizard when a reset disk exists. Set `scdisablerecovery` = 1 (DWORD) under `HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\System` to remove this link.

As a last resort the policy can be disabled offline: connect the system disk to another computer, load the `%windir%\System32\config\SOFTWARE` hive in regedit (**File > Load Hive**), set `Microsoft\Windows\CurrentVersion\Policies\System\scforceoption` to 0, unload the hive and reconnect the disk. Full disk encryption (BitLocker) prevents this recovery by an attacker.

5.3 Certificate revocation policy

Defines whether the revocation status of the certificate is checked at logon, using the CRL or OCSP addresses published in the certificate.

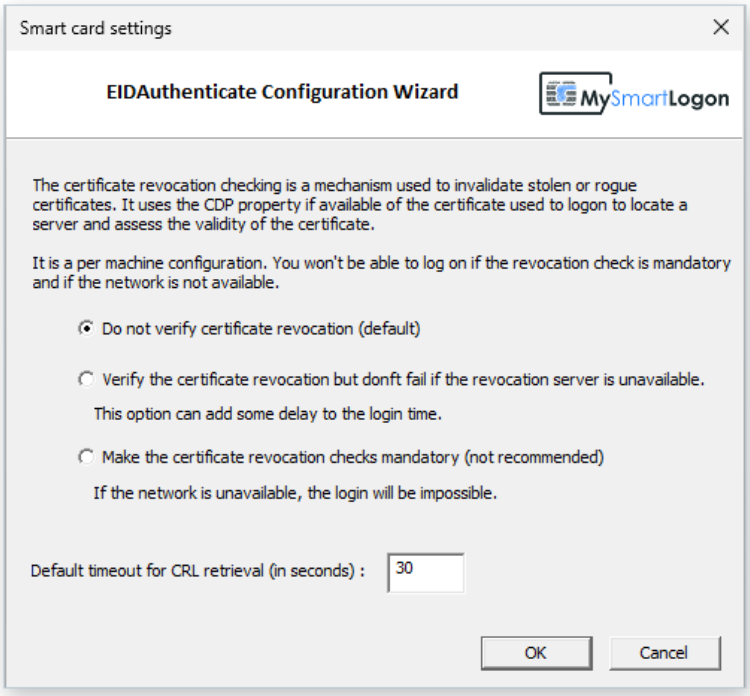


Figure 23 – Certificate revocation policy

Option	Effect	ForceCRLCheck
Do not verify certificate revocation (default)	No revocation check.	0
Verify the certificate revocation but don't fail if the revocation server is unavailable	A revoked certificate is refused; the logon succeeds when no revocation information can be obtained.	1
Make the certificate revocation checks mandatory	The logon fails when no fresh or cached revocation information is available.	2

The timeout for downloading revocation data defaults to 30 seconds. Revocation data is cached by Windows, so a successful check keeps working offline until the cached CRL expires.

> **WARNING**
>
> Mandatory checks block every logon while the revocation server is unreachable - for example on a laptop without network or behind a misconfigured proxy.
>
> Revocation data is downloaded by the SYSTEM account, whose proxy is not the Internet Explorer proxy. Display it with `netsh winhttp show proxy` and set it with `netsh winhttp set proxy <server:port>`. To test a certificate as SYSTEM: `psexec -s certutil -urlfetch -verify user.cer`.

5.4 Other settings

5.4.1 Windows Safe Mode

Windows loads only its built-in password provider in Safe Mode, so the smart card cannot be used there. This is by design: Safe Mode is a recovery environment. To load third-party credential providers in Safe Mode anyway, set `ProhibitFallbacks` = 1 (DWORD) under `HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\Credential Providers`.

5.4.2 Showing the smart card first on the logon screen

Windows proposes the sign-in method used last. When the policy *Interactive logon: Don't display last signed-in* is enabled, the password prompt can take precedence. To always propose the smart card for new sessions, set `LastLoggedOnProvider` to `{FC866A0A-DB08-4835-A26F-414B46F3244C}` under `HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\LogonUI` and deny the *Set Value* permission on that key to SYSTEM (**Edit > Permissions > Advanced**).

6 Using the smart card

6.1 Signing in and unlocking

Insert the card. The logon screen shows the account associated with the certificate and a **Smart card credential** field: type the PIN and press Enter. If the password field is displayed instead, click **Sign-in options** and select the smart card icon.

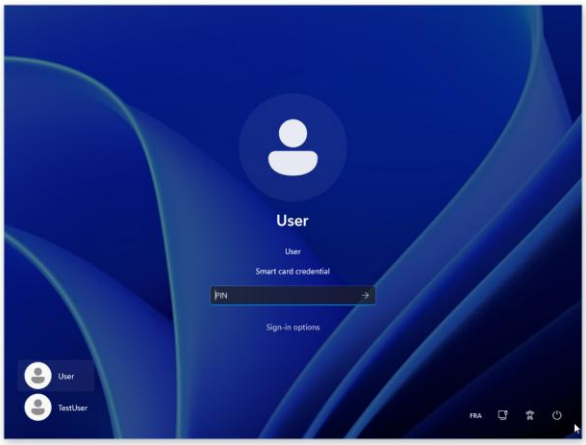


Figure 24 – Smart card sign-in

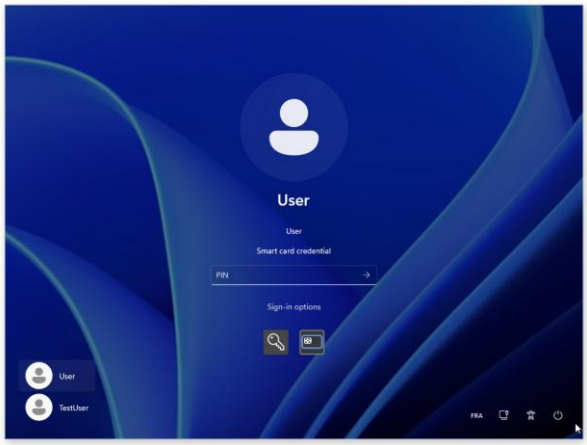


Figure 25 – Sign-in options: password or smart card

The same screen is used to unlock a locked session. When the computer is locked, only the certificates of the signed-in user are offered; use **Switch user** to sign in with a card associated with another account.

A wrong PIN is reported on screen. Each wrong PIN decrements the counter of the card, which blocks itself after the number of attempts set by its issuer; a blocked card must be unblocked with its administrator key (PUK) using the issuer's tool.

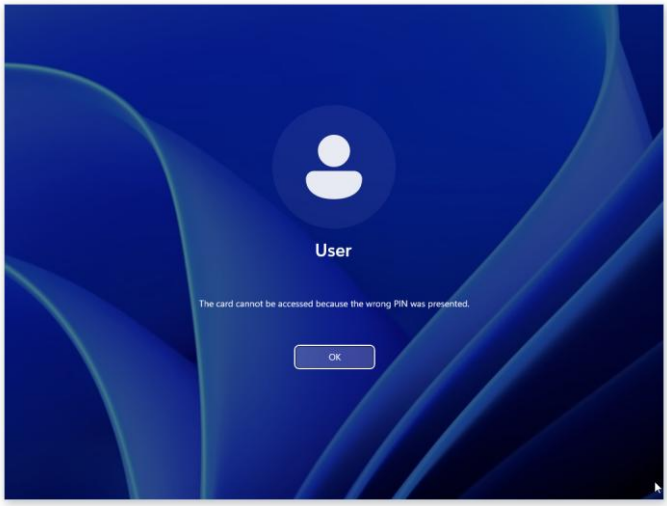


Figure 26 – Wrong PIN at sign-in

6.2 Elevation prompts and applications

The card is also offered by every Windows credential prompt (CredUI): elevation prompts when the policy *User Account Control: Behavior of the elevation prompt for administrators* is set to *Prompt for credentials*,

Remote Desktop, mapped drives and any application that uses `CredUIPromptForWindowsCredentials`. The configuration wizard's test (Figure 16) uses exactly this prompt.

> **TIP**
> Developers can call `LsaLogonUser` with the credential returned by the prompt to open an interactive session, or use SSPI (`AcquireCredentialsHandle`, `InitializeSecurityContext`, `AcceptSecurityContext`) with the Negotiate package for a client/server protocol. Contact support@mysmartlogon.com for integration guidance.

6.3 Changing the PIN

For cards using a minidriver, Windows can change the user PIN (not the administrator key):

1. Press **Ctrl+Alt+Del** and choose **Change a password**.
2. Click **Sign-in options** and select the smart card reader icon.
3. Type the current PIN, then the new PIN twice, and press Enter.

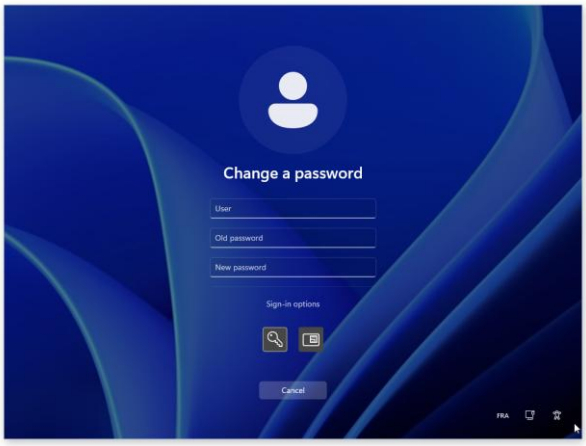


Figure 27 – Change a password: sign-in options

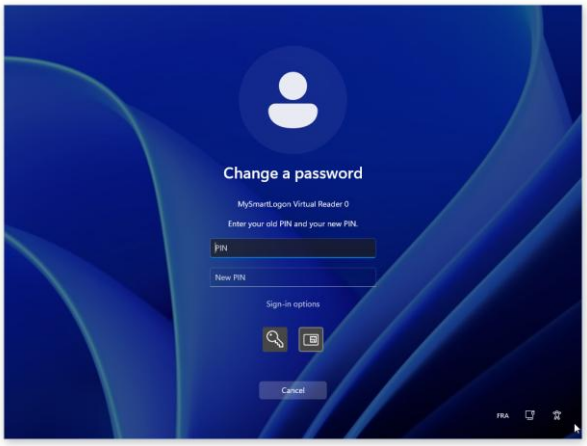


Figure 28 – Changing the PIN of the card in the reader

Cards that rely on their own CSP must use the PIN tool provided by the card vendor.

7 Remote and network authentication

EIDAuthenticate 2.0 can use the card to authenticate to another computer: Remote Desktop, PowerShell remoting (WinRM) and network file shares (SMB). The card and the PIN stay on the client; only a signature computed by the card crosses the network.

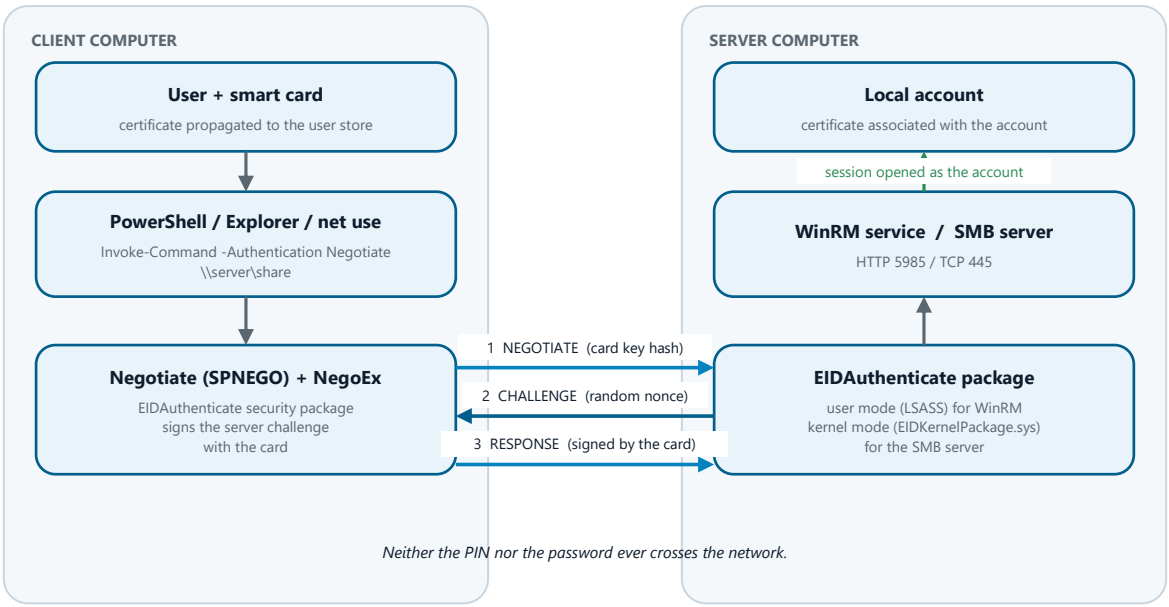


Figure 29 – Network authentication with the smart card (WinRM and SMB)

7.1 Principles and prerequisites

- EIDAuthenticate 2.0 is installed on **both** computers.
- On the **server**, the certificate of the card is associated with the local account the user must obtain there - with the wizard, or with `EIDCmd /USER <account> /NEW /CERTIFICATE user.cer` from the exported certificate (`EIDCmd /SAVECERTIFICATES` on the client exports it). The server identifies the card by its certificate: no password or PIN is stored or sent.
- On the **client**, the certificate is also associated with the local account of the user, and the card is in the reader. Windows copies the card certificate into the user's personal store (Certificate Propagation service).
- The two computers can reach each other (WinRM: TCP 5985; SMB: TCP 445) and the server firewall allows the service.

> **IMPORTANT**
>
> A network logon is decided by the **association** on the server. The revocation, expiry and trust policies of chapter 5 are evaluated on the computer where the card is inserted, not by the server. To revoke a card's remote access, dissociate its certificate on the server (`EIDCmd /USER <account> /DELETE /CERTIFICATE user.cer`).

7.2 Remote Desktop

With Network Level Authentication (NLA), the Remote Desktop client authenticates the user before the remote session is created. When the connection prompt appears, choose **More choices** and the smart card, then type the PIN. The server authenticates the card with EIDAuthenticate and opens the session of the associated account.

Without NLA, the remote logon screen is displayed inside the session and the card is used exactly like a local logon, through smart card redirection. To connect without NLA, clear *Allow connections only from computers running Remote Desktop with Network Level Authentication* on the server and add `enablecredsspsupport:i:0` to the .rdp file on the client.

7.3 PowerShell remoting and WinRM

> **NEW IN 2.0**
> EIDAuthenticate 2.0 authenticates WinRM with the standard **Negotiate** protocol. Unlike CredSSP it requires no delegation policy and works with every card: the signature is computed by the card through the normal Windows cryptographic stack.

7.3.1 Preparing the server

SERVER - ELEVATED POWERSHELL

```
Enable-PSRemoting -Force
# Local accounts that are members of Administrators are filtered by UAC over the network.
# To administer the server with such an account, lift the remote UAC restriction:
New-ItemProperty -Path HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System `
    -Name LocalAccountTokenFilterPolicy -Value 1 -PropertyType DWord -Force
```

7.3.2 Preparing the client

Between computers that are not in a domain, WinRM only accepts Negotiate towards trusted hosts:

CLIENT - ELEVATED POWERSHELL

```
Set-Item WSMan:\localhost\Client\TrustedHosts -Value 'server01' -Concatenate -Force
```

7.3.3 Connecting with the card

The card is given to PowerShell as a credential whose user name designates the certificate (the Windows *marshalled certificate credential*) and whose password is the PIN. The following script builds that credential for the certificate of the card and runs a command on the server:

CLIENT - POWERSHELL

```
Add-Type -Namespace EID -Name Cred -MemberDefinition @'
[StructLayout(LayoutKind.Sequential)] public struct CERT_CREDENTIAL_INFO {
    public uint cbSize;
    [MarshalAs(UnmanagedType.ByValArray, SizeConst = 20)] public byte[] rgbHashOfCert; }
[DllImport("advapi32.dll", CharSet = CharSet.Unicode, SetLastError = true)]
public static extern bool CredMarshalCredential(int type, ref CERT_CREDENTIAL_INFO cred, out
IntPtr marshaled);
[DllImport("advapi32.dll")] public static extern void CredFree(IntPtr buffer);
'@

# The card certificate, propagated to the user's personal store
$cert = Get-ChildItem Cert:\CurrentUser\My | Where-Object { $_.HasPrivateKey } | Out-GridView -
```

```
PassThru
$info = New-Object EID.Cred+CERT_CREDENTIAL_INFO
$info.cbSize = [Runtime.InteropServices.Marshal]::SizeOf($info)
$info.rgbHashOfCert = $cert.GetCertHash()
$ptr = [IntPtr]::Zero
[void][EID.Cred]::CredMarshalCredential(1, [ref]$info, [ref]$ptr)   # 1 = CertCredential
$user = [Runtime.InteropServices.Marshal]::PtrToStringUni($ptr); [EID.Cred]::CredFree($ptr)

$cred = New-Object PSCredential($user, (Read-Host 'Smart card PIN' -AsSecureString))
Invoke-Command -ComputerName server01 -Authentication Negotiate -Credential $cred -ScriptBlock {
whoami }
Enter-PSSession -ComputerName server01 -Authentication Negotiate -Credential $cred
```

The remote command runs under the account associated with the certificate on the server. The server records events 1005 and 1006 in its audit channel (chapter 10).

7.4 Network file shares (SMB)

> **NEW IN 2.0**
>
> Windows file servers authenticate SMB sessions in the kernel. EIDAuthenticate 2.0 installs a kernel-mode security package, **EIDKernelPackage**, started automatically, so that `\\server\share` accepts the card.

Share the folder on the server as usual (for example with `New-SmbShare`) and grant access to the associated account. On the client, connect with the same marshalled credential as for WinRM:

CLIENT - POWERSHELL

```
# $user and the PIN as built in the previous section
New-SmbMapping -LocalPath Z: -RemotePath \\server01\Finance -UserName $user `
               -Password ([Net.NetworkCredential]::new('', (Read-Host 'PIN' -
AsSecureString)).Password)
Get-ChildItem Z:
```

> **NOTE**
>
> Do not pass the marshalled user name to `net use`: net.exe interprets the `@@` prefix as an e-mail address and alters it. Use `New-SmbMapping`, Explorer's credential prompt or the Win32 function `WNetAddConnection2`.
>
> Connections to a share are kept per logon session; disconnect with `Remove-SmbMapping -LocalPath Z:`.

8 Running a program as another account (EIDRunAs)

EIDRunAs is the smart card counterpart of the Windows `runas` command: it starts a program under another local account after authenticating that account's card. No password is typed or stored.

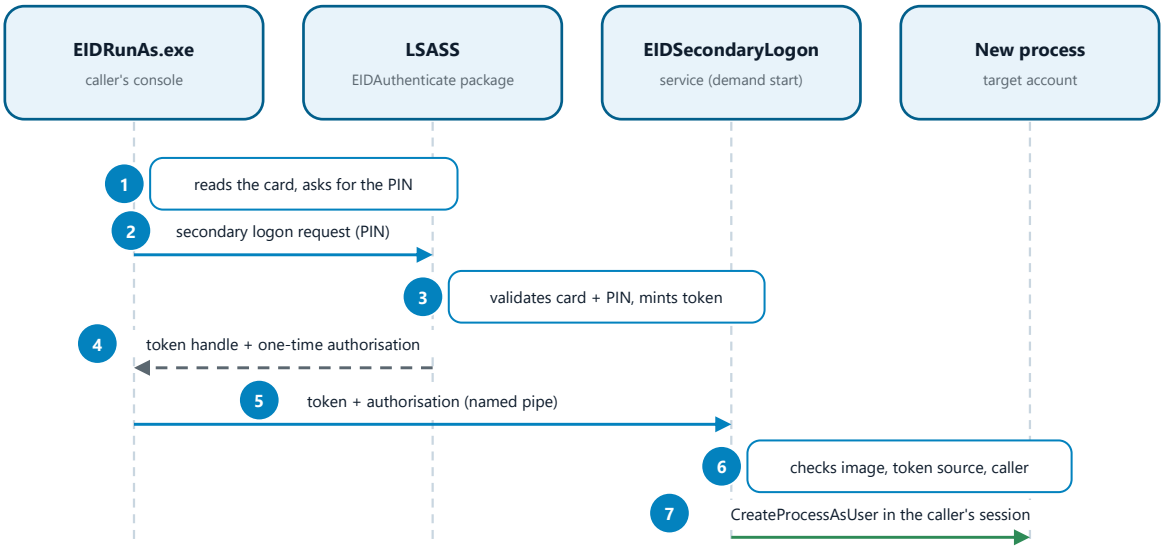


Figure 30 – How EIDRunAs starts a program under another account

8.1 Usage

SYNTAX

```
EIDRunAs /user:USER program [arguments]
EIDRunAs /user:USER "program arguments"

Examples:
  EIDRunAs /user:administrator mmc %windir%\system32\compmgmt.msc
  EIDRunAs /user:administrator notepad C:\Windows\System32\drivers\etc\hosts
  EIDRunAs /user:administrator "cmd /k whoami"

EIDRunAs /install      register the secondary logon service (elevated, once)
EIDRunAs /uninstall    remove it
```

1. Insert the card associated with the target account and run EIDRunAs from a command prompt.
2. EIDRunAs displays the program it resolved, the arguments and the account, looks for the matching certificate in the readers and asks for the PIN.
3. The program starts in the target account's profile directory, in its own window. The exit code of the program is reported when it ends.

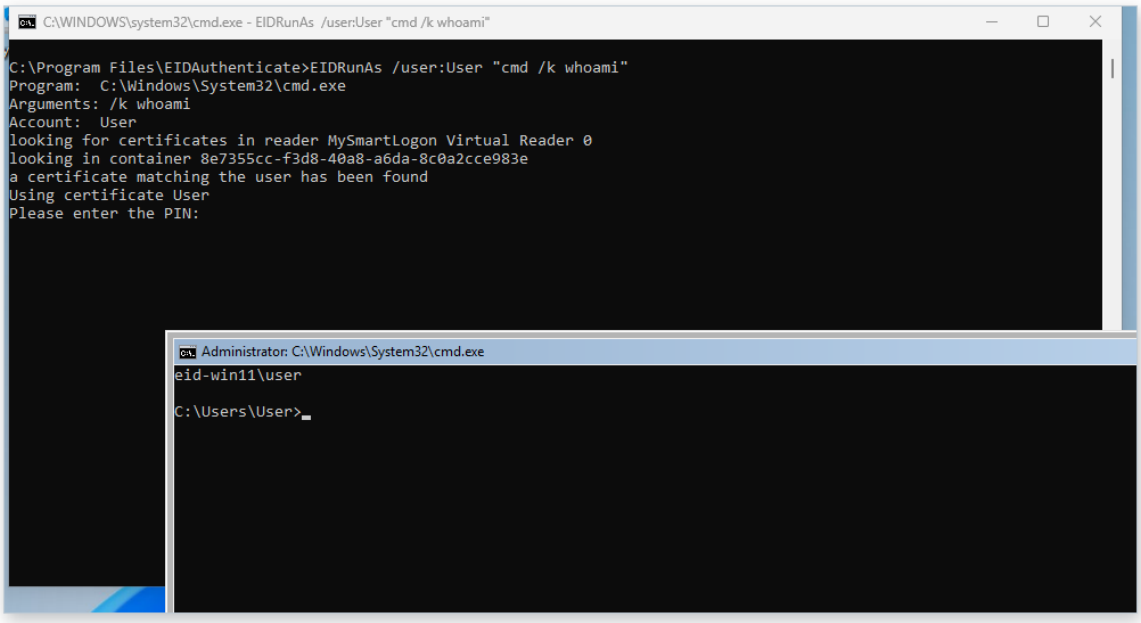


Figure 31 – EIDRunAs asks for the PIN in the calling console; the program starts in its own window under the target account

> **NOTE**
>
> The program runs in a separate window: the output of a console command such as `whoami` is not written in the console that started EIDRunAs. Keep the window open with `cmd /k`, or redirect the output inside the command: `EIDRunAs /user:admin "cmd /c whoami > %PUBLIC%\out.txt"`.
>
> A process that cannot start (for example missing access to the desktop) is reported as a startup failure with its NTSTATUS code rather than as a result.

8.2 Standard users and the EIDSecondaryLogon service

Creating a process for another account requires a privilege that standard users do not hold. When EIDRunAs is run by an administrator, it starts the program directly. From a standard user session, it hands the token minted by EIDAuthenticate to the **EIDSecondaryLogon** service, which only creates the process in the caller's session. The PIN never reaches the service and the service never touches the card; it refuses any token that was not issued by EIDAuthenticate, any authorisation issued for another command or another process, and any authorisation that was already used or has expired.

The service is installed by the MSI (manual start, started on demand). `EIDRunAs /install` registers it on a computer where EIDRunAs was deployed without the MSI.

8.3 Hardening policy

Because it can run programs with administrative rights, the service checks the program before starting it: the file is opened and locked for the whole operation, every junction and symbolic link is resolved, and the resolved path is what is validated and launched. The checks are configured under `HKLM\Software\MySmartLogon\EIDAuthenticate\SecondaryLogon`:

Value	Type	Default	Effect
`Enforcement`	DWORD	0	0: a failed check is audited (event 1301) but the program starts. 1: the program is refused.

Value	Type	Default	Effect
`AllowNetworkImages`	DWORD	0	1 allows programs located on a network share.
`AllowRemovableImages`	DWORD	0	1 allows programs located on a removable drive.
`RequireSignedImage`	DWORD	0	1 refuses programs without a valid Authenticode signature.
`AllowStdinRelay`	DWORD	0	1 allows relaying the console input to the program.
`AllowedImageRoots`	MULTI_SZ	(empty)	Folders trusted even if a non-administrator could modify them.

By default a program that a non-administrator could replace is only audited. Review the 1301 events, adjust `AllowedImageRoots`, then set `Enforcement` to 1. When EIDRunAs is refused, it prints the exact `reg add` command that would allow the operation.

9 Command-line administration

9.1 EIDCmd.exe

`EIDCmd.exe`, in `C:\Program Files\EIDAuthenticate`, performs the configuration tasks of the wizard without user interface, for scripting and mass deployment. Commands using `/USER`, `/LISTALL` or `/REMOVEALL` must run elevated. Put an argument between double quotes when it contains spaces.

Command	Description
`[/USER u] [/PASSWORD p] [/SIGNATURE] /NEW /CERTIFICATE file.cer`	Associate a certificate with an account, from the certificate file only - the card does not need to be present. Without /PASSWORD the password is asked on a secure prompt (type "" for an empty password). /SIGNATURE stores a credential usable by signature-only cards.
`[/USER u] /DELETE /CERTIFICATE file.cer`	Remove the association of this certificate.
`[/USER u] /SAVECERTIFICATES [/DIRECTORY dir]`	Export the certificates associated with the account as .cer files (audit, server enrolment).
`[/USER u] /LIST`	List the certificates associated with the account, with their SHA-1 thumbprint. **New**
`/LISTALL`	List the associations of every local account. **New**
`/REMOVEALL`	Remove every association on the computer. **New**
`[/USER u] /REQUIRESMARTCARD ON|OFF`	Set or clear the account flag *Smart card is required for interactive logon*. **New**
`/GETUSERNAME /CERTIFICATE file.cer`	Show the account a certificate is associated with. **New**
`[/USER u] [/PASSWORD p] /KIOSK`	Configure the account in generic kiosk mode (section 9.3).
`/CREATESECUREACCOUNT <CACID> /FNAME f /LNAME l [/PASSWORD p]`	Create an account bound to a CAC/PIV identifier (section 9.3).
`/LICENSE`	Show the serial number and whether it is valid.
`/UPDATELICENSE <serial>`	Replace the serial number (format 12345678-12345678-12345678).

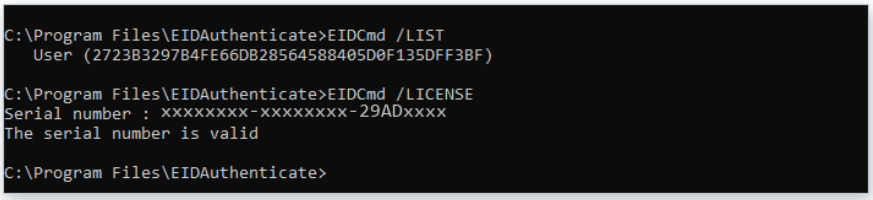


Figure 32 – EIDCmd /LIST and /LICENSE (serial number masked)

EXAMPLES

```
REM Associate the exported certificate of a user with the account "jdoe" on a server
EIDCmd /USER jdoe /NEW /CERTIFICATE C:\Enrolment\jdoe.cer

REM Inventory of all associations for an audit
EIDCmd /LISTALL > %COMPUTERNAME%-associations.txt

REM Which account does this certificate open?
EIDCmd /GETUSERNAME /CERTIFICATE C:\Enrolment\jdoe.cer
```

9.2 Configuration wizard switches

Switch	Effect
`/USER username`	Open the wizard for the given account.
`/ADVANCED`	Show advanced options, such as requiring the smart card for a single account.
`/ENABLESIGNATUREONLY` `/DISABLESIGNATUREONLY`	Allow or refuse signature-only keys for logon.
`/ENABLENOEKU` `/DISABLENOEKU`	Allow or refuse certificates without the Smart Card Logon EKU.
`/ENABLETIMEINVALID` `/DISABLETIMEINVALID`	Allow or refuse expired or not yet valid certificates.
`/LANG`	Use the language of the Regional format instead of the display language.

9.3 Kiosk modes

9.3.1 Generic kiosk

An account can accept **any trusted certificate** that is not associated with another account - useful for a shared kiosk. Run `EIDCmd /KIOSK /USER <account>` and give the account password when asked. The wizard then shows a special kiosk entry for that account; dissociate it to leave kiosk mode.

9.3.2 PIV kiosk (US DoD)

With the policy `PIVKiosk` = 1 (DWORD) under `HKLM\SOFTWARE\Policies\Microsoft\Windows\SmartCardCredentialProvider`, a local account is created automatically at the first logon of any card meeting all of the following:

- the certificate has the Smart Card Logon EKU and is trusted by the **computer** stores;
- the subject name ends with a dot followed by a number (the PIV/EDIPI identifier);
- the UPN of the certificate is that number followed by `@mil`.

Each creation is recorded as event 1004. `EIDCmd /CREATESECUREACCOUNT` creates such an account in advance.

10 Auditing

EIDAuthenticate 2.0 writes a structured audit trail designed for SIEM rules: every logon attempt with the card and certificate that performed it, the reason of every refusal, and every administrative change.

10.1 Where to find the events

Open Event Viewer (`eventvwr.msc`) as an administrator: **Applications and Services Logs > EIDAuthenticate > Operational**. The channel name for scripts and event forwarding is `MySmartLogon-EIDAuthenticate/Operational`.

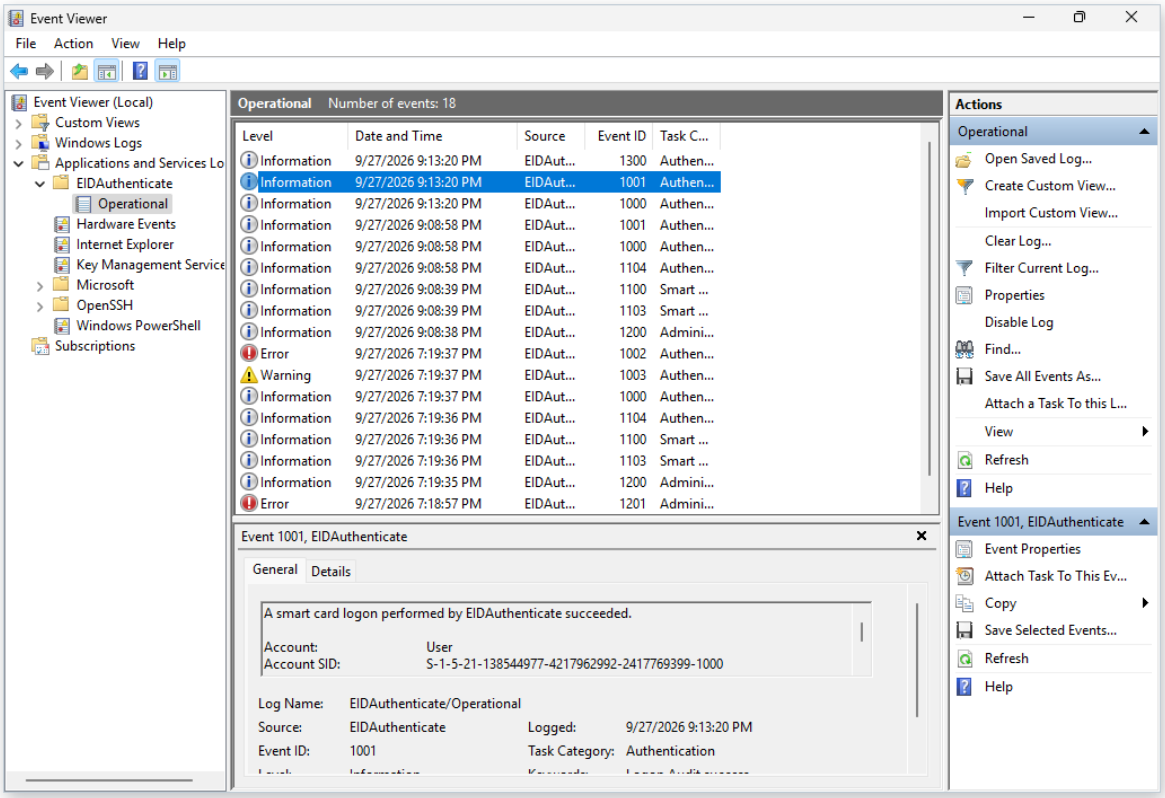


Figure 33 – The EIDAuthenticate audit channel in Event Viewer

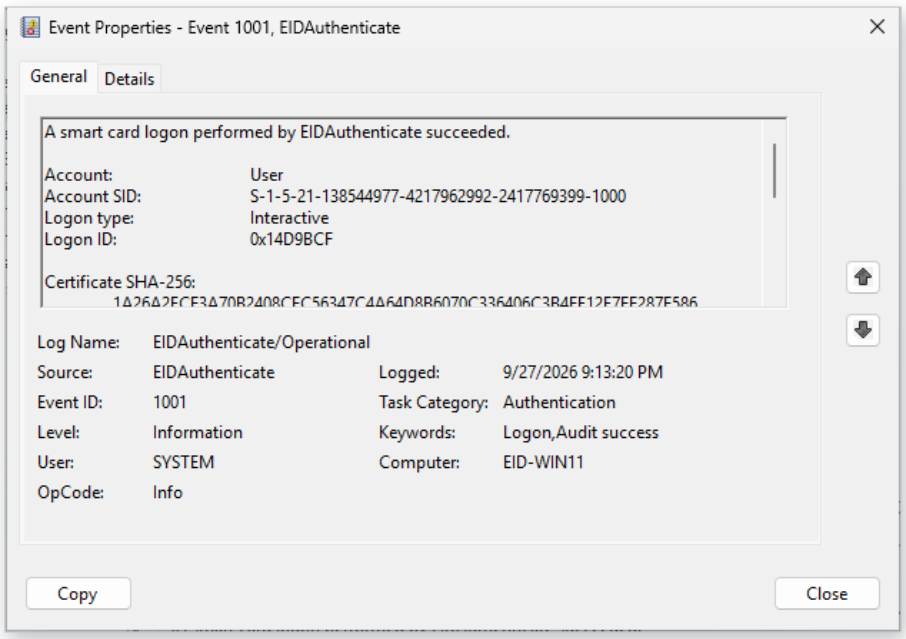


Figure 34 – Event 1001 - account, logon type, LogonId and certificate of a successful logon

QUERYING THE CHANNEL

```
wevtutil qe MySmartLogon-EIDAuthenticate/Operational /f:text /c:20 /rd:true

Get-WinEvent -LogName 'MySmartLogon-EIDAuthenticate/Operational' -MaxEvents 50 |
    Where-Object Id -in 1002,1003,1007,1201,1301 | Format-List TimeCreated, Id, Message
```

10.2 Event reference

ID	Level	Event	Written by
1000	Information	Logon attempt (reader, card, container)	Package (LSASS)
1001	Information	Logon succeeded - account, SID, LogonId, certificate thumbprints, subject, issuer, serial, effective policy	Package
1002	Error	Logon failed - same fields plus FailureReason, Status and SubStatus	Package
1003	Warning	Wrong PIN - remaining attempts (-1 unknown, 0 blocked)	Package
1004	Information	Account created by PIV kiosk mode	Package
1005 / 1006 / 1007	Info / Info / Error	Network logon attempt / succeeded / failed (WinRM, SMB)	Package, server side
1100 / 1101	Information	Card inserted (number of logon tiles) / removed	Credential provider
1102	Warning	Certificate not offered for logon, with the reason	Credential provider
1103 / 1104 / 1105	Information	Logon tile offered / submitted / result reported	Credential provider
1200 / 1201	Info / Error	Configuration change applied / refused (association, kiosk, flags...)	Package, for the caller
1300 / 1301	Info / Error	Secondary logon launched / refused (EIDRunAs)	EIDSecondaryLogon
1900	Information	Package initialized - version actually loaded by LSASS	Package

10.2.1 Logon failure reasons (1002, 1007)

Value	Meaning	Value	Meaning
1	License not valid	10	Associated account cannot be resolved
2	Malformed logon buffer	11	Wrong PIN
3	PIN could not be unprotected	12	Stored credential cannot be decrypted with this card
4	Certificate cannot be read	13	Logon session cannot be created
5	No account associated with the certificate	14	Profile cannot be created
6	Certificate not trusted	15	Computer name cannot be determined
7	Certificate revoked	16	Internal error
8	Certificate expired or not yet valid	17	Signed network response does not verify (1007 only)
9	Revocation status unknown		

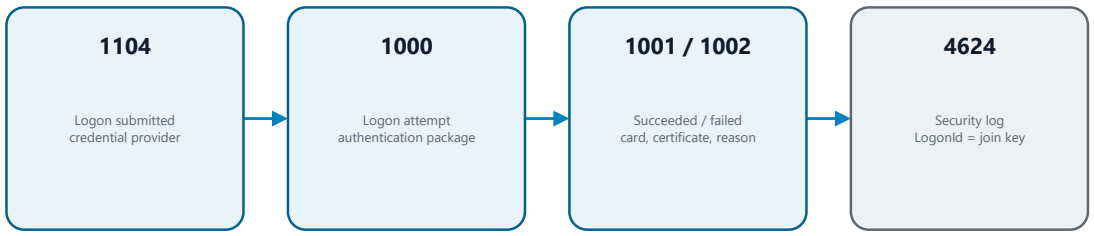


Figure 35 – Correlating the audit channel with the Windows Security log

10.3 Privacy and integrity

- No PIN, password or key material is ever logged. Certificates are identified by public data: SHA-1 and SHA-256 thumbprints, subject, issuer and serial number.
- Only SYSTEM can write to the channel; administrators can read and clear it; members of **Event Log Readers** can read it, so forwarding agents do not need administrator rights.
- The log is 32 MB and archives itself when full instead of overwriting records.
- An administrator can always tamper with a local log. When non-repudiation matters, forward the channel with **Windows Event Forwarding** to a collector and treat the collector as the reference copy.

> **TIP**
> Rules worth creating: 1003 repeated for one account (PIN brute force), 1201 with STATUS_ACCESS_DENIED (attempt to reconfigure an account), 1301 with refusal 10, 11 or 13 (reuse of an EIDRunAs authorisation), and any 1200 or 1004 outside a planned enrolment.

11 Troubleshooting

11.1 Common problems

Symptom	Cause and solution
The card is inserted but no smart card sign-in appears	Check the card with `certutil -scinfo` (next section). Event 1100 gives the number of tiles produced and each 1102 the reason a certificate was rejected (no EKU, not associated, untrusted, expired...).
"No valid certificates found" when the computer is locked	The card belongs to another account: click **Switch user**.
The logon fails with a revocation error	The revocation policy is mandatory and the CRL/OCSP server cannot be reached - see section 5.3, or check the SYSTEM proxy.
"The EID Security package is not loaded"	LSA did not load the package - usually LSA protection or an interrupted installation. Repair the installation and look for event 1900.
WinRM returns an access denied error	The certificate is not associated with an account on the server, the client is not in TrustedHosts, or the account is a local administrator filtered by remote UAC (LocalAccountTokenFilterPolicy). Event 1007 on the server gives the reason.
EIDRunAs: "The secondary logon service refused this command"	A hardening check failed; the message names the check and the command that relaxes it. See section 8.3.
The wizard test fails with "wrong PIN" although the PIN is right	Another credential provider or a card middleware may be answering the prompt. Run the test again and look at the 1104/1105/1000 sequence in the audit channel: a 1104 and 1105 without 1000 means the request never reached EIDAuthenticate.

11.2 Checking the card with certutil

`certutil -scinfo` (built into Windows) shows what Windows sees in the reader. Run it from a command prompt; add `-silent` to avoid the PIN prompts.

```
The Microsoft Smart Card Resource Manager is running.
Current reader/card status:
Readers: 1
  0: MySmartLogon Virtual Reader 0
--- Reader: MySmartLogon Virtual Reader 0
--- Status: SCARD_STATE_PRESENT | SCARD_STATE_UNPOWERED
--- Status: The card is available for use.
---   Card: Identity Device (Microsoft Generic Profile)
---    ATR:
       3b 8c 01 4d 79 53 6d 61  72 74 4c 6f 67 6f 6e a5   ;..MySmartLogon.


================================================
Analyzing card in reader: MySmartLogon Virtual Reader 0

------------================================-------------
================ Certificate 0 ================
--- Reader: MySmartLogon Virtual Reader 0
---   Card: Identity Device (Microsoft Generic Profile)
Provider = Microsoft Base Smart Card Crypto Provider
Key Container = 870b91be-9f1b-4545-a483-a2bc4eebb5e1 [Default Container]

No AT_SIGNATURE key for reader: MySmartLogon Virtual Reader 0
Serial Number: 29da638e1c5b4d6d
Issuer: CN=EID-WIN11
 NotBefore: 9/27/2026 7:17 PM
 NotAfter: 9/27/2036 7:17 PM
Subject: CN=User
Non-root Certificate
Cert Hash(sha1): 2723b3297b4fe66db28564588405d0f135dff3bf

Performing AT_KEYEXCHANGE public key matching test...
Public key matching test succeeded
  Key Container = 870b91be-9f1b-4545-a483-a2bc4eebb5e1
  Provider = Microsoft Base Smart Card Crypto Provider
  ProviderType = 1
  Flags = 1
    0x1 (1)
  KeySpec = 1 -- AT_KEYEXCHANGE
```

Figure 36 – certutil -scinfo on a healthy card

What you see	Meaning	What to do
SCARD_STATE_PRESENT, a Card name, the certificate and a successful key matching test	The card works.	-
SCARD_STATE_EMPTY	No card detected in the reader.	Reseat the card, try another reader.
An ATR but an empty Card line, "Cannot retrieve Provider Name"	No minidriver or CSP is installed for this card (or only a 32-bit one on a 64-bit system).	Install the card middleware or let Windows Update find the minidriver.
"The Smart Card Resource Manager is not running"	The Smart Card service is stopped or disabled.	Set the **Smart Card** service to Automatic and start it.

11.3 Collecting traces with EID Log Manager

EID Log Manager (`EIDLogManager.exe`) captures the debug trace of every EIDAuthenticate component. It is the first thing support will ask for.

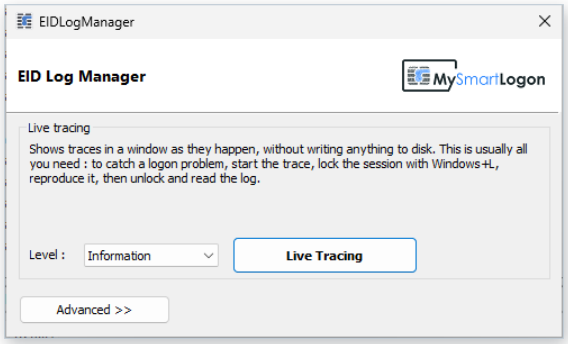


Figure 37 – Live tracing

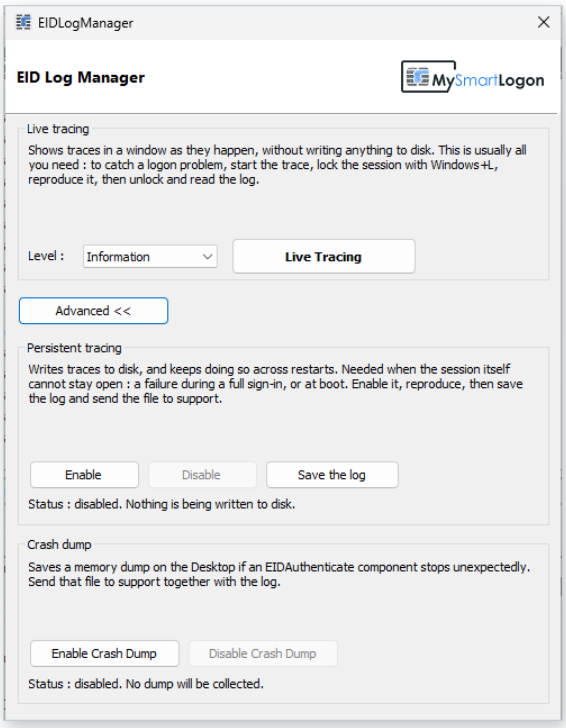


Figure 38 – Advanced: persistent tracing and crash dumps

11.3.1 Live tracing

Choose the level and click **Live Tracing**: traces are displayed as they happen, without writing anything to disk. To capture a logon problem, start the trace, lock the session with Windows+L, reproduce the problem, unlock and read or **Save** the log. Errors are highlighted in red and counted in the status bar.

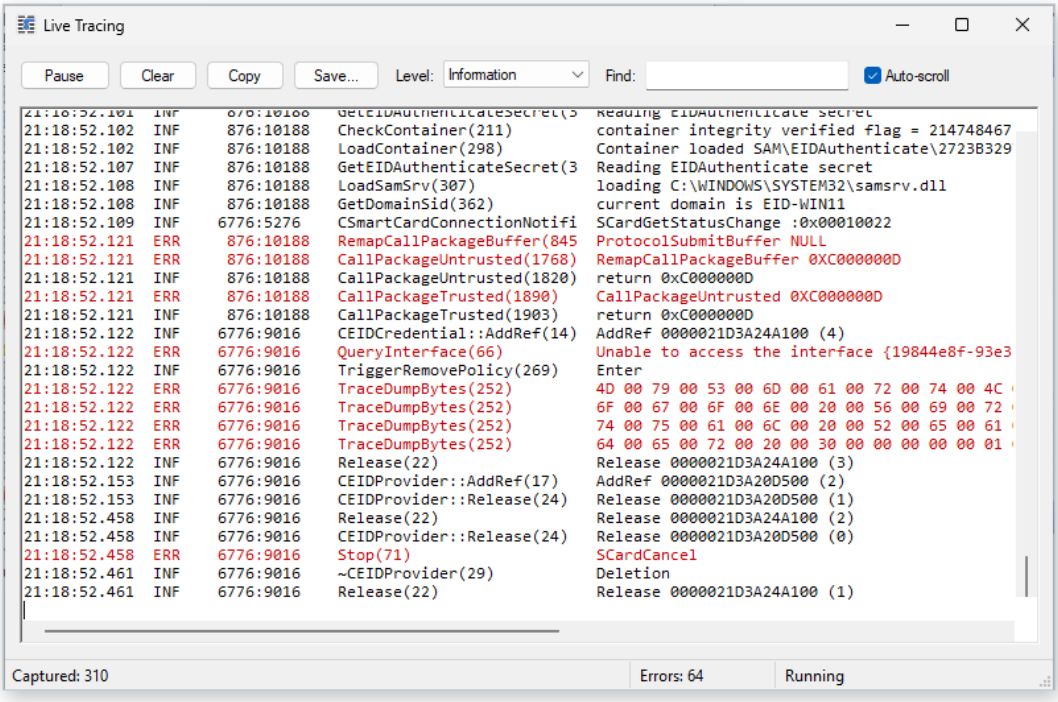


Figure 39 – Live tracing window during a logon

11.3.2 Persistent tracing

When the problem prevents a session from staying open (a failure during a full sign-in, or at boot), use **Advanced >> Persistent tracing**: **Enable** writes traces to disk, also across restarts; reproduce the problem, then **Save the log** and **Disable**.

11.3.3 Crash dumps

Enable Crash Dump saves a memory dump on the desktop if an EIDAuthenticate component stops unexpectedly. Most crashes inside LSASS are caused by faulty card drivers. Send the dump together with the log.

> **WARNING**
> Traces never contain the PIN or the password, but they contain account names and card information. A memory dump of LSASS can contain secrets: transmit it only through a secure channel.

11.4 Capturing traces manually

On computers where the tool cannot be used, the same ETW provider can be recorded with the Windows tools:

ELEVATED COMMAND PROMPT

```
logman create trace EIDAuthenticateTrace -p {4AE3C5F9-BB41-41A5-A82B-80EEE8C38C52} -o trace.etl
logman start EIDAuthenticateTrace
REM ... reproduce the problem ...
logman stop EIDAuthenticateTrace
logman delete EIDAuthenticateTrace
```

Send the resulting .etl file to support.

11.5 Troubleshooting the installation

VERBOSE INSTALLATION LOG

```
msiexec /i EIDAuthenticateEnterpriseEdition_x64.msi /l*v install.log
```

The custom actions of the product (license check, package registration, audit channel, services) write their diagnostics in this log.

12 Appendix - registry reference

Policies read from `HKLM\SOFTWARE\Policies\Microsoft\Windows\SmartCardCredentialProvider` (DWORD unless stated). They can be deployed with Group Policy Preferences.

Value	Default	Meaning
`AllowSignatureOnlyKeys`	0	Allow certificates whose key can only sign.
`AllowCertificatesWithNoEKU`	0	Allow certificates without the Smart Card Logon EKU.
`AllowTimeInvalidCertificates`	0	Allow expired or not yet valid certificates.
`ForceCRLCheck`	0	0 no check, 1 check without blocking, 2 mandatory check.
`UrlRetrievalTimeout`	30000	Revocation download timeout, in milliseconds.
`RevocationFreshnessTime`	86400	Maximum age of cached revocation data, in seconds.
`AdditionalCRLPath`	(none)	REG_SZ - folder of .crl files (DER) used for offline revocation checks.
`AllowGetTheRemainingPinAttemptUsingMinidriver`	0	Display the remaining PIN attempts when the minidriver reports them.
`PIVKiosk`	0	Create accounts automatically for PIV cards (section 9.3).

Windows policies used by the product, under `HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\System`:

Value	Meaning
`scforceoption`	1 = Interactive logon: Require smart card.
`scremoveoption`	REG_SZ - 0 No action, 1 Lock, 2 Log off, 3 Disconnect remote session.
`scdisablerecovery`	1 = hide the emergency link of the force smart card policy.

EIDRunAs hardening: `HKLM\Software\MySmartLogon\EIDAuthenticate\SecondaryLogon` (section 8.3).