
privacyIDEA Credential Provider Documentation

Release 2.2

NetKnights GmbH

May 26, 2021

Contents

1	Introduction	3
2	Installation	5
2.1	Prerequisites	5
2.2	MSI package	5
2.3	Manual Installation	6
3	Configuration	9
3.1	Registry Settings	9
4	Development, Maintenance and Support	13
5	Indices and tables	15

Contents:

CHAPTER 1

Introduction

The privacyIDEA Credential Provider is a tool to improve the logon security of your Windows Desktops, Servers and Windows Terminal Servers. It is used to add a second factor for authentication, when logging into your Windows system.

The privacyIDEA Credential Provider does this by communicating with the privacyIDEA Authentication System¹. The privacyIDEA Authentication System can manage many different kind of second factors for the domain users. Ranging from classical OTP tokens, one time codes via SMS, Smartphone Apps to the Yubikey.

Users need to authenticate with their windows password and additionally with their token as second factor.

¹ <https://privacyidea.org>

2.1 Prerequisites

To use the privacyIDEA Credential Provider you need to have a privacyIDEA Authentication System. The installation and setup of this backend is covered in another documentation¹.

Ask the company NetKnights to get an evaluation version of the privacyIDEA Credential Provider².

2.2 MSI package

The privacyIDEA Credential Provider comes as a 32bit and 64bit MSI package. You can install it manually or use your software distribution tool.

2.2.1 Start installation

In the first step you can decide, if you want to make the privacyIDEA Credential Provider the default provider. This means, that no other credential provider is active on this machine. The user can not login with only his Windows password anymore.

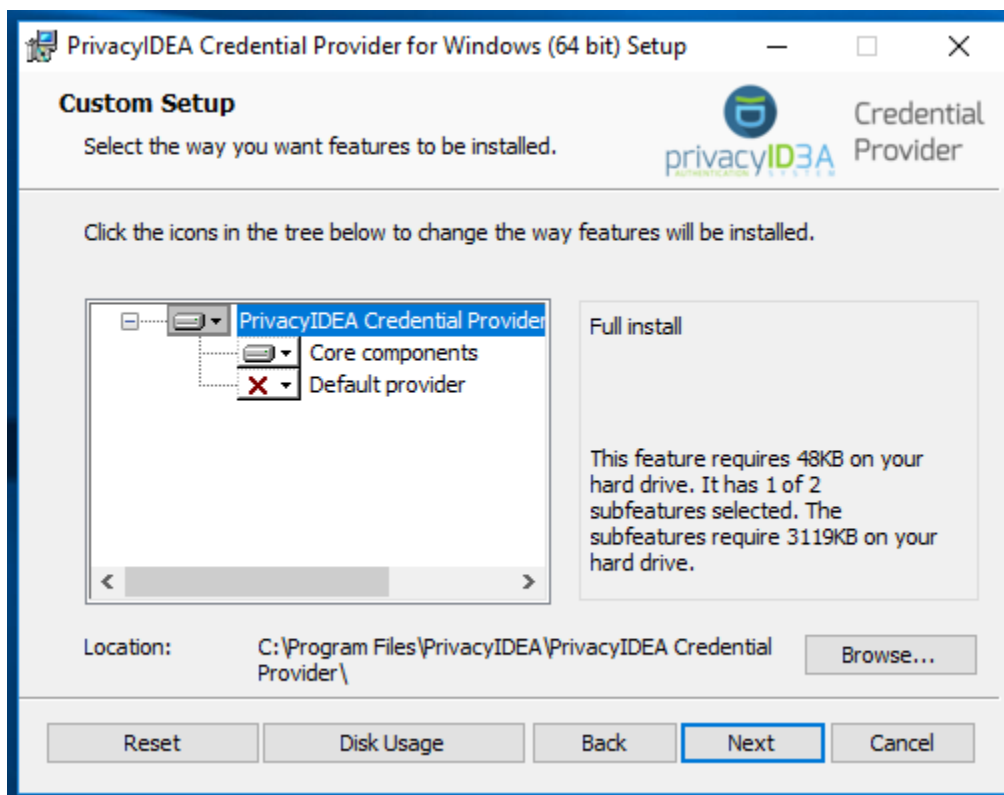
Note: We recommend not activating this setting during installation. First you should configure the privacyIDEA Credential Provider and check, if it works right. After this, you can change the installation and change this configuration.

2.2.2 Configure the privacyIDEA Authentication Server

In the next step, you can configure the communication to the privacyIDEA Authentication Server. The credential provider and the server communicate via the REST API `POST /validate/check`.

¹ <http://privacyidea.readthedocs.io/en/latest/installation/index.html>

² <https://netknights.it/en/unternehmen/kontakt/>



Note: You only need to specify the hostname of the authentication server. In most cases you only need to enter the hostname like *yourserver.example.com*. Additionally the path can be specified if there is. Something like */path/to/pi*.

You can specify a custom **login text**, which will be displayed underneath the provider.

You can also specify if certain SSL errors shall be ignored.

Warning: We recommend NOT to ignore any SSL errors in productive use. Otherwise you will be vulnerable to man-in-the-middle attacks. An attacker who intercepts the communication could modify the authentication response and thus make the second factor useless.

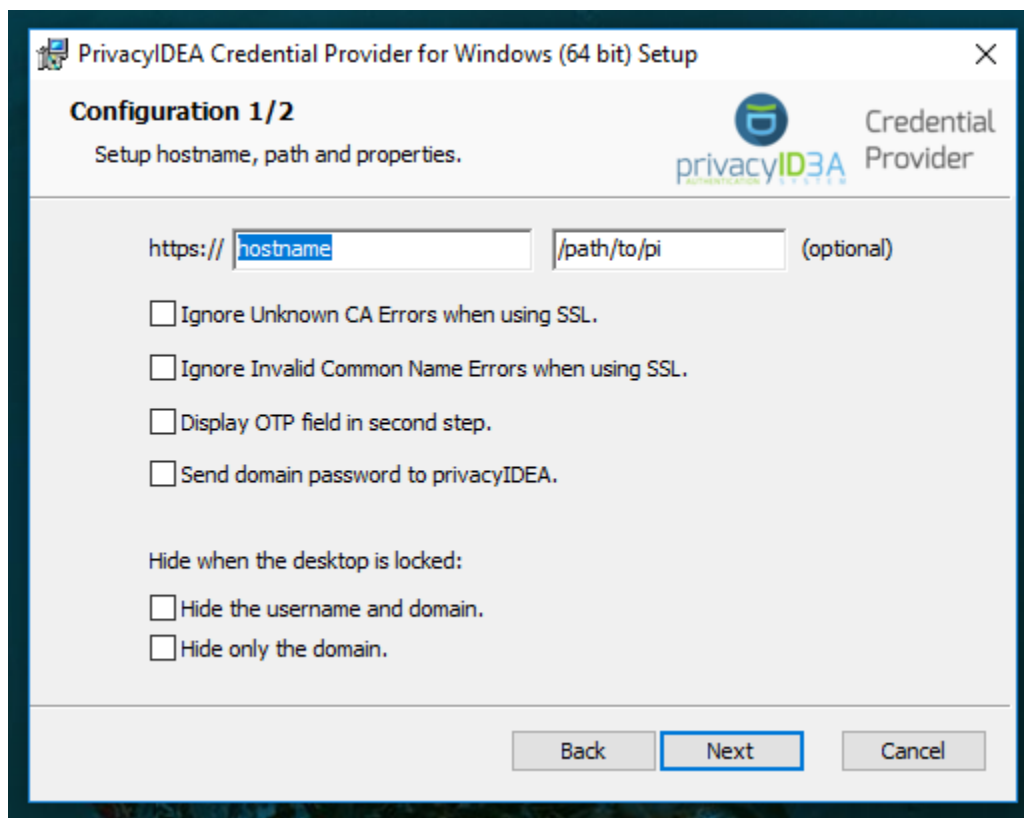
You may specify the path to a custom login image.

Note: The image must be a BMP version 3 file.

After these two steps the privacyIDEA Credential Provider is installed on your system and can be chosen for login.

2.3 Manual Installation

The privacyIDEA Credential Provider and Filter can also be registered manually. To do this, the file `PrivacyIDEACredentialProvider.dll` has to be put into `%windir%\System32`. (If desired, the `PrivacyIDEACredentialProviderFilter.dll` can be added aswell).



Next, the privacyIDEA Credential Provider has to be registered to be loaded into the logon process. This is done by adding its CLSID to the list of Credential Providers at `HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\Credential Providers\`. Add a new key here with the name `{7BAF541E-F8E0-4EDF-B69A-BD2771139E8E}` (the CLSID). Afterwards set the data of the default to `PrivacyIDEACredentialProvider`. Finally, the DLL has to be registered with the system. To do this, go to `HKEY_CLASSES_ROOT\CLSID\` and add a new key with the CLSID from above. Add another key to the on just created with the name `InprocServer32`. Set the default data to `PrivacyIDEACredentialProvider.dll` and add another `REG_SZ` with the name `ThreadingModel` and data `Apartment`. Now the privacyIDEA Credential Provider is registered and should be visible at the next Login attempt. This can also be done via the file `RegisterProvider.reg`.

If you wish to also use the privacyIDEA Credential Provider Filter, do the steps above again with the CLSID of the Filter which is `{34065473-D75F-4BC2-9782-E98E63ED0D41}` and registration at `HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\Credential Provider Filters\`. Alternatively, the file `RegisterFilter.reg` can be used.

To unregister, the corresponding files `UnregisterXXX.reg` can be used. This does not remove the configuration, DLL files or CLSID entries, it only removes the Provider or Filter from the Authentication flow at `HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\`.

During installation of the privacyIDEA Credential Provider you already configured all necessary settings, but it can be interesting to change settings later. Like changing the available credential providers or changing the verification of the authentication server certificate.

3.1 Registry Settings

If you want to change the configuration after the installation, you can only do this by editing the registry keys. You can use administrative templates to deploy the credential provider on many desktops in your network.

The configuration is located at `Computer\HKEY_LOCAL_MACHINE\SOFTWARE\NetKnights GmbH\PrivacyIDEA-CP\`.

NOTE: Not all registry entries listed below will be generated from installing the credential provider. Those have to be added manually.

3.1.1 Connection Settings

These settings define the connection to the privacyIDEA server. The connection is established via https by default, like indicated in the installer.

hostname

The hostname of the privacyIDEA Authentication Service. That usually is something like *yourserver.example.com* without any additional path information.

path

The path to the privacyIDEA Authentication Service if there is. E.g. */test/path/pi*

NOTE: The entry */path/to/pi* is a placeholder. If it is read by the Credential Provider, it is treated as an empty entry.

ssl_ignore_invalid_cn

Set to 1 if the privacyIDEA Credential Provider should ignore SSL errors originating from an invalid common name.

ssl_ignore_unknown_ca

Set to 1 if the privacyIDEA Credential Provider should ignore SSL errors originating from an unknown CA.

custom_port

This entry is not there by default. You can add it to declare a custom port. The value has to be of type *REG_SZ* with the name *custom_port*.

NOTE: By default the port is the default https port, which is 443.

resolve_timeout, connect_timeout, send_timeout, receive_timeout

With these entries you can specify the timeout (in ms) for the corresponding phase. This might be interesting if the offline feature is used. The default timeouts are infinite / 60s / 30s / 30s.

3.1.2 Login behaviour

Using these settings you can specify the behaviour of the privacyIDEA Credential Provider. The credential provider can ask for the username, the password and the otp value in one step or in two steps.

two_step_hide_otp

Set to 1 if the privacyIDEA Credential Provider should ask for the user's OTP in a second step. In the first step the user will only be asked for the password.

two_step_send_password

Set to 1 if the privacyIDEA Credential Provider should send the user's password to the privacyIDEA Authentication Service.

two_step_send_empty_password

Set to 1 if the privacyIDEA Credential Provider should send an empty password to the privacyIDEA Authentication Service.

NOTE: If both **two_step_send_password** and **two_step_send_empty_password** are set to 1, the privacyIDEA Credential Provider will send an empty password to the privacyIDEA Authentication Service. NOTE: Sending the windows or an empty password can be used to trigger token types like SMS or Email.

excluded_account

Specify an account that should be excluded from 2FA. The format is required to be `domain\username` or `computername\username`.

3.1.3 Disabling for specific scenarios

There different *credential provider usage scenarios* ("cpus"). The available scenarios are **logon**, **unlock** and **credui**.

It is possible to configure both the Credential Provider and the Filter¹ for each of the scenarios. This way the administrator can define a different behaviour if a users either logs in or unlocks his desktop.

The behaviour in each scenario can be configured via the corresponding registry entry named **cpus_logon**, **cpus_unlock** and **cpus_credui**.

These entries expect a *REG_SZ*, that consist of a digit 0, 1, 2, 3 and a character "e" or "d".

- 0: relevant for *remote* (RDP) and *local* operation
- 1: relevant for *remote* operation

¹ The Filter is the component that defines, if only the privacyIDEA Credential Provider is be available for login. If the Filter is not installed, then the privacyIDEA Credential Provider and all other credential providers are available.

- 2: relevant for *local* operation
- 3: relevant for *remote* and *local* operation - but privacyIDEA Credential Provider completely disabled.

The characters stand for:

- “e”: Only the privacyIDEA Credential Provider is available. All other credential providers are not available.
- “d”: In addition all other credential providers are available.

E.g. This would result in:

- `cpus_logon = 0e`: Only the privacyIDEA Credential Provider is available for Logon via remote and locally. (0d would be the contrary.)
- `cpus_unlock = 1d`: Remotely the locked desktop can be unlocked with all available credential providers. (1e would be the contrary.)
- `cpus_unlock = 2e`: Locally unlocking the desktop is only possible with the privacyIDEA Credential Provider. (2d would be the contrary.)
- `cred_ui = 3d`: For credui scenarios the privacyIDEA Credential Provider is completely disabled, no matter if remotely or locally. Only the other credential providers are available. (Note: “3e” does not exist)

If there is no entry for a scenario, the default is assumed: The privacyIDEA Credential Provider will be available and the Filter will be active, if installed.

3.1.4 Recommended setup for remote desktop scenarios

In scenarios where the privacyIDEA Credential Provider shall be used for RDP connections, it is recommended to install the privacyIDEA Credential Provider only on the RDP target together with the Filter. It is also recommended to use the `two_step_hide otp` setting to skip entering the windows password a second time.

3.1.5 Customization of the Look and Feel

You can also change the look and feel of the privacyIDEA Credential Provider.

login_text

Specify the text that is displayed underneath the credential logo and on the right side where available credentials are listed. The default is “privacyIDEA Login”.

otp_text

Specify the text that is displayed in the OTP input field. Usually this is “One-Time Password”, but you can change it to any other value you like.

otp_hint_text

Specify the text that is displayed when prompted to enter the OTP in the second step. The default is “Please enter your second factor!”.

otp_fail_text

Specify a custom text that is shown when the OTP verification failed. The default is “Wrong One-Time Password!”. NOTE: An error on either the client or server side overwrites this message.

hide_domainname

Set to 1 if you want the privacyIDEA Credential Provider to hide only the domain name when the desktop is locked.

hide_fullname

Set to 1 if you want the privacyIDEA Credential Provider to hide the user and domain name when the desktop is locked. Instead only the contents of the *login_text* settings will be displayed.

v1_bitmap_path

The complete path and filename of a bitmap image. This is a customized login image. The image must be a version 3 Windows BMP file with a resolution of 128x128 pixels.

no_default

Add this registry entry and set it 1 to not have the privacyIDEA Credential Provider selected by default when logging in.

show_domain_hint

Set this to 1 to show the Domain that is currently used to log in.

prefill_username

Set this to 1 to have the username field prefilled with the user that last logged on.

offline_file

Specify the **absolute** path to where the offline file should be saved. The default is C:offlineFile.json. NOTE: Either txt or json file type is recommended.

offline_try_window

Specify how many offline values shall be compared to the input at max. Default is 10. A value of 0 equals the default.

3.1.6 Realms

Realms are implemented by mapping Windows domains to privacyIDEA realms. When a matching mapping exists, the `&realm=...` parameter is added to the request.

default_realm

Specify a default realm. If set, it is appended to every request that has no other matching mapping.

The mapping is done in the sub key `realm-mapping` (\Rightarrow `HKEY_LOCAL_MACHINE\SOFTWARE\Netknights GmbH\PrivacyIDEA-CP\realm-mapping`). Here you can specify the Windows domains as the names and the privacyIDEA realms as data of `REG_SZ` entries.

3.1.7 Log file

release_log

Set to 1 if you want the privacyIDEA Credential Provider to write a logfile in the release version. The log only contains errors and is located at `C:\privacyIDEAReleaseLogFile.txt`.

The log file of the debug version contains more detailed information and is located at `C:\privacyIDEADebugLogFile.txt`

log_sensitive

In some cases it can be useful to log sensitive data (e.g. passwords) to find the cause of a problem. By default sensitive data is not logged. To log sensitive data aswell, create a new registry key of type `REG_SZ` with the name *log_sensitive* and a value of *1*. This can be deleted after creating a logfile. NOTE: This only affects the *debug* versions of the privacyIDEA Credential Provider.

Development, Maintenance and Support

The privacyIDEA Credential Provider was first developed by Last Squirrel IT¹. The company has a long experience in Microsoft Windows security tools. They developed many different credential providers and plugins for Active Directory Federation Services.

Since 2018 the development is continued by NetKnights. You will get maintenance and support via the company NetKnights². NetKnights also maintains the privacyIDEA Authentication System and issues different service level agreements³ for the privacyIDEA Credential Provider and the privacyIDEA Authentication System.

¹ <http://www.lastsquirrel.com>

² <https://netknights.it>

³ <https://netknights.it/en/produkte/privacyidea-credential-provider/>

CHAPTER 5

Indices and tables

- `genindex`
- `modindex`
- `search`