Authentication Agent Manual

Publication date February 29, 2024

Abstract
This document describes how to install and configure the Authentication Agent.

Table of Contents

List of Procedures

Chapter 1. Introduction

Developed by BalaSys, Authentication Agent (AA) is an authentication client, capable of cooperating with the PNS firewall and the Authentication Server (AS) to identify the users initiating network connections. Authentication Agent enables the complete network traffic to be audited on the user level.

Chapter 2. Authentication and PNS

Authentication Agent (AA) is an authentication client, capable of cooperating with the PNS firewall and the Authentication Server (AS) to identify the users initiating network connections. The authentication process and the related communication between the components is summarized below. For more details, see *Chapter 15, Connection authentication and authorization* in *Proxedo Network Security Suite 2 Administrator Guide*.

The authentication aims to determine the identity of the user. During the authentication process the user initiating the connection shares a piece of sensitive information (for example, a password) with the other party that verifies the user's authenticity.

Several procedures (so called authentication methods) exist for verifying the identity of the user:

1. The user owns some pieces of sensitive information, for example, a password, PIN code, the response to a challenge, and so on.
2. The user owns a device, for example, a hardware key, chipcard, SecurID token, and so on.

Naturally, the above methods can be combined to implement strong two-factor level authentication in sensitive environments.

2.1. Authentication on the network

The aim of network authentication is to authenticate the connections initiated by the users in order to ensure that only the proper users can access the services. Basically there are two types of authentication:

1. *Inband*: Authentication is performed by the application-level protocol — the data traffic required for the authentication is part of the protocol. Inband authentication is used for example in the HTTP, FTP, or SSH protocols. The protocols usually support different authentication methods — these are usually described in the specifications of the protocol.
2. *Outband*: Authentication is performed in a separate data channel completely independent from the protocol of the accessed service. Outband authentication is realized by the combination of the Authentication Agent (AA), Authentication Server (AS), and PNS softwares. The advantage of outband authentication is that it can be used to authenticate any protocol, regardless of the authentication methods supported by the original protocol. That way, strong authentication methods (for example, chipcards) can be used to authenticate protocols supporting only the weak username/password method (for example, HTTP).

2.2. Procedure – Outband authentication with PNS

Purpose:

PNS implements outband authentication according to the following procedure:

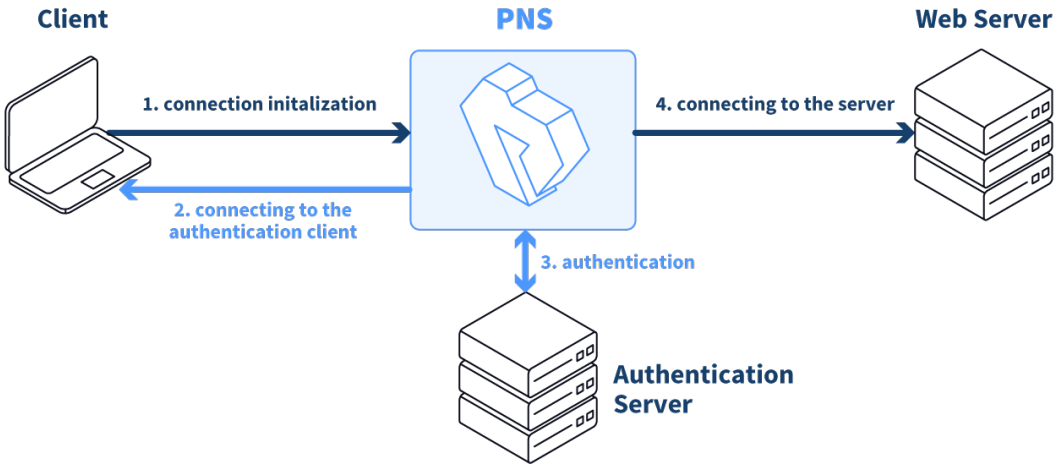

Figure 2.1. Outband authentication with PNS

Steps:

Step 1. The client initiates a connection towards the server.

Step 2. PNS determines the service to be accessed based on the IP address of the client and the server. If authentication is required for the connection (an authentication policy is assigned to the service), PNS initiates a connection towards the client using the Authentication Agent protocol.

Step 3. Depending on the authentication methods available (for example, for password-based authentication), the dialog of the Authentication Agent is displayed on the client machine. The user enters the username that the Authentication Agent forwards to PNS.

Step 4. The PNS firewall connects to Authentication Server (AS) and retrieves the list of authentication methods enabled for the particular user. Multiple authentication methods can be enabled for a single user (for example, x509, Kerberos, password, and so on). The authorization of the user is also performed in this step, for example, the verification of the LDAP group membership.

Step 5. PNS returns the list of available methods to the client. The user selects a method and provides the information (for example, the password) required for the method.

Step 6. The Authentication Agent sends the data (for example, the password) to PNS that forwards it to AS.

Step 7. AS performs the authentication and notifies PNS about the result (success/failure).

Step 8. PNS returns the result to the client and — if the authentication was successful, builds a connection towards the server. In case of a failed authentication it terminates the connection to the client.

Chapter 3. Installing the Authentication Agent (AA)

This section describes the installation and configuration of the Authentication Agent on Microsoft Windows and GNU/Linux platforms. The Authentication Agent has to be installed on every computer having access to authenticated services.

The agent has two components:

1. *Authentication Agent Multiplexer*: It is a daemon running in the background, accepting the connections coming from PNS and verifying the TLS certificates of PNS (if the communication is encrypted). In a multi-user environment the Multiplexer displays the dialog of the *Authentication Agent* on the desktop of the user initiating a connection requiring authentication.
2. *Authentication Agent*: This application collects the information required for the authentication, for example, the username, authentication method, password, and so on.

The following platforms are supported:

- Windows 10 LTSB (Long-Term Servicing Branch)
- Windows Server 2016, 2019
- Ubuntu 22.04 LTS

AA is distributed as a portable AppImage package on GNU/Linux platforms without needing superuser permissions to install the application.

3.1. Installing the Authentication Agent on Microsoft Windows platforms

3.1.1. Procedure – Installing the Authentication Agent on Microsoft Windows

Purpose:

The Authentication Agent (AA) installer is located in the `\platforms\windows\` folder of the PNS CD-ROM, its latest version is also available from the *BalaSys website*.

The installer is available as Windows Installer Package (.msi)

Steps:

Step 1. Place the PNS CD-ROM into the CD drive and start the `authentication-agent-<version>.msi` file located in the `\platforms\windows\` folder.

Warning
Administrator privileges are required to install the application.

Step 2. Check **I accept the terms in the License Agreement** to accept the End-User License Agreement, which is displayed after the installer starts. Click **Next** to continue installation process. To cancel the installation at any time during the process, click **Cancel**.

Figure 3.1. Accepting the EULA

Step 3. Select the destination folder for the application and click **Next** to continue. The default folder is `C:\Program Files\auth-agent`.

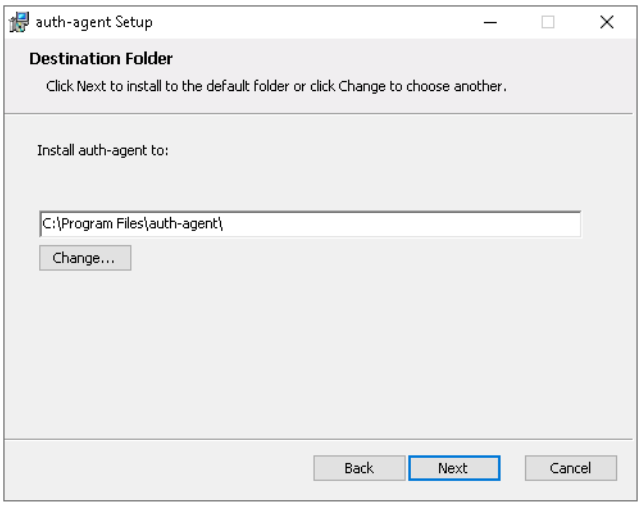

Figure 3.2. Selecting the destination folder

Step 4. *Optional step*: Click **...** button, select the CA certificate to import, then click **Open** to import the CA certificate.

Note
For authentication purposes, when PNS communicates with AA, AA expects TLS-encrypted communication. For details, see section *Section 4.1.1, Registry entries on Microsoft Windows platforms (p. 10)* and section *Section 4.1, Configuring Authentication Agent on Microsoft Windows platforms (p. 10)*.

If the Authentication Agent and PNS communicate through a TLS-encrypted channel (recommended), the certificate of the Certificate Authority (CA) signing the certificates of the PNS firewalls can be imported to the Authentication Agent Multiplexer's certificate store.

Note
The CA certificate has to be in `DER` or `PEM` format. (with typical file extensions of *.der, *.pem, *.crt, *.cert) It is not necessary to import the certificate during the installation, it can also be done later. For details about encrypting the agent-PNS authentication, see *Section 4.1.3, Configuring TLS connections on Microsoft Windows platforms (p. 12)*.

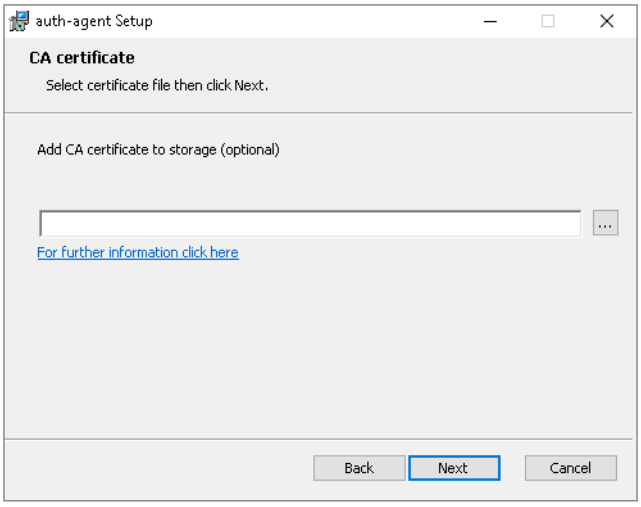

Figure 3.3. Importing the CA certificate

Step 5. Click **Install** to start the installation process. The installer copies the required files and registers the service called **Authentication Agent Multiplexer**, which is started after the registration.

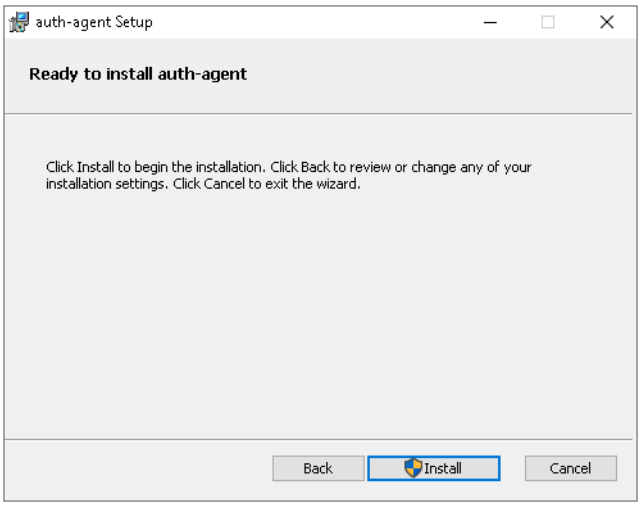

Figure 3.4. Ready to start installation

Step 6. After the installer has completed the above steps, click **Finish**.

Step 7. The Authentication Agent (AA) logo is displayed on the system tray, indicating that the application is running. It is also started automatically after each Windows startup.

3.1.2. Procedure – Installing Authentication Agent with Group Policy Object (GPO) deployment

Prerequisites:

- Create the necessary certificates as instructed in section *Procedure 11.3.8.2, Creating certificates* in *Proxedo Network Security Suite 2 Administrator Guide*.
- Set the parameters for the AS certificate.
- Export the CA certificate signed by AS in `DER` format for the Windows client.

Steps:

Step 1. Download the .msi installer. The browser application or the Windows Defender Cloud might send a notification or a warning due to the new and unknown installer program, this can be disregarded.

Step 2. Install the Windows Client and import the CA certificate during the installation. Reboot the system, if it is necessary.

Step 3. Define the preferences with the help of the GUI or via the registry.

Step 4. Test the expected behaviour by initiating traffic.

Step 5. Export the following registries:

- Export the *`HKEY_CURRENT_USER\Software\Balasys\AuthAgent`* registry to the *hlcuaa.reg* file, which contains the user settings for AA. The result shall be as follows:

```
Windows Registry Editor Version 5.00

[HKEY_CURRENT_USER\Software\Balasys]

[HKEY_CURRENT_USER\Software\Balasys\AuthAgent]
"HasPreferences"=dword:00000000
"TLS"=dword:00000001
"Automatic"=dword:00000001
"Details"=dword:00000000
"CanRemember"=dword:00000001
"ForgetPassword"=dword:00000000
"ForgetPasswordInterval"=dword:00000001
```

- Export the *`HKEY_LOCAL_MACHINE\SOFTWARE\Balasys\AuthAgent`*, which contains the AA Multiplexer settings, into the *hklmaa.reg* file. The result shall be as follows:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Balasys]

[HKEY_LOCAL_MACHINE\SOFTWARE\Balays\AuthAgent]
"InstallLang"="1033"
```

The *service private certificate store*, used by the AA Multiplexer, can also be deployed as a registry key.

- Export the *HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Services\auth-agent-mpxd* registry to the *hklmaacert.reg* file. The result shall be as follows:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Services\auth-agent-mpxd]

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Services\auth-agent-mpxd\
SystemCertificates]

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Services\auth-agent-mpxd\
SystemCertificates\My]

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Services\auth-agent-mpxd\
SystemCertificates\My\Certificates]

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Services\auth-agent-mpxd\
SystemCertificates\MY\Certificates\6421DCB8501C2E1F15DB8BD3A94F435C01DB7CD3]
"Blob"=hex:03,00,00,00,01,00,00,00,14,00,00,00,64,21,dc,b8,50,1c,2e,1f,15,db,\

  ...
  ...
  ...
  ...
  ...
  64,0a,87,e9,45,99,04,9e,28,cb,c0,6c,2a,e5,c7,cb,ce,29,d8,b1,e1
```

Note
Note that there can be several empty paths created by the system automatically, which can be included safely.

For further details on registries, see *Section 4.1.1, Registry entries on Microsoft Windows platforms (p. 10)*.

As a result, there will be four registries exported.

Step 6. Switch to the GPO administrator system and download the AA *msi flavor* installer and place it in the Windows share where the other remotely installled applications are stored.

Step 7. Continue with the procedures detailed in section *Procedure 4.1.5, Configuring Group Policy Object (GPO) deployment (p. 21)*

3.2. Procedure – Using AA on GNU/Linux platforms

Purpose:

To run AA on a GNU/Linux system, complete the following steps.

Steps:

Step 1. Make the AppImage file executable:

- In the terminal, enter the following command: `chmod a+x authentication-agent-2.0.0-x86_64.AppImage`.

Step 2. Run the AppImage file:

- In the terminal, enter the following command: `./authentication-agent-2.0.0-x86_64.AppImage`.

Chapter 4. Configuring Authentication Agent (AA)

4.1. Configuring Authentication Agent on Microsoft Windows platforms

4.1.1. Registry entries on Microsoft Windows platforms

Some settings of Authentication Agent (AA) can be modified through the Windows Registry. Launch the registry editor by issuing the `regedit` command (either from a command prompt or through the **Start** button).

In the Registry Editor, the Authentication Agent parameters are located under: *`HKEY_LOCAL_MACHINE\SOFTWARE\Balasys\AuthAgent`* for the Multiplexer and *`HKEY_CURRENT_USER\Software\Balasys\AuthAgent`* for the Client application.

The component has to be restarted if a value is modified (that is, the **Authentication Agent Multiplexer** service for Authentication Agent Multiplexer, the Authentication Client application for Authentication Agent).

To restart the Authentication Agent Multiplexer, select the **Start** button, type **Services** and then press **Enter**. Select **Authentication Multiplexer** on the list, then **Restart** it.

The following settings are available from the registry:

The following table presents the available settings from the registry for the Client application. (These setting may not exists by default, and should be created to override default behaviour)

Name	Description	Default value
Automatic	To enable the automatic Kerberos authentication without user interaction with the Authentication Agent, set it to `1`. In this case, Authentication Agent will use the username provided during Windows login.	0
CanRemember	To save your credentials so that the client will fill the username and password automatically for later authentication attempts, set this parameter to `1`. If it is set to `0`, the credentials will not be saved and have to be reentered again.	1
Details	The Authentication Agent displays the details of the connection in the popup dialog if this parameter is set to `1`. The following information is displayed: the name of the application initiating the connection, the IP address and the port of the destination server, the name of the PNS service started, and the type of the connection (TCP/UDP). If the details are disabled, only the name of the service is displayed.	0

Name	Description	Default value
ForgetPassword	To enable password expiration defined by ForgetPassword interval, set this value to 1. Default value of 0 disables password expiration.	0
ForgetPasswordInterval	To prevent unauthorized initiation of network connections through unattended machines, configure this parameter. Enter the number of minutes after which Authentication Agent deletes the stored password and requires authentication for new connection requests.	1
HasPreferences	To enable the **Preferences** menu item in the system tray icon of Authentication Agent, set this parameter to `1`. Otherwise, this menu item will not be available.	1
LogClient	It marks the verbosity level of the authentication client, ranging from `0` (lowest) to `9`. Increase the log verbosity only if it is necessary (for example, for troubleshooting purposes), because setting it to higher than `3` can result in very large log files. The log file is stored in the user's home directory.	0

Table 4.1. Registry setting options for the Client application

The following table presents the available settings from the registry for the Multiplexer.

Name	Description	Default value
AliasFile	This is the name and path (for example, `C:\tmp\aliases`) of a text file. Using the information contained in this file, the Authentication Agent Multiplexer can redirect the authentication of certain users to a different user in multi-user environments. For example, to redirect the connection authentication of the `Administrator` user to `MainUser` enter the following line: `Administrator: MainUser`.	
Log	It is the verbosity level of the Authentication Agent Multiplexer, ranging from `0` (lowest) to `9`. Increase log verbosity only if it is necessary (for example, for troubleshooting purposes), because setting it to higher than `3` can result in very large log files.	0

Name	Description	Default value
	The log file is stored in the `%SYSTEMROOT%\System32\config\systemprofile` folder.	
TLS	To configure the Authentication Agent Multiplexer so that it uses only TLS-encrypted connections, set this parameter to `1`.	1
VerifyDepth	It is the maximum length of the verification chain.	3

Table 4.2. Registry setting options for the Multiplexer

4.1.2. Command line parameters on Microsoft Windows platforms

To display the version number of the client, enter `auth-agent-client.exe --version`.

The Authentication Agent Multiplexer (`auth-agent-mpxd.exe`) has the following command line options:

`--install_service`	It registers the Authentication Agent service.
`--remove_service`	It removes the Authentication Agent service.
`--start_service`	It starts the Authentication Agent service.
`--stop_service`	It stops the Authentication Agent service.

4.1.3. Configuring TLS connections on Microsoft Windows platforms

Authentication Agent Multiplexer and PNS can communicate through an TLS-encrypted channel. For this, a certificate has to be available on the PNS firewall that PNS uses to authenticate the connection to the Authentication Agent Multiplexer. The Authentication Agent Multiplexer verifies this certificate using the certificate of the CA issuing PNS's certificate, therefore the certificate of the CA has to be imported to the machine running the Authentication Agent.

Note
During authentication, when PNS communicates with AA, AA expects TLS-encrypted communication. In order to disable this and to use the communication without encryption (which is strongly against the recommendation, but useful for debugging purposes), the TLS encryption shall be disabled by setting the *`TLS`* registry key to value '0'. For details on this parameter, see *Section 4.1, Configuring Authentication Agent on Microsoft Windows platforms (p. 10)*. Also see, *Procedure 3.1.1, Installing the Authentication Agent on Microsoft Windows (p. 4)*.

Note
It is highly recommended to encrypt the communication between PNS and the Authentication Agent, because without it, anyone can connect to the Authentication Agent Multiplexer, resulting in the authentication information obtained by unauthorized people. It is essential to use encryption when password authentication is used. For details on encryption, see *Procedure 3.1.1, Installing the Authentication Agent on Microsoft Windows (p. 4)*.

4.1.3.1. Procedure – Encrypting the communication between PNS and the Authentication Agent on Microsoft Windows platforms

Purpose:

To enable encryption between PNS and the Authentication Agent, complete the following steps. For the steps to be completed from Management Console (MC), see *Chapter 11, Key and certificate management in PNS* in *Proxedo Network Security Suite 2 Administrator Guide*.

Steps:

Step 1. Create a CA (for example, `AA_CA`) using the Management Console (MC). This CA will be used to sign the certificates shown by the PNS firewalls to the Authentication Agents.

Step 2. Export the CA certificate into `DER` format.

Step 3. Generate certificate request(s) for the PNS firewall(s) and sign it with the CA created in Step 1.

Note
Each firewall shall have its own certificate. Do not forget to set the firewall as the **Owner host** of the certificate.

Step 4. Distribute the certificates to the firewalls.

Step 5. Install the Authentication Agent (AA) application to the workstations and import to each machine the CA certificate exported in Step 2.
There are three ways to import the CA certificate:

1. Import the CA certificate by using the installer of the Authentication Agent.
2. Import the CA certificate manually by using the `addcert` and `getcert` programs (see *Procedure 4.1.3.2, Importing the CA certificate manually (p. 13)*).
3. You can also import the CA certificate by using the `Microsoft Management Console` (see *Procedure 4.1.3.3, Importing the CA certificate using Microsoft Management Console (MMC) (p. 14)*).

Step 6. Create the appropriate outband authentication policies in MC and reference them among the services of PNS. See *Chapter 15, Connection authentication and authorization* in *Proxedo Network Security Suite 2 Administrator Guide* for details.

4.1.3.2. Procedure – Importing the CA certificate manually

Procedure:

To import the certificate of the CA using the `addcert` and `getcert` programs, complete the following steps.

Steps:

Step 1. The certificate can be imported using the `addcert.exe` program located in the installation folder of the Authentication Agent (`C:\Program Files\auth-agent` by default). The program can be started from a command prompt. Provide the name and the path of the `DER`-formatted certificate as an input parameter, for example:

```
C:\Program Files\auth-agent\bin\addcert.exe C:\temp\AuthenticationAgent_CA.crt
```

Note
Running `addcert.exe` requires administrator privileges.

Step 2. Verify that the certificate has been successfully imported by running `getcert.exe`. Running `getcert.exe` lists the Subject of all imported certificates.

Step 3. Restart the **Authentication Agent Multiplexer** service.

4.1.3.3. Procedure – Importing the CA certificate using Microsoft Management Console (MMC)

Purpose:

To import the certificate of the CA complete the following steps.

Steps:

Step 1. Start Microsoft Management Console by executing `mmc.exe` after selecting the **Start** button.

Note
Running `mmc.exe` requires administrator privileges.

Step 2. Select **Add/Remove Snap-in**, from the **File** menu.

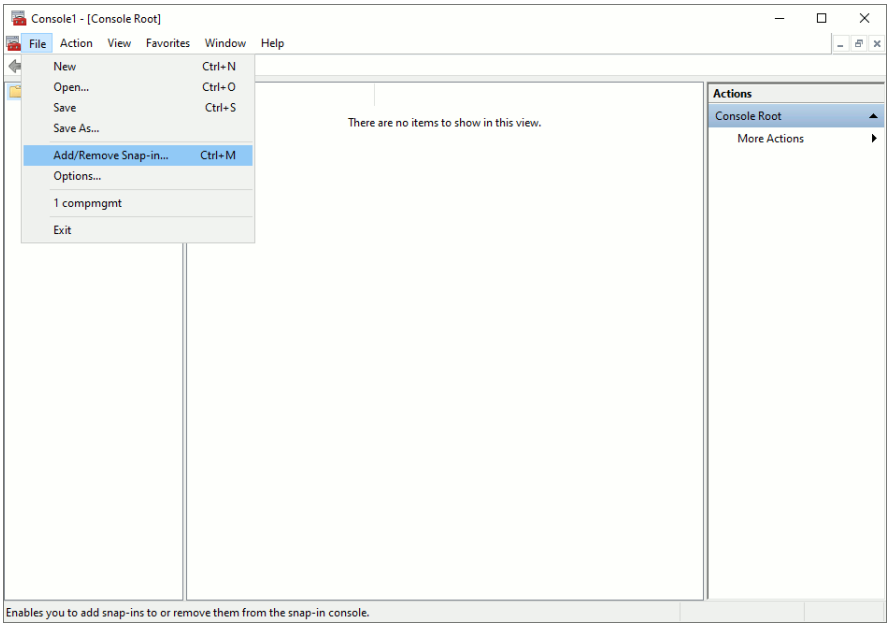

Figure 4.1. Adding a snap-in

Step 3. Select **Certificates** and click **Add** from the **Available snap-ins** list.

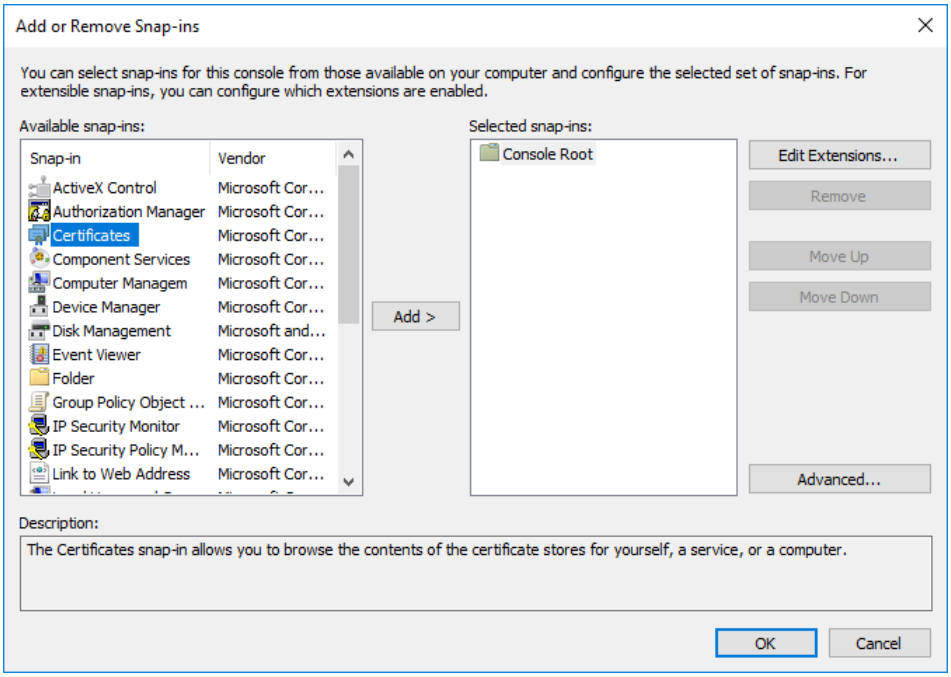

Figure 4.2. Adding certificates

Step 4. Select **Service account** and click **Next**.

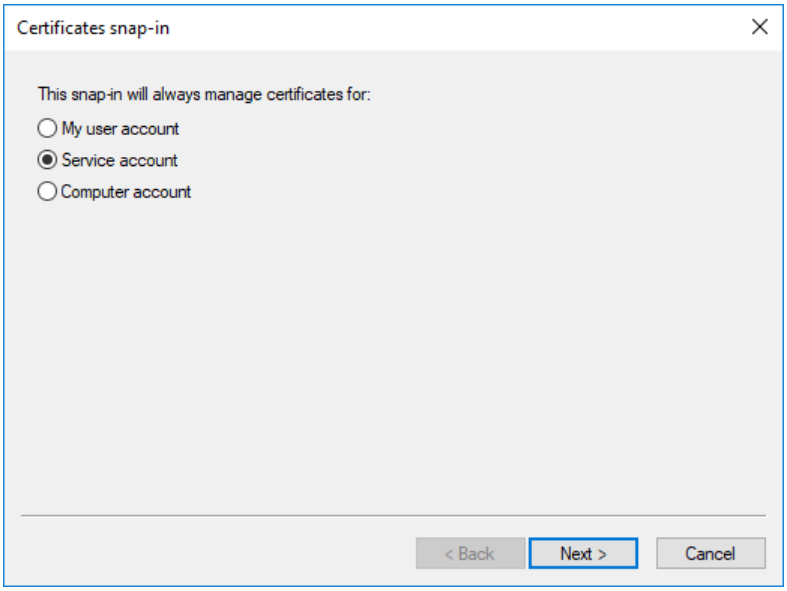

Figure 4.3. Selecting the service account

Step 5. Select **Local menu** and click **Next**.

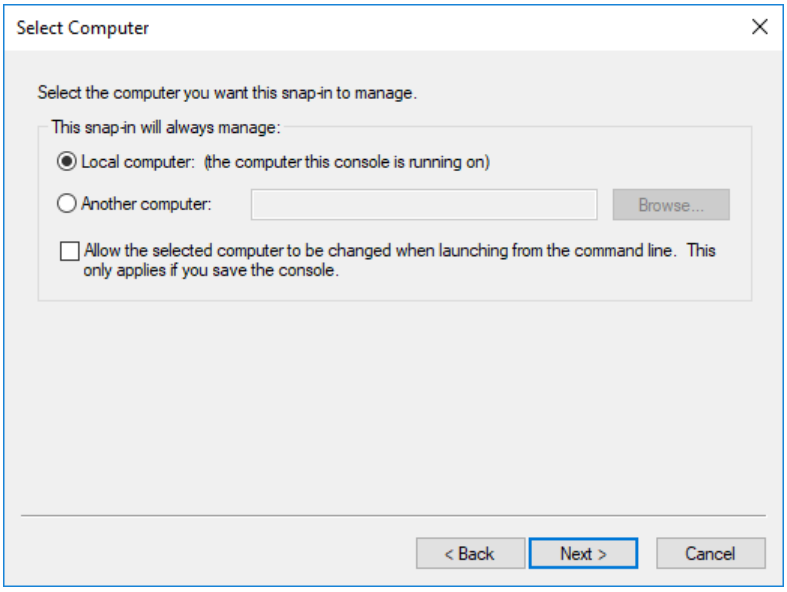

Figure 4.4. Selecting the managed computer

Step 6. Select the **Authentication Agent Multiplexer** service and click **Finish**.

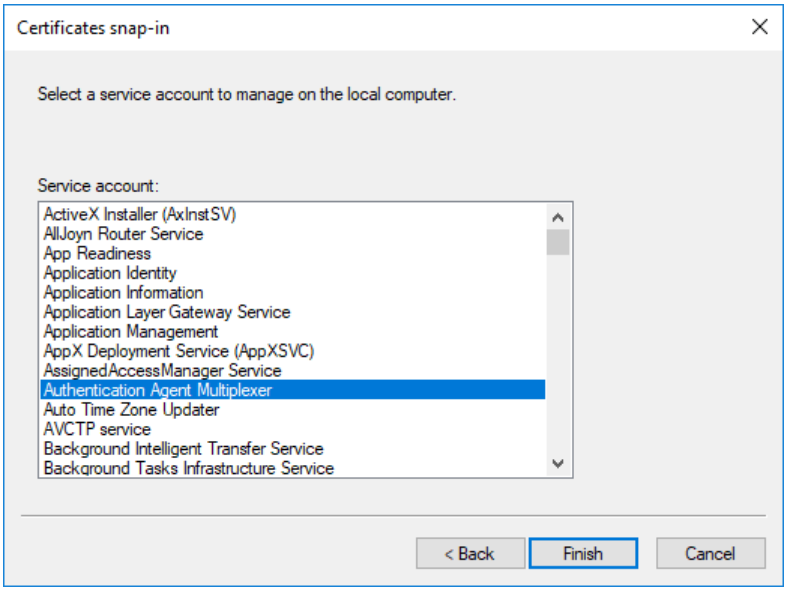

Figure 4.5. Selecting the service

With the above steps a snap-in module has been configured that enables to conveniently manage the certificates related to the Authentication Agent Multiplexer.

Step 7. Navigate to **Certificates - Service (Authentication Multiplexer) > auth-agent-mpxd\Personal > Certificates**, and click **Add**.

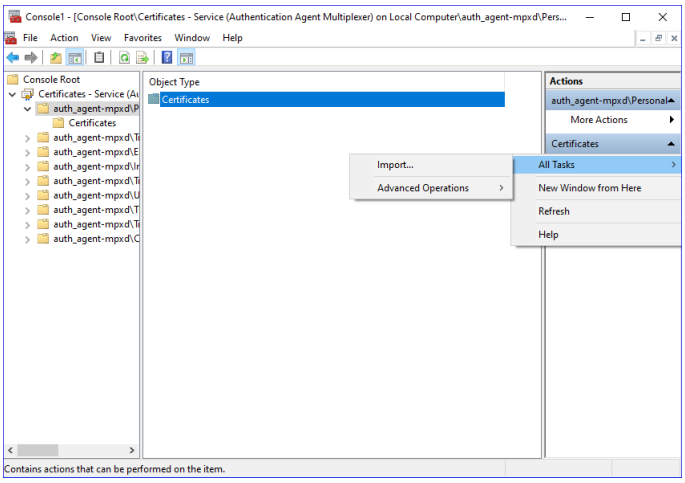

Figure 4.6. Importing the CA certificate

Step 8. Right-click **Certificates**, navigate to **All tasks > Import**. The **Certificate Import Wizard** is displayed. Click **Next**.

Step 9. Select the certificate to import and click **Next**.

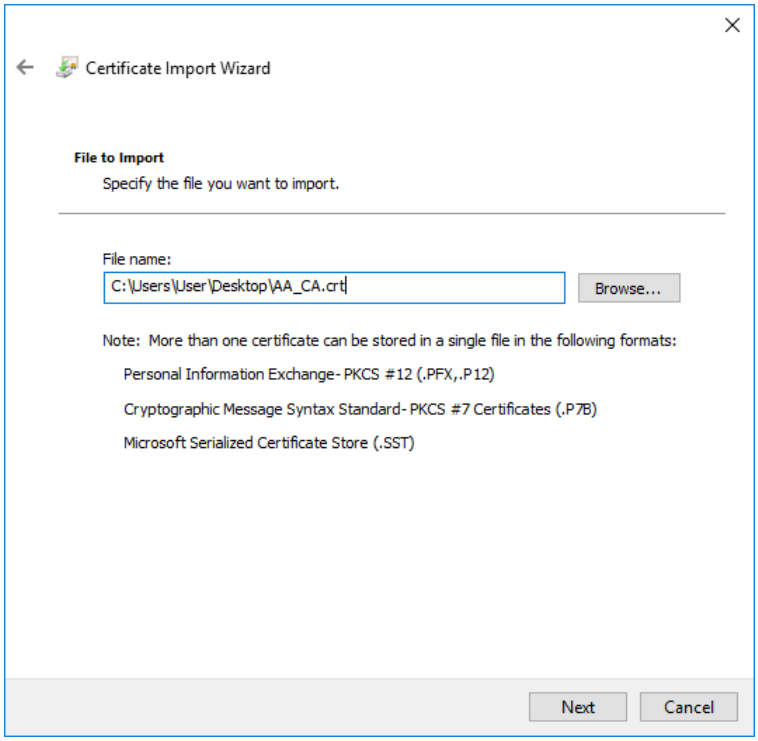

Figure 4.7. Selecting the certificate to import

Step 10. Click **Next**, when Windows offers a suitable certificate store by default.

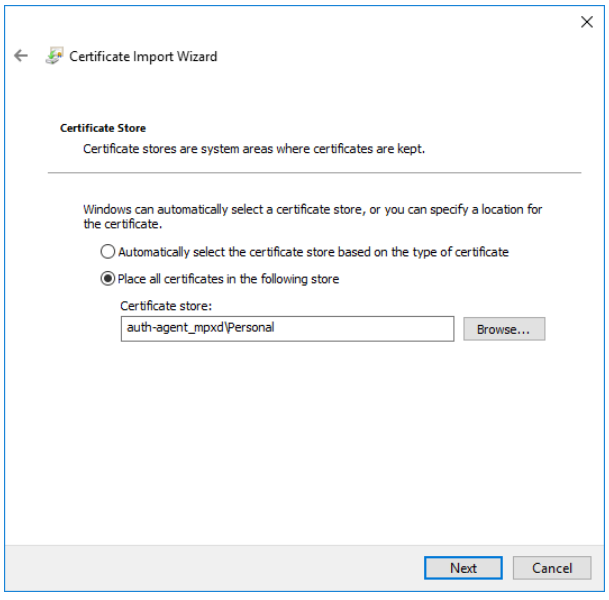

Figure 4.8. Selecting the certificate store

Step 11. Click **Finish** on the summary window and **OK** on the window that marks the successful import of the certificate.

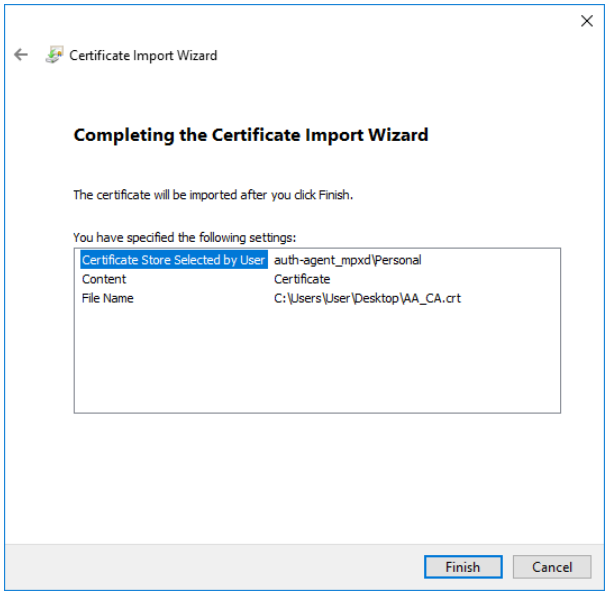

Figure 4.9. Summary

The main window of MMC is displayed with the imported certificate.

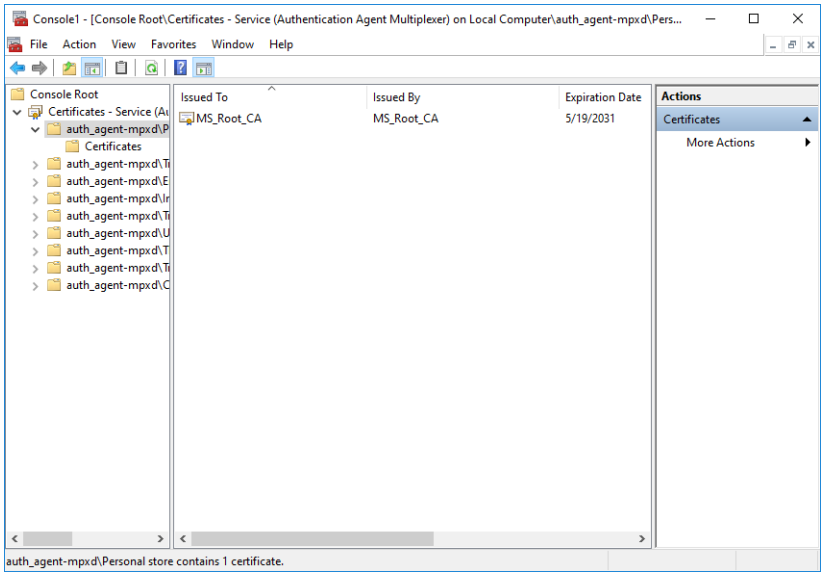

Figure 4.10. The imported certificate

Step 12. Restart the Authentication Agent service. Scroll to the **Authentication Agent Multiplexer** among the list of Services and right-click on it. Navigate to **All Tasks** > **Restart**.
It is also possible to start and stop the Authentication Agent here.

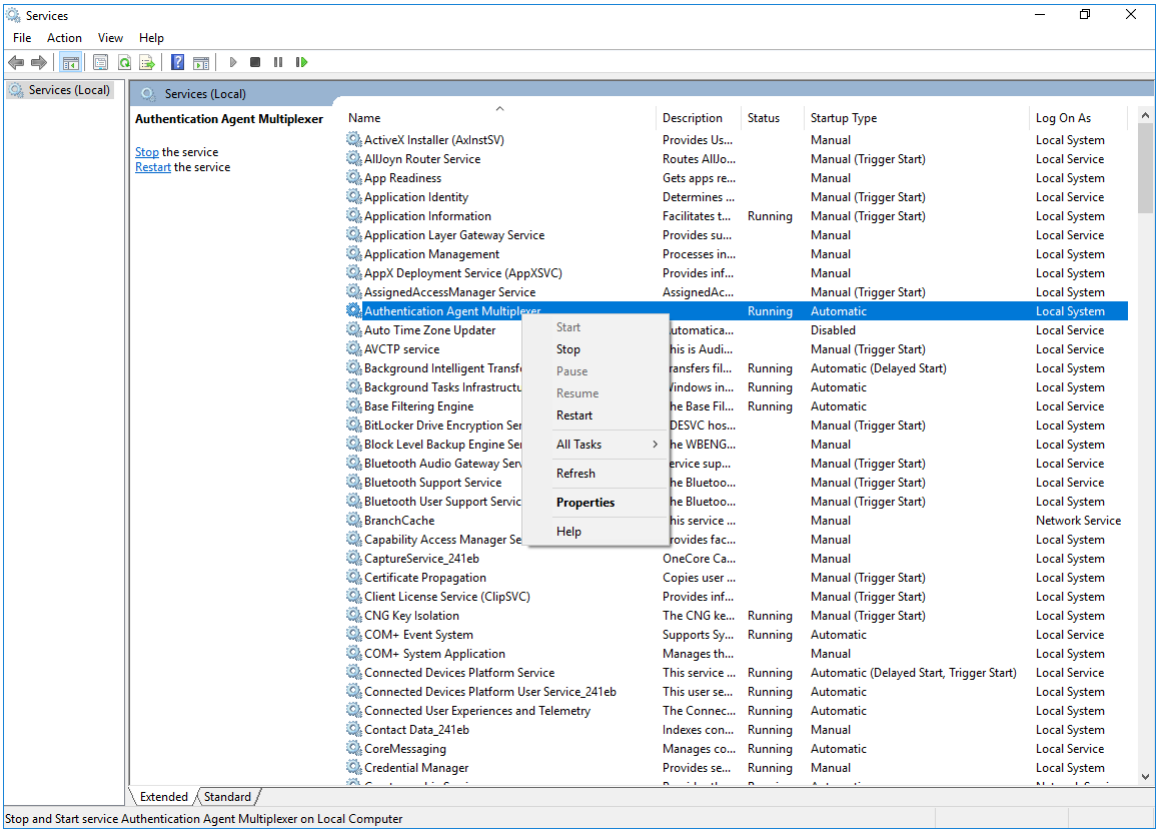

Figure 4.11. Restarting the Authentication Agent

4.1.4. Procedure – Configuring X.509 certificate based authentication on Microsoft Windows platforms

Purpose:

For authentication based on X.509 certificates the certificate and the private key of the user has to be deployed onto the workstation. Import the certificate of the user into their personal certificate store. This can be accomplished most easily through the **Certificates** Control Panel item.:

Steps:

Step 1. Click the **Start** button and type **Manage user certificates** then press **Enter**.

Step 2. Navigate to **Certificates - Current User** > **Personal** > **Certificates**.

Step 3. Right-click **Certificates** and navigate to **All tasks - Import**.
The **Certificate Import Wizard** is displayed.

Note
Hardware keys and tokens having a suitable driver for Windows are also displayed in this store and can be used from the Authentication Agent.

Step 4. Import the certificate, using the **Certificate Import Wizard** tool.

4.1.5. Procedure – Configuring Group Policy Object (GPO) deployment

Step 1. Import all four registry files to the GPO configurator system, so that the Registry Wizard can browse them. Later, remove the registry information if it is no longer required. If it is not possible to remove them, all four files have to be configured as registry keys.

Step 2. Create a new policy to the corresponding forest as *AA deployment*.

Step 3. Configure the corresponding parameters, as, for example, target scope or filetring and so on.

Step 4. Edit the *AA Deployment* policy.

Step 5. Add the *AA msi installer* as a new package under the **Computer Configuration/Policies/Software Settings/Software installation** path.

Step 6. Browse the network share for the newly added package, select it, and set it to Auto installation.

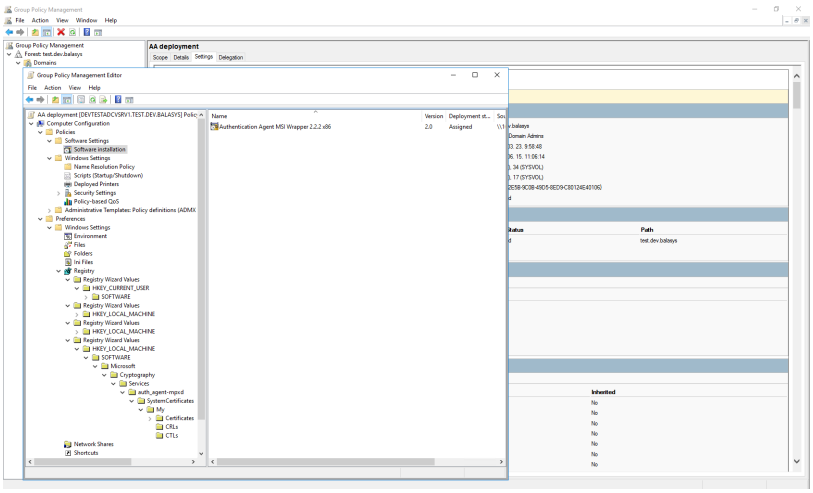

Figure 4.12. The result of auto installation

Step 7. Import all four registry settings with the help of the Registry Wizard. The *HKLM* registries under the **Computer Configuration/Preferences/Windows Settings/Registry** path, and the *HKCU* registries under the **User Configuration/Preferences/Windows Settings/Registry** path.

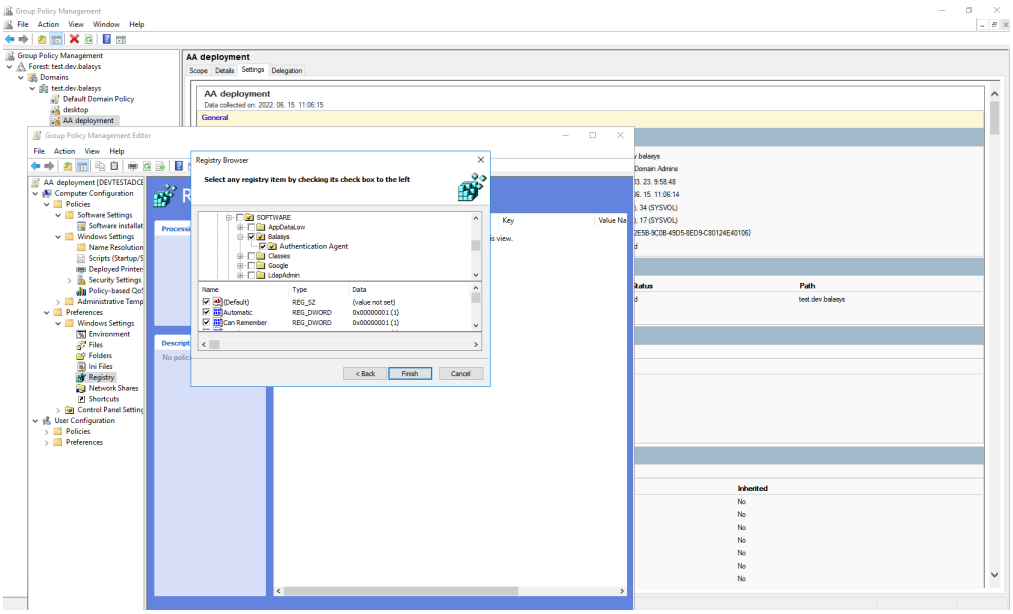

Figure 4.13. Importing registries

Step 8. Close the GP editor.

4.1.6. Procedure – Enabling Kerberos authentication in AS

Complete the following steps to enable Kerberos authentication in Authentication Server using Windows Active Directory (AD) environment.

Steps:

Step 1. In MC select **Authentication Server > Instances > Edit**.

Step 2. Select the **GSSAPI/Kerberos5** checkbox at **Methods** section and provide the *realm* at **Principal name** field.

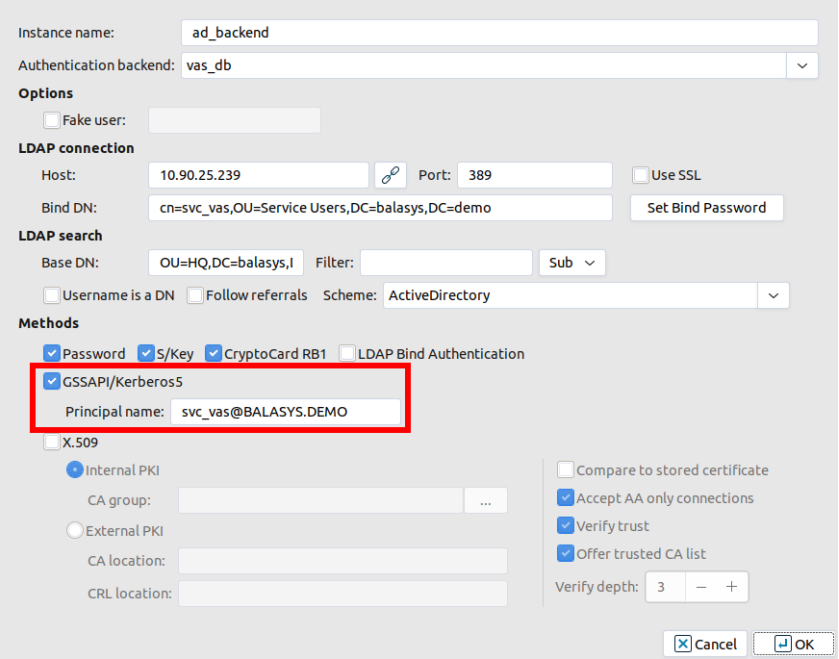

Figure 4.14. Providing Kerberos realm

Step 3. Create the domain user in the **Active Directory**. Use the **Principal name** provided in the previous step.

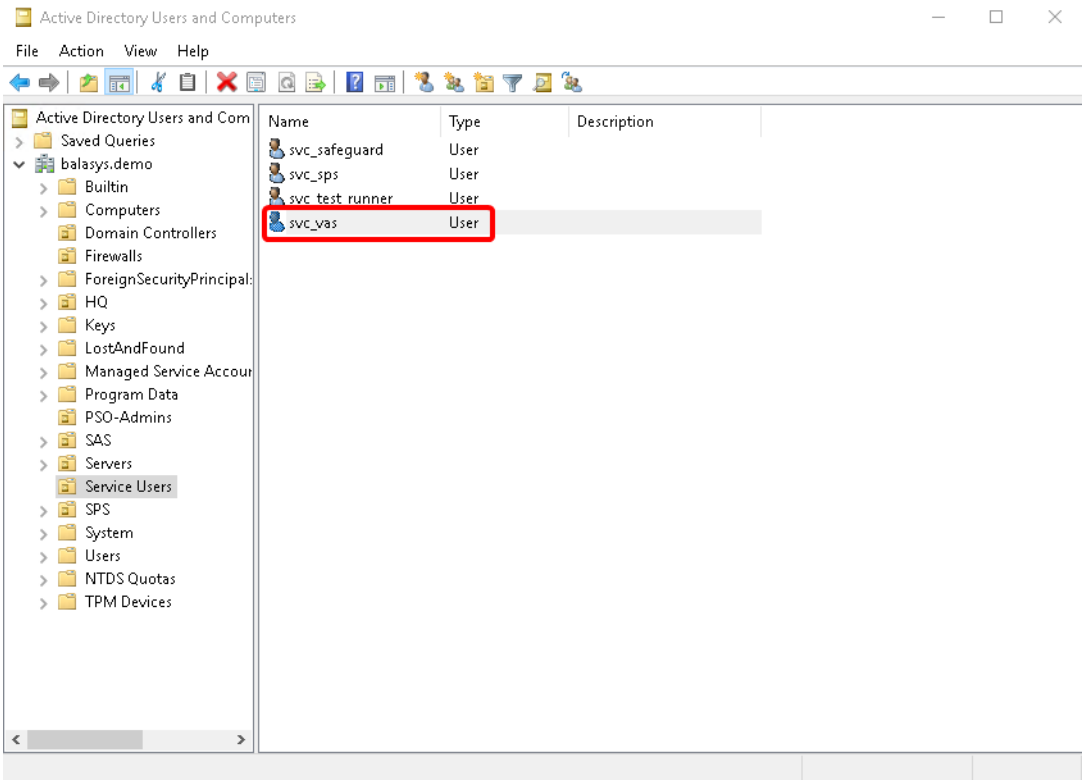

Figure 4.15. Creating the domain user

Step 4. Start the Command Prompt in the Domain Controller with Administrator privileges.

Step 5. Run the following command:

```
setspn -a http/ <username> <username>
```

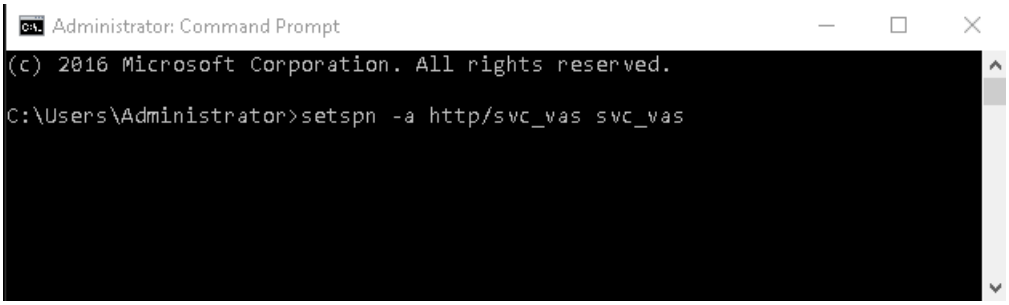

Figure 4.16. Running the command

Step 6. In the **Active Directory** window, select the user created in Step 3. and open the user's **Properties**.

Step 7. A new **Delegation** tab is available now. Select the **Trust this user for delegation to any service (Kerberos only)** option. Click **Apply**.

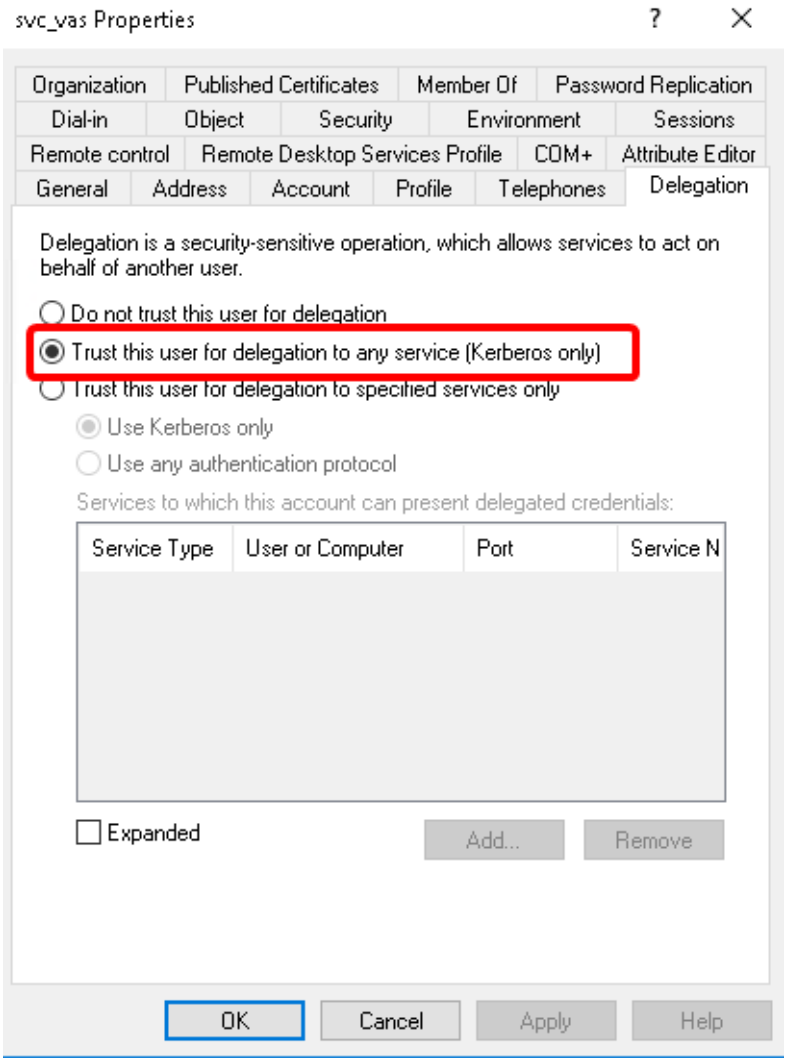

Figure 4.17. Authenticating a user

Step 8. Switch to the **Account** tab in the **Properties** menu item. Select the **This account supports Kerberos AES 256 bit encryption** option and click **OK** to apply the setting.

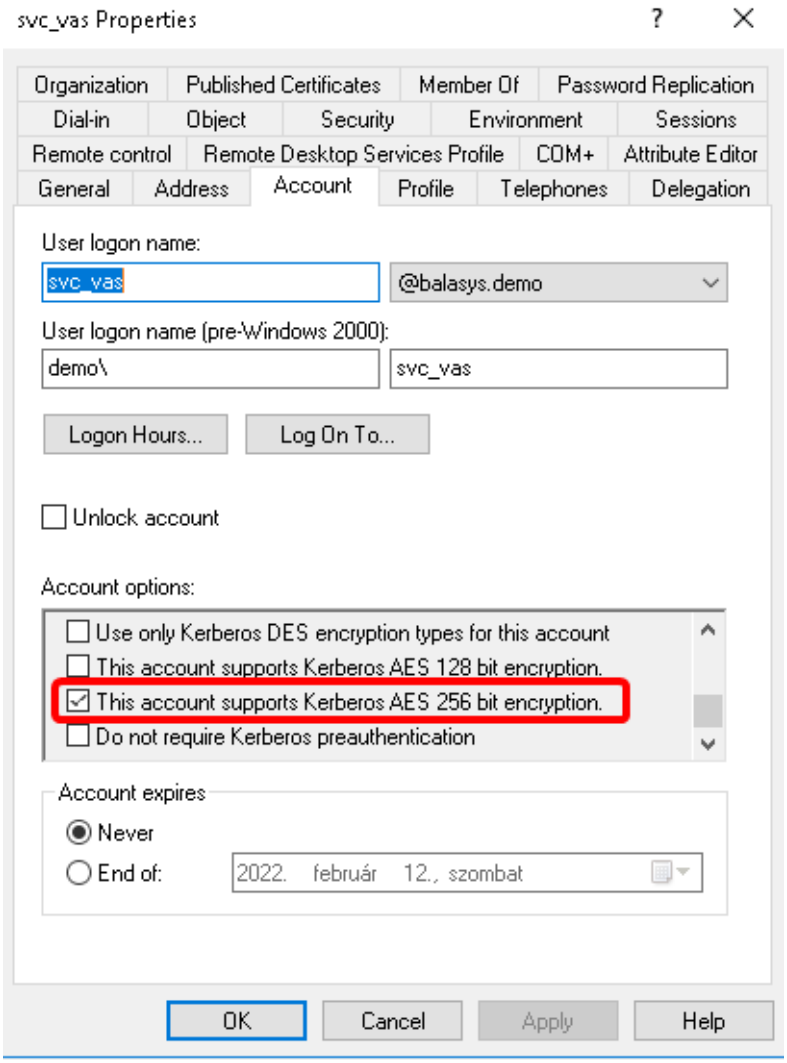

Figure 4.18. setting encryption

Step 9. Install the Kerberos packages on the required server, for example on Authentication Agent.

```
#:apt-get install krb5-user
```

Step 10. Provide the FQDN of the default realm during the installation process.

Step 11. Test Kerberos with the following commands. In the example the FQDN is BALASYS.DEMO.

```
#:kinit svc_vas@BALASYS.DEMO
#:klist -e
#:kdestroy
```

Step 12. Set Kerberos with the following commands:

```
#:ktutil
ktutil:addent -password -p svc_vas@BALASYS.DEMO -k 1 -e
aes256-cts-hmac-sha1-96
```

```
ktutil:addent -password -p svc_vas@BALASYS.DEMO -k 2 -e
aes256-cts-hmac-sha1-96
ktutil:addent -password -p svc_vas@BALASYS.DEMO -k 3 -e
aes256-cts-hmac-sha1-96
ktutil:addent -password -p svc_vas@BALASYS.DEMO -k 4 -e
aes256-cts-hmac-sha1-96
ktutil:addent -password -p svc_vas@BALASYS.DEMO -k 5 -e
aes256-cts-hmac-sha1-96
ktutil:addent -password -p svc_vas@BALASYS.DEMO -k 6 -e
aes256-cts-hmac-sha1-96
ktutil:wkt /etc/krb5.keytab
ktutil:exit
#:chown vas /etc/krb5.keytab
```

4.2. Configuring AA on Linux platforms

4.2.1. Command line parameters on Linux platforms

The graphical client (`auth-agent-gtk`) has the following command line parameters:

`--help` or `-?`	It displays a brief help message.
`--version` or `-V`	It displays version number and compilation information.
`--automatic` or `-a`	It enables automatic Kerberos authentication.
`--no-syslog` or `-l`	It sends log messages to the standard output instead of syslog.
`--verbose <verbosity>` or `-v <verbosity>`	It sets verbosity level to <verbosity>. The default verbosity level is 3; the possible values are `0-10`.
`--logtags;` or `-T`	It prepends log category and log level to each message.

Authentication Agent Multiplexer (`auth-agent-mpxd`) has the following command line parameters:

`--help` or `-?`	It displays a brief help message.
`--version` or `-V`	It displays the version number of `auth-agent-mpxd`.
`--no-syslog` or `-l`	It sends log messages to the standard output instead of syslog.
`--verbose <verbosity>` or `-v <verbosity>`	It sets verbosity level to <verbosity>. The default verbosity level is 3; possible values are `0-10`.
`--logtags;` or `-T`	It prepends log category and log level to each message.
`--aliasfile;` or `-a`	It is the name (including full path) of a text file (for example, `/tmp/aliases`) used by Authentication Agent Multiplexer to redirect the authentication requests of certain users to a different user in multiuser environments. For example, to redirect all authentication request of the *`root`* user to *`MainUser`* add the following line to the file: *`root: MainUser`*.
`--log-spec;` or `-s`	It sets verbosity mask on a per category basis. Each log message has an assigned multi-level category, where levels are separated by a dot. For example, HTTP requests are logged under

	http.request. The `<spec>` is a comma-separated list of log specifications. A single log specification consists of a wildcard matching log category, a colon, and a number specifying the verbosity level of that given category. The categories match from left to right, for example, `--logspec 'http.*:5,core:3'`. The last matching entry will be used as the verbosity of the given category. If no match is found the default verbosity specified with `--verbose` is used.
`--no-require-tls;` or `-S`	It turns off the TLS encryption of the communication between PNS and the Multiplexer.
`--bind-address;` or `-b` and , `--bind-port;` or `-p`	It is the IP address and the port, the Multiplexer is accepting connections on.
`--crt-dir;` or `-t`	It is the path of the directory containing the certificate of the CA that issued the certificate of the PNS firewall.
`--crl-dir;` or `-r`	It is the path of the directory containing the Certificate Revocation List (CRL) related to the above CA.

4.2.2. Configuring TLS connections on Linux platforms

To enable encryption between PNS and the Authentication Agent complete the following steps. For the steps to be completed from MC, see *Chapter 11, Key and certificate management in PNS* in *Proxedo Network Security Suite 2 Administrator Guide*.

Note
During authentication, when PNS communicates with AA, AA expects TLS-encrypted communication. In order to disable this and to use the communication without encryption (which is strongly against the recommendation, but useful for debugging purposes), the TLS encryption shall be disabled by setting the *`--no-require-tls; or -S`* command line parameter.

4.2.2.1. Procedure – Encrypting the communication between PNS and the Authentication Agent on Linux platforms

Steps:

Step 1. Create a CA (for example, *`AA_CA`*) using the Management Console (MC). This CA will be used to sign the certificates shown by the PNS firewalls to the Authentication Agents.

Step 2. Export the CA certificate into `PEM` format.

Step 3. Generate certificate request(s) for the PNS firewall(s) and sign it with the CA created in Step 1.

Note
Each firewall shall have its own certificate. Do not forget to set the firewall as the **Owner host** of the certificate.

Step 4. Distribute the certificates to the firewalls.

Step 5. Install the Authentication Agent (AA) application to the workstations and import to each machine the CA certificate exported in Step 2.
To import the CA certificate complete the following steps:

Step a. Create the `/etc/auth-agent/ca` directory:
```
mkdir /etc/auth-agent/ca
```

Step b. Copy the certificate exported into `PEM` format in Step 2 into the `/etc/auth-agent/ca` directory.

Step c. Create symlinks to the certificate files:
```
c_rehash .
```

Step d. Restart the **Authentication Agent Multiplexer daemon**:
```
systemctl restart auth-agent-mpxd.service
```

The authentication client is now ready to accept encrypted connections from PNS.

Step 6. Create the appropriate outband authentication policies in MC and reference them among the services of PNS. For details, see *Chapter 15, Connection authentication and authorization* in *Proxedo Network Security Suite 2 Administrator Guide*.

4.2.3. Configuring X.509 certificate-based authentication on Linux platforms

For authentication based on X.509 certificates the certificate and the private key of the user has to be deployed onto the workstation. Create a directory called `.auth-agent` in the home folder of the user and copy the certificate and private key of the user in `PEM` format into this directory. Use the `cert.pem` and `key.pem` filenames, or create symlinks with these names pointing to the certificate and the key file. The Authentication Agent will automatically use the certificate found in this directory.

Chapter 5. Using the Authentication Agent (AA)

Purpose:

When the user launches an application that requires authentication (for example, a web browser, e-mail client, and so on) the PNS firewall automatically displays the authentication client on the user's screen.

The client displays the name of the service requiring authentication (`intra_http` in the above example), and — provided that the administrator enabled it — further details of the connection (for example, destination IP address).

Steps:

Step 1. To save your credentials so that the client will fill in the username and password automatically for later authentication attempts, select **Remember password**. For details on configuring password storage period length and deleting a previously saved password, see *Procedure 6., (p. 33)*.
To cancel the authentication at any time, click **Abort**.

Step 2. Enter your user name in the **Enter your user name** field and click **Next**.

Figure 5.1. The Authentication Agent

Step 3. Select the authentication method to use from the **Select authentication method** list. The list displays only the methods that are available for this user.

Figure 5.2. Selecting authentication method

Step a. To authenticate with a password, select **Password authentication**.

Step b. To use Kerberos authentication, select **GSSAPI authentication**.

Note
When using Kerberos authentication the authentication client is not displayed if you have configured **Automatic Kerberos authentication** in **Preferences**. For details, see *Procedure 6., (p. 33)*.

Step c. To authenticate with an X.509 certificate, select **X.509 certificate**.

Step 4. Provide the information required for the selected authentication method. For example, for **Password authentication**, enter your password.

Figure 5.3. Entering the password

Note
After successful authentication, the window of the authentication client is closed automatically, and the connection to the target server is established. If the authentication fails, the client displays an error message.

Chapter 6. Configuring Authentication Agent preferences

Purpose:

Authentication Agent is launched on desktop environment startup, and places its icon on the system tray. To configure Authentication Agent preferences, complete the following steps.

Note
To display the version number and other information about Authentication Agent, right-click the system tray icon and click **About**.

Steps:

Step 1. Right-click the system tray icon and click **Preferences**.

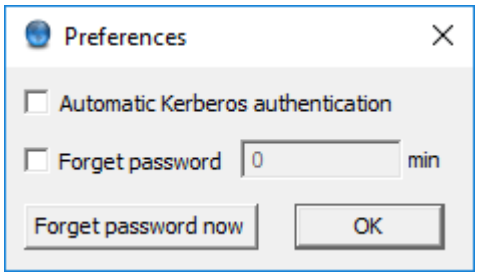

Figure 6.1. Preferences

Step 2. To enable automatic Kerberos authentication without user interaction with the Authentication Agent, select **Automatic Kerberos authentication**. In this case, Authentication Agent will use the username provided during Windows or Linux desktop session login.

Step 3. To prevent unauthorized initiation of network connections through unattended machines, configure **Forget password**. Enter the number of minutes after which Authentication Agent deletes the stored password and requires authentication for new connection requests.

Step 4. To immediately delete the stored password from the Authentication Agent and require authentication for new connection requests, click **Forget password now**.
AA stores its preferences in the `~/.config/aa/aa.cfg` configuration file on Linux, and in the Windows Registry on Microsoft Windows platforms, for more information see *Section 4.1.1, Registry entries on Microsoft Windows platforms (p. 10)*.

Chapter 7. Starting and stopping Authentication Agent

To start or stop Authentication Agent, perform one of the following steps.

- To stop Authentication Agent, right-click the system tray icon and click **Exit**.
- To restart the Authentication Agent select the **Start** button, type **Authentication Agent** and then press **Enter**.