Caution

Before you start the implementation, make sure you have the latest version of this document. You can find the latest version at the following location: service.sap.com/instguides.

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0</td>
<td>2016-11-04</td>
<td>Document version for PCo 15.1 (SP03)</td>
</tr>
</tbody>
</table>
1 SAP Plant Connectivity

1.1 Introduction

This document outlines the available options in the Microsoft Windows operating system that you use to implement the security policy that controls the access to SAP Plant Connectivity (PCo).

⚠️ Caution
This guide does not replace the administration or operation guides that are available for productive operations.

ℹ️ Note
PCo does not provide built-in user management. It uses standard Microsoft Windows domain user accounts. It does not provide additional roles for controlling the user or group accounts specific to PCo functionality. If a user can launch PCo, there is no limit to the functionality, provided the user has administrator privileges. For more information about user management and security audits, see [technet.microsoft.com/en-us/library/cc781549%28WS.10%29.aspx](technet.microsoft.com/en-us/library/cc781549%28WS.10%29.aspx).

- For a complete list of the SAP Security Guides, see the SAP Service Marketplace at [service.sap.com/securityguide](service.sap.com/securityguide).
- For a complete list of SAP Security Notes, see the SAP Service Marketplace at [service.sap.com/securitynotes](service.sap.com/securitynotes).

1.2 Security Required by Plant Connectivity

With the increasing use of distributed systems and the Internet for managing business data, the demands on security are also on the rise. When using a distributed system, you need to be sure that your data and processes support your business needs without allowing unauthorized access to critical information. User errors, negligence, or attempted manipulation of your system should not result in loss of information or processing time. These demands on security apply likewise to SAP Plant Connectivity (PCo).

The following prerequisites apply to the correct operation of SAP Plant Connectivity:

- PCo can only be installed by a user with administrator privileges.
- Agent instances cannot be created or deleted unless the user has administrator privileges.
- Agent instances that use a specific Microsoft Windows user for execution are granted the Log on as a service privilege.
- If the PCo Management Console is started by a user without administrator privileges it is shown in display mode only. The display mode allows access to all configuration data and logs but does not allow configuration changes or starting and stopping of agents.
2   About this Document

The Security Guide provides an overview of the security-relevant information that applies to PCo.

2.1 Overview of the Main Sections

The Security Guide comprises the following main sections:

- **Before You Start**
  This section contains information about why security is necessary, and how to use this document.

- **Technical System Landscape**
  This section provides an overview of the technical components and communication paths that are used by PCo.

- **User Administration and Authentication**
  This section provides an overview of the following user administration and authentication aspects:
  - Recommended tools to use for user management
  - User types that are required by PCo

- **Authorizations**
  This section provides information about the authorizations required to run PCo.

- **Network and Communication Security**
  This section provides an overview of the communication paths used by PCo and the security mechanisms that apply. It also includes our recommendations for the network topology to restrict access at network level.

- **Data Storage Security**
  This section provides an overview of any critical data that is used by PCo and the security mechanisms that apply.

- **Dispensable Functions with Impacts on Security**
  This section provides an overview of the functions that have an impact on security, but which can be disabled or removed from the system if necessary.
3 Before You Start

3.1 Important SAP Notes

For a list of additional security-relevant SAP Hot News and SAP Notes, see SAP Service Marketplace at service.sap.com/securitynotes.

3.2 Additional Information

For more information about specific topics, see the Quick Links as shown in the table below.

<table>
<thead>
<tr>
<th>Content</th>
<th>Quick Link on SAP Service Marketplace or SDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security</td>
<td>sdn.sap.com/irj/irj/security</td>
</tr>
<tr>
<td>Security Guides</td>
<td>service.sap.com/securityguide</td>
</tr>
<tr>
<td>Related SAP Notes</td>
<td>service.sap.com/notes</td>
</tr>
<tr>
<td></td>
<td>service.sap.com/securitynotes</td>
</tr>
<tr>
<td>Released Platforms</td>
<td>service.sap.com/pam</td>
</tr>
<tr>
<td>Network Security</td>
<td>service.sap.com/securityguide</td>
</tr>
<tr>
<td>SAP Solution Manager</td>
<td>service.sap.com/solutionmanager</td>
</tr>
<tr>
<td>SAP NetWeaver</td>
<td>service.sap.com/solutionmanager</td>
</tr>
<tr>
<td></td>
<td>sdn.sap.com/irj/irj/sdn/netweaver</td>
</tr>
</tbody>
</table>
4 Technical System Landscape

For an overview of the Plant Connectivity system landscape, refer to the corresponding Master Guide in the SAP Service Marketplace.

Table 3

<table>
<thead>
<tr>
<th>Topic</th>
<th>Guide / Tool</th>
<th>Quick Link on SAP Service Marketplace or SDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical description of PCo</td>
<td>Master Guide</td>
<td>help.sap.com/pco#section2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>service.sap.com/instguides under SAP Business Suite Applications</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SAP Manufacturing &gt; Plant Connectivity</td>
</tr>
<tr>
<td>Technical Landscape Design</td>
<td>See applicable technical documents</td>
<td>sdn.sap.com/irj/sdn/landscapedesign</td>
</tr>
<tr>
<td>Security</td>
<td>See applicable security documents</td>
<td>sdn.sap.com/irj/sdn/security</td>
</tr>
</tbody>
</table>
5 User Administration and Authentication

PCo does not have its own user management, but instead uses the standard Microsoft Windows domain user accounts. PCo also does not provide additional roles for controlling user or group accounts specific to PCo functionality.

Note
For more information about user management and security audits, see technet.microsoft.com/en-us/library/cc781549%28WS.10%29.aspx.
6 Authorizations

Prerequisites

To launch the Plant Connectivity Management Console in order to perform configuration changes or to start and stop agents, you must have the corresponding administrator privileges.

Note

Under Windows Vista and higher Windows versions, the UAC (User Account Control) automatically detects that administrator privileges are required.

Users without administrator privileges can start the PCo Management Console in display mode only. They have read-only access to configuration details and logs. They are also allowed to export the configuration.

Agent instances running as a service may be configured to use a specific user. The user account used for this needs the Log on as a service privilege. This can be configured in the Group Policy Editor. You can find more information about this on the Microsoft Developer Network (MSDN) under the following links:

- Log on as a service privilege: msdn.microsoft.com/en-us/library/ms813948.aspx

To control the general access to the Plant Connectivity Management Console, you can use the following technologies for the PCo system folder (usually found under C:\Program Files (x86)\SAP\Plant Connectivity \System):

Table 4

<table>
<thead>
<tr>
<th>Technology</th>
<th>Quick Link</th>
</tr>
</thead>
</table>
7 Communication and Network Security

Your network infrastructure is extremely important in protecting your system. Your network needs to support the communication necessary for your business needs without allowing unauthorized access. A well-defined network topology can eliminate many security threats based on software flaws (at both the operating system level and the application level) or network attacks such as eavesdropping.

If users cannot log on to your application or database servers at the operating system or database layer, then there is no way for intruders to compromise the machines and gain access to the back-end system’s database or files. Additionally, if users are not able to connect to the server LAN (local area network), they cannot exploit well-known bugs and security holes in network services on the server machines.

7.1 Communication Channel Security

The table below shows the communication channels used by PCo, the protocol used for the connection, and the type of data transferred.

<table>
<thead>
<tr>
<th>Communication Path</th>
<th>Protocol Used</th>
<th>Type of Data Transferred</th>
<th>Data Requiring Special Protection</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPC (DA, HDA, and A&amp;E) data source to PCo (and vice versa)</td>
<td>COM/DCOM</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td></td>
</tr>
<tr>
<td>OPC UA data source to PCo (and vice versa)</td>
<td>HTTP (deprecated)/ HTTPS or OPC UA specific protocol via TCP</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td>Password for user session (if used), private key of certificates</td>
</tr>
<tr>
<td>OPC UA destination</td>
<td>Inherited from OPC UA source system</td>
<td>Method call</td>
<td>Inherited from OPC UA source system</td>
</tr>
<tr>
<td>Data source to PCo (and vice versa) (this applies to the following agents: Citect, IP21, PI, PI-AF, Proficy)</td>
<td>Vendor-specific proprietary protocol</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td>Password for logon to data source</td>
</tr>
<tr>
<td>Modbus data source</td>
<td>TCP/Serial Protocol</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td></td>
</tr>
<tr>
<td>PCo to MII (destination)</td>
<td>HTTP/HTTPS</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td>Password for logon to MII</td>
</tr>
<tr>
<td>PCo to Web service destination</td>
<td>SOAP</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td>Password for Web service</td>
</tr>
<tr>
<td>PCo to ABAP NetWeaver AS (destination)</td>
<td>RFC</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td>Password for logon to NetWeaver system</td>
</tr>
<tr>
<td>Communication Path</td>
<td>Protocol Used</td>
<td>Type of Data Transferred</td>
<td>Data Requiring Special Protection</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>------------------------</td>
<td>------------------------------------------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>PCo to OData destination</td>
<td>HTTP/HTTPS</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td>Password for OData service</td>
</tr>
<tr>
<td>PCo to Restful Web service destination</td>
<td>HTTP/HTTPS</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td>Password for Restful Web service</td>
</tr>
<tr>
<td>PCo to Universal Web service destination</td>
<td>HTTP/HTTPS</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td>Password for Web service</td>
</tr>
<tr>
<td>PCo to Sybase ESP destination</td>
<td>HTTP/HTTPS</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td>Password for Sybase ESP</td>
</tr>
<tr>
<td>PCo to ODBC destination</td>
<td>ODBC</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td>Password for ODBC data source</td>
</tr>
<tr>
<td>MII (TagQuery) to PCo</td>
<td>LISA (MII specific binary protocol)</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td></td>
</tr>
<tr>
<td>MII (PCoQuery) and NetWeaver to PCo</td>
<td>xMII data transfer protocol (new MII specific protocol)</td>
<td>Tag values, metadata, and hierarchical structure of tags</td>
<td></td>
</tr>
<tr>
<td>PCo Remote Client to PCo Management Host</td>
<td>NET.TCP (and potentially also WCF (Windows Communication Foundation))</td>
<td>Commands, status information, configurations, and logs</td>
<td>(Encrypted) passwords in configuration data</td>
</tr>
<tr>
<td>MII (PCoQuery) to PCo Management Host</td>
<td>HTTP/HTTPS</td>
<td>Commands, status information, configurations, and logs</td>
<td>(Encrypted) passwords in configuration data</td>
</tr>
<tr>
<td>WebSocket</td>
<td>HTTP/HTTPS</td>
<td>Commands, status information, configurations</td>
<td></td>
</tr>
</tbody>
</table>

Note the following protection options:

- RFC connections can be protected using Secure Network Communications (SNC).
- HTTP connections can be protected using the Secure Sockets Layer (SSL) protocol.
- SOAP connections are protected with Web services security.
- Universal Web service destination allows the usage of tokens to prevent CSRF attacks. For more information, see Application help for Universal Web service destination.

**Recommendation**

We strongly recommend using secure protocols (SSL, SNC) whenever possible.

**DCOM Security**

The OPC standards DA, HDA, and A&E use DCOM security. To implement DCOM security, carry out the following steps:

1. On the machine where the OPC server is running, use the command `dcomcnfg.exe` to carry out the following steps:
   - Check the launch and the activation limits under `My Computer` `Properties` `COM Security`.
   - Check the access limits.
○ Verify the OPCEnum application’s DCOM security settings for launch/activation and access.
○ Verify the respective OPC server application’s DCOM security settings for launch/activation and access.
○ Then make the following settings for the agent instance:
  ○ Grant the Anonymous Logon account local and remote permissions.
  ○ Grant the Everyone group local and remote permissions.

For a simple setup of both, you can grant the Anonymous Logon account and the Everyone group local and remote permissions.
○ Set a valid domain user account.

2. On the machine where PCo is installed and the agent instance is running, use the file dcomcnfg.exe to verify or change the launch and activation limits.
○ OPC server credentials for the local system:
  ○ Grant the Anonymous Logon account local and remote permissions.
  ○ Grant the Everyone group local and remote permissions.
○ Valid domain user account
  The Everyone group covers all authenticated user accounts. You can grant permission to a specific user or group to achieve a higher level of security.

**OPC UA**

This section applies to both the OPC UA source system, the OPC UA server, new as of PCo 15.1 (SP03), and the OPC UA destination system, which inherits its properties from an OPC UA source system.

**Note**

Recommendations for security policies may change over time due to increasing computing power. You should review your settings regularly.

OPC UA security is divided into the following categories:

- **Channel Communication:** The channel communication uses the standard X.509 v3 certificates.
- **User Session:** The user session may use Anonymous, User name and Password, or Credentials.

OPC UA security for channel communication is governed by the OPC UA specification, which has been made open source in 2015. It is based on the usage of X509 v3 certificates for both client and server. It defines several security modes:

- None
- Sign
- SignAndEncrypt

The profile security category lists several policies:

- None
- Basic128Rsa15
- Basic256
- Basic256Sha256

The policies can be combined in certain, predefined ways.

The links above provide detailed information about the algorithms used for each policy and also recommendations on usage.
- Security policy **None** only supports security mode **None** and should be used only for testing or for connecting to devices that do not support other security modes.
- Security policy **Basic128Rsa15** is rated unsecure and should no longer be used, unless legacy servers cannot be connected to securely using other policies.
- Security policy **Basic256** is considered to be a reasonable secure policy if certificates are signed with **Sha256**, even though **Sha1** is supported.

If computation power of a device to which you want to connect is low and confidentiality not an issue, but data integrity is important, you may consider using the security mode **Sign** which transfers data unencrypted but with digital signatures that allow verification of data integrity.

In PCo, the security policies **Basic128Rsa15** and **SHA256** are supported for the security modes **Sign** and **SignAndEncrypt**. The security policy **Basic256SHA256** is currently not supported.

The OPC UA specification details the process of certificate validation for the creation of secure channels (Release 1.03, Part 4, section 6.1.3). This process description allows the creation of a secure channel even though some checks may have failed.

The PCo UA source system and server rely on the (open source) .Net SDK of the OPC Foundation, which currently does not fully support this part of the specification, meaning that the same holds for SAP Plant Connectivity OPC UA applications. This means for the UA source system, in particular:

- Certificate revocation lists are only supported for file-based storage of certificates. They are taken into account automatically if they are present in the **crl subfolder** of the folder that is configured as trusted store or trusted issuer store.
- Missing certificate revocation lists do not lead to certificate validation failure.
- Certificate revocation list checks cannot be configured.

The PCo UA server currently does not support validation failure exceptions at all which leads, in general, to higher security but fewer configuration options. The PCo UA server only supports certificate revocation lists for file-based certificate storage. As for the UA source system, missing certificate revocation lists do not lead to certificate validation failures.

Providing the option to use session security is the task of the OPC UA server. The PCo OPC UA server functionality currently does not support user-based or certificate authentication for sessions; only anonymous access is supported.

The PCo UA source system supports user/password and certificate-based authentication if the UA server it connects to does.

### Application Certificate Management in Plant Connectivity OPC UA Components

OPC UA source system and server need an **X509 v3** certificate as application certificate for setting up a secure connection, and for this application certificate the private key has to be accessible to the OPC UA application. The PCo OPC UA applications will not start without a valid certificate. If an OPC UA application provides an application certificate, it has to be a formally valid certificate or empty (null), otherwise the connection is refused by the other party. No secure connection can be established with an empty certificate, of course.

**Note**

SAP Plant Connectivity does not provide a means to create **X509 v3** certificates for OPC UA application certificates. You can use self-signed certificates, certificates created by a UA server certificate creation utility from another vendor, or certificates from a certificate authority to create application certificates.

PCo UA applications support both certificate chains as well as self-signed certificates for the creation of secure channels. Since some UA applications may not be able to receive (partial) certificates, PCo UA applications allow
you to configure whether a certificate chain is sent or whether only the application certificate is sent. This is the default.

In order to send a certificate chain, you have to store the application certificate in the Microsoft Certificate store and the other (intermediate and CA) certificates of the chain in the folders your application uses as store for trusted issuers or the trust store. If you send only a partial chain, the missing parts have to be available to the other party by other means (usually in the trusted store or trusted issuer store). Collecting the certificates of a chain in order to send them to the other party relies on finding, recursively, the certificate that was used to sign those certificates in the chain that were already found. This is only possible if there is no gap in the chain as it is stored in the folders in which the application can search for them.

In Plant Connectivity the application certificate has to be stored in the Microsoft certificate store in order to protect the private key. The recommended way to provide it is to import it as a.pfx file into the personal folder of the LocalMachine store in the Microsoft certificate store. Other folders can be used to provide backward compatibility. If the Current user store is used, PCo agent instances, which run as a service, may not be able to access the private key, unless the service is run with the identity of the same user.

File-based certificate stores can only be used as a trusted store for application certificates from other UA applications (without their private key). This can be set up on the Security tab of UA source systems and UA servers. If you choose to change the default settings, you should follow the instructions in the ‘Prerequisites’ section in order to protect that location appropriately.

### Note

The OPC UA specification requires that certain attributes and extensions of application certificates are set to specific values. These requirements may differ from what you are used to from other certificate-based secure connections. They are detailed in section 6.2.2, Part 6 of revision 1.03 of the OPC UA specification.

### Note

Some UA applications may refuse to establish a secure connection if a self-signed certificate does not specify CertificateSigning as intended key usage. The specification only requests digitalSignature, nonRepudiation, keyEncipherment, and dataEncipherment.

### Note

The validation functionality for CA-signed certificates or chains of the Microsoft certificate store is, in general, different from the one used by OPC UA applications. If the Microsoft certificate store is able to validate a certificate signed by a CA, then this means that the CA certificate is stored in the Trusted Root Certification Authorities folder of the Microsoft certificate store. This is not normally the place where an OPC UA application searches for the CA certificate. These latter locations are configured on the Security tabs of the PCo UA application.

### SSL-Based Secure Communication in PCo

PCo uses Secure Sockets Layer (SSL) to secure the communication between PCo and the following:

- SAP MII, if PCo acts as MII client
- SAP ME, if PCo acts as Web client
- OData server, if PCo acts as OData client
- Web server for Restful Web services, if PCo acts as Web client
- ESP server, if PCo acts as ESP client
- SAP MII, if PCo acts as socket server
WebSocket clients, for example, SAP MII, if PCo acts as WebSocket server.

In addition, SSL is used in PCo services offered by the Management Host Service.

SSL is based on the exchange of certificates between server and client. During the handshake between server and client, the server identifies itself by a certificate that is sent to the client. The client checks whether the given certificate is a trusted and a valid certificate. Optionally, the server could request a client certificate. In this case, the client also has to send the certificate to the server that is checking the validity and the trustworthiness of the certificate. A secure communication channel is only created successfully, if all checks are passed.

In principle, there are two types of certificates:

- Self-signed certificates
- Certificates issued by a trusted Certification Authority (CA)

The type of certificate determines how to set-up the certificate infrastructure to get the secure communication working:

- If certificates issued by a trusted CA are used, it is not necessary that the client knows the server certificate and vice versa. It is only necessary that the client knows the CA certificate that issued the server certificate. Accordingly, the server has to know the CA certificate that issued the client certificate. This means that the server keeps the server certificate and the CA certificate in its certificate store, while the client stores the client certificate and the CA certificate.
- If self-signed certificates are used, the certificates have to be exchanged between server and client. This means that the server has to import the client certificate without a private key, while the client has to import the server certificate without a private key. Finally, the server certificate store contains the server certificate (with a private key) and the client certificate (without a private key). Accordingly, the client certificate store contains the client certificate (with a private key) and the server certificate (without a private key).

In order to grant a Secure Socket communication between MII and PCo, a server and a client authentication have been implemented. For more information about Configuring the Use of SSL on the AS Java, see help.sap.com/saphelp_nw73ehp1/helpdata/en/4a/015cc68d863132e10000000a421937/content.htm. The steps in PCo are described in the section Enabling SSL for an Agent Instance.

If PCo acts as an MII client or ME Web client, only a server authentication with certificates takes place. According to the description above, a server certificate has to be imported onto the MII server or the ME server. On the PCo side, the appropriate certificate has to be imported into the trusted certificate store.

In order to enable SSL for the Management Host service, the steps described in the corresponding section have to be considered.

Note

Self-signed certificates for PCo can be created with certificate creation tools like MakeCert or OpenSSL. (See PCo application help for Certificate handling [external document]).

In order to create a certificate request that could be sent to a CA, Microsoft IIS or certificate creation tools as described above could be used.

Enabling SSL for the Management Host Service

You can enable SSL for the Management Host service by editing the file ManagementHost.exe.config. Since the Management Host service is based on the Windows Communication Framework (WCF), potentially all options provided by the WCF apply. It cannot be guaranteed, however, that all possible and valid configurations will work in your environment. In particular, it is known that message security with wsHttpBinding for connections between WCF-based applications and Java applications is problematic or difficult to set up.

One particular example setup which enables transport security with the basicHttpbinding is described below:
To do this, carry out the following steps:

1. Open the file in the editor, go to the basicHttpbinding section and replace
   ```xml
   <security mode="TransportCredentialOnly">
   <transport clientCredentialType="Basic"/>
   with
   <security mode="Transport">
   <transport clientCredentialType="None"/>
   ```

2. You can then modify the corresponding endpoint in the Services section by replacing
   ```xml
   address="http://
   with
   address="https://
   ```

3. In the serviceBehaviors section, replace
   ```xml
   with
   <serviceMetadata httpsGetEnabled="True" httpsGetUrl="https://hostname:port"
   ```

4. Assign a certificate to the SSL port for the host name with IIS.

5. Open IIS Manager and select SSL Settings for Default Web Site:

   ![Figure 1](image)

   On the right side of the screen in the Actions menu, select Edit Site > Bindings.

   In the following pop up, you have to specify the IP address and the port of the Management Host for the type https. Additionally, you have to assign a valid certificate for the server as shown below:
Enabling SSL for an Agent Instance

1. Create a server certificate that is either self-signed or issued by a CA.
2. Import the server certificate to the personal certificate store.
3. Create a client certificate that is either self-signed or issued by a CA.
4. Import the client certificate or import the CA certificate to the trusted root CA store.
5. On the PCo Management Console, select an agent instance.
6. On the Query Ports tab, select the entry SAP MII.
7. Select Certificate as the authentication type.
8. Click Browse to select the appropriate certificate.

7.2 Network Security

If the business system and the data source that are to be connected by PCo are located in different network segments and/or separated by a firewall, the following considerations should be taken into account:

- The basic recommendation is to run PCo (or the agent instances, to be precise) close to the data source.
- For the notification scenario, PCo has to be able to reach the business system. The details for this are configured in the notification destination in the PCo Management Console.
- For the query scenario, the business system needs to be able to reach the agent instance. The port used for this can be configured in the agent configuration in the PCo Management Console.
- To use the remote console, the MMC (Microsoft Management Console) has to be able to reach the PCo Management Host. The port used for this can be configured in the PCo Management Console.
- To create a data source of type PCo Connector in MII, the system needs to be able to access the PCo Management Console.

Ports
All ports that an agent instance opens for incoming connections can be configured in the PCo Management Console. For this, navigate to the **Query Ports** tab in the agent instance configuration.

Using the PCo Management Console, you can also configure the port that is opened by the PCo Management Host. For this, navigate to **Tools ➤ Options ➤ Management Host ➤ Settings**.
Security for Configuration Files

PCo stores all configuration information for source systems, destination systems, and agent instances in configuration files. The location of these files depends on the operation system, but usually you can find the folder under:

C:/ProgramData/SAP/PCo

This folder also contains the configuration audit log with the change history. From the PCo Management Console you can reach this log using the View Menu.

Note
PCo does not store any runtime data and does not provide other security audit capabilities.

For the encryption of passwords in the configuration database files, PCo uses built-in Microsoft operating system support.
9 Dispensable Functions with Impacts on Security

The following functions that are relevant to security aspects can be switched off if you are not using them:

- **Management Host**: The Windows service for the PCo Management Host can be deactivated. The Management Host is required for the following scenarios:
  - Creation of a PCoConnector data server in MI1
  - Use of the Remote Client

- **Active Monitor**: If you are not using the Active Monitor scenario, you can deactivate the corresponding Windows service or you deselect the corresponding component already during installation.
# Typographic Conventions

<table>
<thead>
<tr>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;Example&gt;</code></td>
<td>Angle brackets indicate that you replace these words or characters with appropriate entries to make entries in the system, for example, “Enter your <code>User Name</code>”.</td>
</tr>
<tr>
<td>Example Example</td>
<td>Arrows separating the parts of a navigation path, for example, menu options</td>
</tr>
<tr>
<td>Example</td>
<td>Emphasized words or expressions</td>
</tr>
<tr>
<td><a href="http://www.sap.com">www.sap.com</a></td>
<td>Textual cross-references to an internet address</td>
</tr>
<tr>
<td><code>/example</code></td>
<td>Quicklinks added to the internet address of a homepage to enable quick access to specific content on the Web</td>
</tr>
<tr>
<td>123456</td>
<td>Hyperlink to an SAP Note, for example, SAP Note 123456</td>
</tr>
<tr>
<td>Example</td>
<td>• Words or characters quoted from the screen. These include field labels, screen titles, pushbutton labels, menu names, and menu options. • Cross-references to other documentation or published works</td>
</tr>
<tr>
<td>Example</td>
<td>• Output on the screen following a user action, for example, messages • Source code or syntax quoted directly from a program • File and directory names and their paths, names of variables and parameters, and names of installation, upgrade, and database tools</td>
</tr>
<tr>
<td>EXAMPLE</td>
<td>Technical names of system objects. These include report names, program names, transaction codes, database table names, and key concepts of a programming language when they are surrounded by body text, for example, <code>SELECT</code> and <code>INCLUDE</code></td>
</tr>
<tr>
<td>EXAMPLE</td>
<td>Keys on the keyboard</td>
</tr>
</tbody>
</table>