USER'S GUIDE

Central Governance

Version 1.1.0

1 March 2018

This documentation describes the following Axway software:

Central Governance 1.1.0

Contents

Preface

This guide describes the tasks for managing registered products such as Transfer CFT with Central Governance. This guide is the print version of the Central Governance online help and has the same content.

About Central Governance

Central Governance is the management platform for Transfer CFT and SecureTransport. It provides:

- A global data flow repository with end-to-end data flow definitions, from business applications and partners to the infrastructure level.
- Centralized supervision of data flows, consistent with definitions in the repository.
- Alert management to track problems linked to products or data flow processing, including a subscription mechanism for alert notifications.
- Standard web dashboards for a global view of data flow activity. You also can create custom dashboards.
- Automatic discovery of products to be managed.
- Centralized management of product configuration and associated deployment, including mass processing capabilities for highly distributed environments, which include groups and configuration policies.
- Centralized day-to-day operations management such as starting and stopping products and viewing their logs.

Who should use this guide

This guide is for people who administrate and use Central Governance to manage registered products. This guide presumes you have knowledge of:

- Your company’s business processes and practices
- Your company’s hardware, software and IT policies
- The Internet, including use of a browser

Others who may find parts of this guide useful include network or systems administrators, database administrators and other technical or business users.

Other documentation

Refer to the Help Center tab in the user interface for complete user documentation with information about configuring and managing Central Governance. Online help also is available throughout the UI.

Axway 5 Suite reference solutions

Central Governance, a Unified Flow Management (UFM) product, is a core part of Axway 5 Suite reference solutions that integrate selected Axway products to solve business issues. UFM governs data flows within your enterprise and externally with business partners. Reference solutions are:

- B2B Integration to exchange, transform, and process standardized electronic business documents within your B2B community.
- Managed File Transfer to securely transfer data in one-to-one, one-to-many, and many-to-many scenarios.
- Data Flow Integration to provide services for standardizing the exchange of business data with internal and external partners.
- Financial Integration to support data transfers in finance channels such as SWIFT and EBICS and transforms data into financial protocols.

Your organization might use Central Governance in the context of reference solutions. Find details about the product's role in documentation on the Axway Support website at support.axway.com.

Help troubleshooting

If you have problems viewing or navigating the help, accessed via help links throughout the user interface, refresh or reload the page. Or clear your browser's history or cache, restart the browser and try again.

Axway online

Go to Axway Support at support.axway.com to contact a representative, learn about training programs, or download software, documentation and knowledge-base articles. The website is for customers with active Axway support contracts. You need a user name and password to log on.

Accessibility

Axway strives to create accessible products and documentation for users. The following describes the accessibility features of Central Governance documentation.

Screen reader support

- Alternative text is provided for images whenever necessary.
- The PDF documents are tagged to provide a logical reading order.

Support for high contrast and accessible use of colors

- The documentation can be used in high-contrast mode.
- There is sufficient contrast between the text and the background color.
- The graphics have the right level of contrast and take into account the way color-blind people perceive colors.

What's new

The following features and enhancements are new for Central Governance 1.1.0.

Govern SecureTransport 5.3.1

Central Governance can manage Axway SecureTransport 5.3.1 in addition to Axway Transfer CFT 3.1.2 and 3.1.3.

Manage application-to-business flows

Central Governance can manage flows involving business partners in addition to application-to-application flows. Flows involving business partners can use these protocols: PeSIT, SFTP, HTTP, FTP.

Fine-grained access control

The role-based access control within the Central Governance Access and Security service enables defining access restrictions on object instances for products, applications and flows.

Make copies of flows

You can make copies of flows. The copies are the same as the originals, except the default names and descriptions identify the flows as copies. Using a copy as the starting point, you can keep or change the original configuration. Best practice is to add copies when you want multiple flows that differ only in details.

Enhance visibility

Central Governance out-of-the-box visibility encompasses application-to-business flows. In addition, actions on transfers can be performed from the Web Dashboard interface.

Enhance Transfer CFT governance

Central Governance can automatically apply a Policy upon Transfer CFT's registration; manage Transfer CFT's CRONJOBs, which enable Transfer CFT to execute scheduled jobs; manage file-based broadcast lists; and manage activation period per flow.

Support TLS 1.2

Central Governance supports TLS 1.2.

Secure database connection

Central Governance can use a secured JDBC connection to connect to an external Oracle, MySQL or SQL Server database.

Identity stores support STARTTLS

You can configure secure LDAP connections with STARTTLS for identity stores. STARTTLS is an extension to plain text communication protocols, which offers a way to upgrade a plain text connection to an encrypted TLS or SSL connection instead of using a separate port for encrypted communication.

Audit for dashboards

Actions performed in the dashboards user interface are stored in the Central Governance audit trail.

Getting started 1

This topic provides a workflow for new users of Central Governance to start using the product as a unified-flow management platform for supported Axway products. This is a high-level outline. Follow the references for more details about tasks.

The Central Governance user documentation assumes you have experience and operational knowledge of the products you register in Central Governance.

Getting started prerequisites

- Central Governance is installed and started. See the installation guide or *Configuration and startup* on page 47.
- You have logged on to the user interface. See *Logging on* on page 61.

Getting started tasks

1. Review the services and components that comprise Central Governance. See *Architecture* on page 25.
2. If you did not complete this task during initial configuration of Central Governance, determine whether you need to replace default certificates. Although you can use the default certificates, best practice is to replace the defaults with your own certificates. In any event, make sure you have resolved this before registering any product with Central Governance. See:
    - *CA services* on page 100
    - *Business certificate authority* on page 54
    - *Governance certificate authority* on page 54
3. Add users and assign them roles with appropriate privileges for using the Central Governance user interface or the UIs of registered Transfer CFTs or both. See *User management* on page 109. Note that Central Governance also supports LDAP connectivity. See *Identity stores* on page 134.
4. Become familiar with tasks for basic operations of Central Governance. See:
    - *Operations* on page 42
    - *Tools* on page 74
5. Register Axway products in Central Governance. Products supported for registration are:
    - Transfer CFT 3.1.2 and later
    - SecureTransport 5.3.1 and later

For details see *Product registration* on page 142 or the documentation of the product you want to register.

6. Learn about flows, the primary objects Central Governance manages. See:
    - *Concepts about Central Governance objects* on page 27
    - *General concepts about flows* on page 230
    - *Transfer CFT flow concepts* on page 256
    - *SecureTransport flow concepts* on page 250
7. Learn about using applications, partners and unmanaged products in flows. See:
    - *Applications* on page 213
    - *Partners* on page 219
    - *Unmanaged products* on page 226
8. Create and deploy flows. See:
    - *Defining flows* on page 272
    - *Save and deploy a flow* on page 293

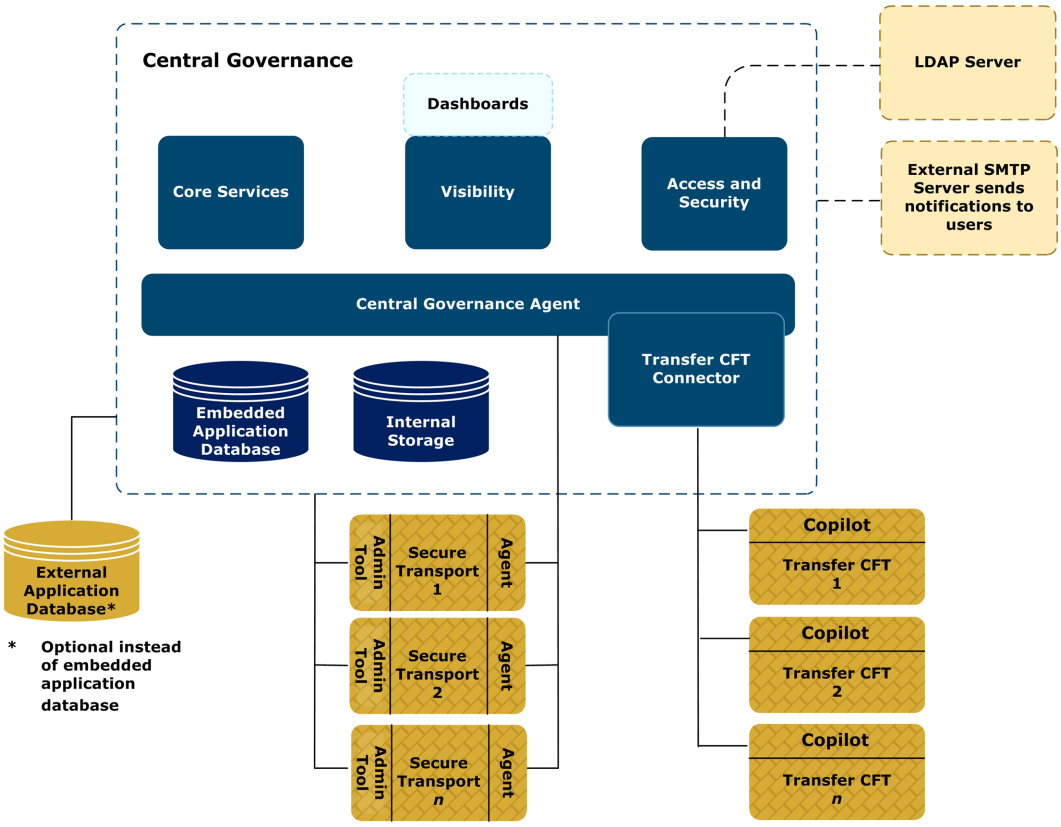
    If you have used Transfer CFT previously, also see *Transfer CFT legacy flows* on page 358
9. Activate default alert rules, modify alert rules or subscribe to alert rules to receive alerts via email. See *Alert rules* on page 405.
10. Use the view all flows report to monitor file transfers. See *Flow and transfer monitoring* on page 65.
11. Use dashboards to view graphical displays of file-transfer activity. See *Dashboards and reports* on page 66.
12. Make plans for regular purging and archiving of records in the database related to audit reports and monitoring of flow transfers to avoid disk space issues. See *Flow monitoring and audit data maintenance* on page 72.

More help for beginners

In addition to this getting started topic, there is a tutorial that walks beginners through the steps for using Central Governance to perform basic file transfers. The tutorial mostly uses default values and is designed to be completed within a short time. See *First file transfers with Transfer CFTs* on page 33 and *First file transfer with SecureTransport and Transfer CFT* on page 37.

Architecture

A single instance of Central Governance can be deployed on one computer per network. The system supports active-passive resiliency in a clustering environment to bring another instance of Central Governance online if the primary fails. The following illustrates the architecture.



The following provides a high-level description of the Central Governance nodes. A node represents processes that deliver one or more services.

Core services

Supports the Central Governance graphical user interface, identity management and management of all functions related to product configuration and flow definition.

Access and security

Provides a role-based access control model based on single sign-on. It manages permissions for users of Transfer CFT that are registered in Central Governance. The node also manages PKI and certificates. Optionally, it can integrate with an LDAP server for externally managed users and their credentials.

Visibility

Tracks transfers and manages notifications and alerts. It has an events monitoring table for users. It also can connect to an external SMTP server and send alert email messages to users specified within Central Governance. The messages provide links for accessing the Central Governance user interface for more information.

The Visibility service is powered by Axway Sentinel, a data-collection and reporting product for monitoring file transfers in unified-flow management solutions.

Agent

Supervises all Central Governance nodes. The main entry point of Central Governance, it starts, stops, and monitors nodes. The agent also deploys configurations and applies updates.

If a node fails and the status changes to crashed, the Agent will try to restart it. If the node cannot be restarted, the Agent stops all the other nodes, and the status of Central Governance changes to crashed.

In addition there is the Transfer CFT connector, a plugin used as a proxy to connect to local or remote Transfer CFTs. Central Governance can communicate with few or many Transfer CFTs.

This version of Central Governance is compatible with Axway Transfer CFT 3.1.2 and 3.1.3 and Axway SecureTransport 5.3.1 and later.

Central Governance has two embedded databases:

- Internal storage is a NoSQL MongoDB database for Central Governance configuration data, including policies, flows and partner definitions.
- Application database is a MySQL database for transfer tracking data, user roles and privileges, certificates and dashboards. Central Governance can be configured to use an external database instead for this purpose.

Concepts about Central Governance objects

Central Governance has many objects with different purposes in managed file transfer (MFT). This topic describes the objects and their roles.

Objects you can use in flows

Objects you can use in flows for transferring files are:

- Registered products
- Unmanaged products
- Applications and application groups
- Partners

Roles in flows

The following table shows the roles of these objects in flows.

Role	Product	Unmanaged product	Application	Application group	Partner
Source	no (note)	yes	yes	yes	yes
Target	no (note)	yes	yes	yes	yes
Relay	yes	yes	no	no	no

Note Registered products must be associated with applications to be used as sources or targets in flows. Products, on their own, cannot be used as sources or targets.

When to use in flows

The following table shows when to use the objects in file transfers.

Legend

A2A - Application to application transfer
A2B - Application to business (partner) transfer
B2A - Business to application transfer
B2B - Business to business transfer

Object	Use in transfers
Product	A2A, A2B, B2A (note 1)
Unmanaged product	A2A, A2B, B2A
Application	A2A, A2B, B2A
Application group	(note 2)
Partner	A2B, B2A, B2B

Note 1. Products are associated with applications.

Note 2. Application groups typically are used for filtering a set of applications with a common business characteristic, managing FGAC, or defining a flow with a large set of applications. See *Applications and application groups* on page 30.

As the previous table infers, applications are associated with business applications within organizations and partners are associated with external organizations with whom you have business relationships. Central Governance is agnostic over the type of systems partners use to communicate with applications.

Descriptions of objects in flows

The following summarizes each object you can use in flows.

Products

Products are instances of Axway products that can register with Central Governance and become governed by it. You can use Central Governance to:

- Start and stop products
- View and change the configurations of products (except SecureTransport is view only)
- View logs of products (except SecureTransport)
- Associate products with applications and application groups for use as sources or targets in flows
- Use products as relays between sources and targets in flows
- Use products in a client or server role when communicating via any of multiple supported protocols
- Apply service packs and patches to products (except SecureTransport)

You also can assign products to groups. Grouping products enables you to deploy configurations and perform operations all together. For example, perform operations such as starting or stopping from the command line interface by specifying the group of products.

For Transfer CFTs you can create policies. A policy represents common configuration settings for multiple Transfer CFTs. You can simultaneously deploy the same configuration changes to all Transfer CFTs assigned to a policy. For example, if multiple Transfer CFTs use parallel TCP (pTCP), you can create a policy with this configuration and deploy it to the Transfer CFTs.

Related topics

Product registration on page 142

Defining flows on page 272

Unmanaged products

Unmanaged products are systems that are not registered in Central Governance, but that are integrated in flows for transferring files. Unmanaged products can be:

- Transfer CFTs 3.1.2 or later that are not registered with Central Governance
- SecureTransports 5.3.1 or later that are not registered with Central Governance
- Earlier versions of Transfer CFT or SecureTransport that cannot register with Central Governance
- Axway products other than Transfer CFT or SecureTransport
- Third-party products

Unlike registered products, Central Governance cannot detect, start, stop or change the configurations of unmanaged products. However, the Central Governance user interface provides a way to define unmanaged products and include them in flows.

Unmanaged products:

- Support only the PeSIT protocol
- Can be used as sources, targets or relays in flows
- Can be used in a client or server role in communications

You can use multiple unmanaged products as sources or targets in flows, but only a single unmanaged product can be used as a relay, although there can be multiple relays.

Central Governance cannot manage the configurations of unmanaged products or apply service packs or patches on them. It also cannot deploy flows to unmanaged products. You must deploy flows on them manually.

Related topics

Unmanaged products on page 226

Defining flows on page 272

Applications and application groups

An application is the logical representation of a business software application that is the true sender or true receiver in a file exchange. An application represents a back-end enterprise resource planning system such as SAP or PeopleSoft.

All applications are associated with products. The products perform the actual communication between applications and other systems.

An application can be associated with one or multiple products of the same or different type. One or multiple applications can be used as the source or target in a flow. An application, by virtue of the associated products, also can be used in a client or server role in communications.

Closely related are application groups. These are logical sets of applications that can be used in flows as sources or targets. Application groups also can be used in client or server roles in communications.

When to use applications or application groups:

- Use applications, singly or multiple, when they represent clearly stable applications in flows that send the same type of information to accounting or financial applications.
- Use application groups when you want to add applications to flows while making no or few other changes to the flows. In short, you expect the number of participating applications to grow.

Related topics

Applications on page 213

Application groups on page 215

Partners

Partners represent entities such as companies that send or receive business data in file transfers governed by Central Governance flows. Partners can use third-party products or Axway products not registered in Central Governance to communicate with other parties over supported protocols.

Partners can be sources or targets in Central Governance flows. They also can be in client or server roles, depending on whether the partners initiate transfers. Partners support multiple types of communication protocols.

Only server communication profiles are managed in partner objects. Client communication profiles for partners are managed in flows.

Related topics

Partners on page 219

Communication profiles on page 236

Who manages objects in flows

Some objects refer to a different abstraction level or perspective (for example, business versus technical).

- Partners, applications and application groups are about *what* participants communicate in a flow. Typically, business users manage these objects.
- Unmanaged products and products are about *how* participants communicate in a flow or mediation. Typically, technical users manage these objects.

Client and server communication profiles

Communication profiles define a a capability or capacity for a client or server to communicate with a sender or receiver. Communication profiles have properties and configurations for protocols (HTTP, SFTP, FTP, PeSIT). You use a client communication profile when the owner is the initiator of the communication. Otherwise, you use a server communication profile.

Client and server communication profiles can be shared between senders and receivers.

With Transfer CFT and unmanaged products, client-server communication are not represented in flows. They are implicitly and automatically used depending on protocol link properties (direction, authentication level, network protocol, acknowledgment).

Client

A client communication profile defines communication capability when the owner is the initiator, or requester, of the communication. You define the client authentication (user name and password, key or certificate). For SecureTransport you also define the network zone to use.

Server

A server communication profile defines communication capability when the owner is a receiver of the initialized communication.

A server communication profile defines the server connection (host, port or URL) and the server authentication. It also might specify requirements for SSL/TLS, FIPS and client authentication level.

PeSIT also requires you to define a network protocol to be used (TCP, UDT, pTCP) and is the only protocol where a server has a login/password for authentication.

For SecureTransport, which has the concept of network zone, a server communication profile can be attached directly to its server, via a private network zone, or to a reverse proxy (edges) in the DMZ.

Related topics

Communication profiles on page 236

Protocols in flows

You define protocols between segments in flows: between source and target, source and relay, relay and relay, and relay and target. Protocols are the communication medium between middleware initiators and receivers in file transfers. In a many-to-one or one-to-many flow, the protocol is a set of protocol links or exchanges. In many-to-many flows from applications to applications, the protocol is all combinations of source-target middleware.

Protocol details

A protocol in a flow specifies:

- The protocol (HTTP, SFTP, FTP, PeSIT).
- The direction of the communication and initiator (client or server).
- Security, such as SSL/TLS server authentication, mutual authentication, password or key.
- The client communication profile used by the initiator.
- The server communication profile used by the receiver.
- Whether acknowledgments are sent (for example, PeSIT).

The Central Governance user interface enforces compatibility of the client and server communication profiles.

Protocol direction

A file is transferred from the source to the target along the configured mediation route, and acknowledgments from receivers to senders go in the opposite direction. However, for each protocol in the flow you can select the initiator in the flow segment.

Sender pushes files

The sender is the client initiator. You must use a client communication profile on the sender and a server communication profile on the receiver.

Receiver pulls files

The receiver is the client initiator. You must use a server communication profile on the sender and a client communication profile on the receiver.

Related topics

General concepts about flows on page 230

Specify the protocol on page 277

Legacy flows for Transfer CFTs

Legacy flows for Transfer CFTs are a feature for enabling long-time users of Transfer CFT to transition flow management to Central Governance. Legacy flows address the following use cases:

- Flows can be managed in Central Governance for individual Transfer CFTs. In the Central Governance user interface, users can manage partners and send and receive templates for a specific Transfer CFT.
- Users can employ an established procedure to migrate Transfer CFT flow definitions to the Central Governance flow-management process.

Related topics

Transfer CFT legacy flows on page 358

First file transfers with Transfer CFTs

Use the following procedures in sequence to configure two Transfer CFTs in Central Governance for a file transfer. The procedures are intended to help new users get started using Central Governance as the managing agent of multiple Transfer CFTs.

The procedures describe simple transfers, with and without security, and mostly using default settings in Central Governance for flows. There are steps for sending a file from one Transfer CFT to another and monitoring the transfer in Central Governance.

For more help see *Getting started* on page 24 for a workflow for using Central Governance to manage registered products.

Prerequisites

- Central Governance is installed and running.
- At least two supported versions of Transfer CFT are installed and have registered successfully with Central Governance. For details, see *Transfer CFT registration* on page 142 or Transfer CFT user documentation.
- Other than installing and registering, you do not have to perform any configuration tasks on the Transfer CFTs.
- You are logged on to the Central Governance user interface with a user with permissions to check product status and configure applications and flows. Minimally, the user is assigned to the default Middleware Manager role or a user-defined role with similar permissions.
- The Transfer CFTs to use in the transfers are started. To verify, select **Products** on the top toolbar in the Central Governance UI and check the status of the Transfer CFTs on the Product List page.
- You have access to CFTUTIL commands on the participating Transfer CFTs.

Identify products

You need the names (for the SEND and RECV commands) and the hostnames (to create the applications). Click **Products** on the top toolbar to open the Product List page. Copy or write the host names of the two Transfer CFTs you want to exchange files.

Add applications

An application must be associated with a registered product for the product to be used as a source or target in a flow. In these steps you add two applications. Each is associated with one Transfer CFT.

1. Click **Applications** on the top toolbar to open the Application List page.
2. Click **Add application**.
3. Type a unique name for the application.
4. Paste or type the host name for the sending or receiving Transfer CFT.
5. Click anywhere on the page. Because the host name is for a registered Transfer CFT, Central Governance populates the Products field with the Transfer CFT name.
6. You can ignore the other fields.
7. Click **Save application**.
8. Repeat the steps to add an application associated with the second Transfer CFT.

Add a flow

Do the following to add a flow that contains the source and target applications. In this flow, the source connects to the target and transfers files to it.

1. Click **Flows** on the top toolbar to open the Flow List page.
2. Click **Add flow**.
3. Type a friendly name for the flow. For example, the daily sales data for stores in the western region might be named **West Daily Sales**.

   You can ignore the details and contact fields.
4. Click **Source** to select the flow source. With the source type set to **Applications** and the product type set to **Transfer CFT**, click **Add source**. Select the application you want as the sender of files and click **Select as source**.
5. Click **Target** to select the flow target. With the target type set to **Applications** and the product type set to **Transfer CFT**, click **Add target**. Select the application you want as the receiver of files and click **Select as target**.
6. Ignore the source and target sections for transfer properties, files properties and processing scripts. This flow uses the default settings for those.
7. Click **Protocol** between the source and target. Use **PeSIT** as the exchange protocol and **Sender pushes file** as the direction.

   Type the flow identifier, an IDF in Transfer CFT. When the flow is deployed, this value is deployed to the participating Transfer CFTs. The flow identifier for the daily sales data from the western region might be named **WR01**.

   Use the default settings for network protocol, SSL/TLS, acknowledgment and PeSIT properties.
8. Click **Save**.

Deploy the flow

Do one of the following to deploy the flow on the source and target Transfer CFTs:

- If on the Flow List page, select the flow and click **Deploy**.
- If on the flow details page, click **Deploy**.

You can verify the deployment by doing one of the following:

- On the Flow List page, click the name of a flow to open its details page. Click the **Deployed at** link under the flow name at the top of the page to open the Flows section of the Deployment List page.
- Select **Administration > Deployments** and click **Flows** on the left side of the Deployment List page.

Add a test file

Set up a file to use in a transfer. For example, you can use a text file named `test.txt` that contains any text. Put the file in a directory the sending Transfer CFT can access. For example, on Windows the directory can be `C:\test`.

Send a file

Do the following to transfer the test file from Transfer CFT in the sending application to Transfer CFT in the receiving application.

Access the CFTUTIL commands on the participating Transfer CFTand execute a command in the following format:

```
CFTUTIL SEND IDF=<flow identifier>, PART=<name of the receiving Transfer
CFT>, FNAME=<path and file name of the file to send>
```

Monitor the transfer

Select **Flows > Flows Report** on the top toolbar to monitor the status of the transfer.

Change direction of the flow and transfer another file

Do the following to change the direction of the flow and transfer another file. Previously, the sender pushed the file. With this change, the target pulls the file.

Change the direction of the flow

1. Click **Flows** to open the Flow List page.
2. Click the name of the flow you added earlier to open its details page.
3. Click **Edit** to change the flow.
4. Click the protocol between the source and target. Set the direction as **Receiver pulls file**.
5. In the file properties for the source, enter the name of the file on the source Transfer CFT to transfer. For example, on Windows `C:\test\test.txt`.
6. Click **Save** and **Deploy**.

Transfer the file

Access the CFTUTIL commands on the participating Transfer CFTand execute a command in the following format:

```
CFTUTIL RECV IDF=<flow identifier>, PART=<name of the source Transfer CFT>
```

Execute a secure file transfer

The previous file transfers were without security. Use this procedure to transfer a file with security.

1. Change the configuration of the two Transfer CFTs to add a PeSIT protocol with security. Do the following for each Transfer CFT.
    a. Click **Products** on the top toolbar to open the Product List page.
    b. Click the name of a product to open its details page.
    c. Click **Configuration** on the right side of the page.
    d. Click **Edit**.
    e. In the Protocols section of the page, click **Add protocol**.
    f. Make sure **SSL_DEFAULT** is selected in the drop-down list.
    g. Enter **1762** as the port in.
    h. Click **Save** and **Deploy** to push the changed configuration to Transfer CFT. When prompted click **Deploy configuration** to deploy now and restart Transfer CFT.
    i. Repeat these steps for the other Transfer CFT.
2. Add a flow as before in *Add a flow* on page 35 using the same applications. However, for the PeSIT protocol select **Mutual authentication** in the SSL/TLS field. Save and deploy the flow to the Transfer CFTs.
3. See *Send a file* on page 36 for the procedure to transfer a file.

You can monitor the file transfer as before.

First file transfer with SecureTransport and Transfer CFT

Use the following procedures in sequence to configure a Transfer CFT and a SecureTransport in Central Governance for a file transfer. The procedures are intended to help new users get started using Central Governance as governance solution for Transfer CFT and SecureTransport.

The procedures describe simple transfers, mostly using default settings in Central Governance for flows. There are steps for sending a file from a Transfer CFT to a SecureTransport, making the file available via SFTP to a business partner and monitoring the transfers in Central Governance.

For more help see *Getting started* on page 24 for a workflow for using Central Governance to manage registered products.

Prerequisites

- Central Governance is installed and running.
- At least one Transfer CFT and one SecureTransportare installed and have registered successfully with Central Governance. For details, see *Product registration* on page 142 or Transfer CFT user documentation.
- Other than installing and registering, you do not have to perform any configuration tasks on the Transfer CFTs. On SecureTransport, make sure that SFTP is enabled before registration.
- You are logged on to the Central Governance user interface with a user with permissions to check product status and configure applications, partners and flows. Minimally, the user is assigned to the default Middleware Manager role or a user-defined role with similar permissions.
- The Transfer CFT and SecureTransport to use in the transfers are started. To verify, select **Products** on the top toolbar in the Central Governance UI and check the status of the Transfer CFT and SecureTransport on the Product List page.
- You have access to CFTUTIL commands on the participating Transfer CFT.

Identify products

Click **Products** on the top toolbar to open the Product List page. Copy or write the host name of the Transfer CFT and the name of the SecureTransport you want to exchange files.

Add an application

An application must be associated with a registered product for the product to be used as a source or target in a flow.

1. Click **Applications** on the top toolbar to open the Application List page.
2. Click **Add application**.
3. Type a unique name for the application.
4. Paste or type the host name of the sending Transfer CFT.
5. Click anywhere on the page. Because the host name is for a registered Transfer CFT, Central Governance populates the Products field with the Transfer CFT name.
6. You can ignore the other fields.
7. Click **Save application**.

Add a partner

Do the following to add a business partner.

1. Click **Partners** on the top toolbar to open the Flow List page.
2. Click **Add partner**.
3. Type a unique name for the partner.
4. You can ignore the other fields.
5. Click **Save**.

Add a flow

Do the following to add a flow that contains the source application and the target partner. In this flow, the source application sends files to SecureTransport via its Transfer CFT. SecureTransport then makes the file available to the target partner.

1. Click **Flows** on the top toolbar to open the Flow List page.
2. Click **Add flow**.
3. Type a friendly name for the flow. For example, the invoices to be sent to customers might be named **Customer Invoices**.

   You can ignore the details and contact fields.
4. Click **Source** to select the flow source. With the source type set to **Applications** and the product type set to **Transfer CFT**, click **Add source**. Select the application you want as the sender of files and click **Select as source**.
5. Click **Target** to select the flow target. With the target type set to **Partners**, click **Add target**. Select the partner you want as the receiver of files and click **Select as target**.
6. Click **+ Relay** to add a relay to the flow.
7. Click **Relay** to select the product used as relay. Click **Edit relay**, select the SecureTransport you want to use as relay and click **Select as relay.**
8. Click **Protocol** between the source and relay. Use **PeSIT** as the exchange protocol and **Sender pushes file** as the direction.

   Type the flow identifier, an IDF in Transfer CFT. The flow identifier for the customer invoices might be named **CI01**.

   Use the default settings for network protocol, SSL/TLS, acknowledgment and PeSIT properties.
9. Click **Protocol** between the relay and target. Use **SFTP** as the exchange protocol and **Receiver pulls file** as the direction.

   Use the default settings for client authentication, FIPS transfer mode and SFTP properties.
10. For the **Client communication profile**, click **Create new one**. Type a friendly name for the Client communication profile **name** and type the desired **login** and **password** the partner will use to connect to SecureTransport.
11. Ignore the source sections for transfer properties, files properties and processing scripts. This flow uses the default settings for those.
12. Click **Send properties** on the relay. In **Directory**, type the directory in which the files will be

made available to the partner. For example, type **/invoices**.

13. Click **Save**.

Deploy the flow

Do one of the following to deploy the flow on the participating Transfer CFT and SecureTransport:

- If on the Flow List page, select the flow and click **Deploy**.
- If on the flow details page, click **Deploy**.

You can verify the deployment by doing one of the following:

- On the Flow List page, click the name of a flow to open its details page. Click the **Deployed at** link under the flow name at the top of the page to open the Flows section of the Deployment List page.
- Select **Administration > Deployments** and click **Flows** on the left side of the Deployment List page.

Add a test file

Set up a file to use in a transfer. For example, you can use a text file named `test.txt` that contains any text. Put the file in a directory the sending Transfer CFT can access. For example, on Windows the directory can be `C:\test`.

Send a file

Do the following to transfer the test file from Transfer CFT in the sending application to SecureTransport.

Access the CFTUTIL commands on the participating Transfer CFTand execute a command in the following format:

```
CFTUTIL SEND IDF=<flow identifier>, PART=<name of the receiving
SecureTransport>, FNAME=<path and file name of the file to send>
```

The file is now made available by SecureTransport to the partner. In order to retrieve the file, the partner must:

1. Connect to SecureTransport with an SFTP client using the credentials defined in the flow.
2. Go to the **/invoices** folder.
3. Download the file.

Monitor the transfer

You can monitor the status of the transfers in Central Governance Visibility user interface.

Select **Flows > Flows Report** on the top toolbar to open the Visibility UI.

Operations 2

The following topics relate to Central Governance operations and processes.

Services

Central Governance provides services for governing the product. The services and their statuses are displayed on the Central Governance Services page. Click **Administration** on the top toolbar to open the page.

Descriptions

The services – also called nodes – are:

- Core services – Manage basic platform functions.
- Access and Security – Manage certificates for securing file transfers.
- Visibility – Monitor file transfers. The Visibility service is powered by Axway Sentinel, a data-collection and reporting product for monitoring file transfers in unified-flow management solutions.
- Transfer CFT connector – Communications services for Transfer CFT.
- Application database – Data storage for access and security and visibility services. This service is listed only if the embedded application database is in use. If an external database is being used instead, this service is not displayed on the page.
- Internal storage – Data storage for core services.

Services page

On the Services page you can:

- View status to determine if all services are running properly.
- Start a stopped service.
- View service logs to determine the cause of problems.

Status of Central Governance and services

You can view the status of Central Governance and its services on the

Central Governance Services page. Click **Administration** on the top toolbar to open the page.

Service statuses

Each service – also called a node – displays one of the following statuses.

Started in error

A service is in abnormal state because it failed to start or stop. The status is always associated with an error message.

Stopped in error

A service is in abnormal state because it failed to initialize or it crashed. The status is always associated with an error message.

Started

A service has started.

Starting

From the time a start command occurs to the time the service returns a status or until a timeout.

Stopped

A service has stopped.

Stopping

From the time a stop command occurs to the time the service returns a status or until a timeout.

Unreachable

Central Governance cannot get the status for a service. This can occur if network issues prevent communication. This status is always associated with an error message.

You can start a stopped service in the user interface. Other operations are available using commands outside of the UI. See *cgcmd command* on page 74 for basic administrative functions and *Command line interface* on page 78 for more advanced options.

Operation	Conditions
Start	Perform on a stopped service.
Stop	Perform on a started or in error service.
Force Stop	Perform on an in error service.
Restart	Perform on a started or in error service.

Central Governance statuses

The status of Central Governance as a whole is determined by the statuses of the various services.

In error

At least one service – also called a node – is started or stopped in error.

Unreachable

At least one service has an unreachable status.

In progress

At least one service is starting or stopping.

Partially started

At least one service is stopped and one is started.

Started

All services are started.

Stopped

All services are stopped. This status is not visible in the user interface.

Crashed

All services are stopped due to a recovery problem. This status is not visible in the user interface.

Crashed indicates Central Governance was stopped because the Agent was unable to recover a crashed node and reacted by stopping all nodes. This differs from the stopped status, which indicates Central Governance was stopped normally.

Unavailable

One or more services cannot start. Restart Central Governance to resolve. This status is not visible if you cannot connect to the user interface.

Service actions

The actions available for Central Governance are:

- If at least one service is starting or stopping, no actions are available.
- If at least one service is stopped, the **Start All** action is available. The action applies only to stopped services.

Visibility service starts, stops in error

If the Visibility service reports a status of *started in error* or *stopped in error*, this is likely the result of the operation timing out before the service can start or stop correctly. For example, stopped in error can occur when a large amount of data was in process when the service was being stopped.

You can prevent these errors by adding a properties file and setting values for properties within it.

1. Create a file named `lifecycle.conf`.
2. Add the following properties to the file:

   ```
   start.timeout=
   ```

   ```
   stop.timeout=
   ```

   These properties require values in milliseconds. To determine values, consider how long it usually takes the Visibility service to start or stop after running `cgcmd start` or `cgcmd stop`. For example, if it takes about two minutes for the service to stop, use `stop.timeout=120000`. If stopped in error status continues to occur, increase the value until the service stops normally.

   The minimum valid value is 500 for both properties, but there is no reason to set either property this low.
3. Navigate to:

   ```
   <install directory>\runtime\com.axway.nodes.sentinel_<UUID>\uma
   ```
4. Add a subdirectory under `uma` named `capabilities`.
5. Copy the `lifecycle.conf` file to:

   ```
   <install directory>\runtime\com.axway.nodes.sentinel_
   <UUID>\uma\capabilities
   ```
6. Restart Central Governance for the new properties to become effective.

Central Governance services logs

Use the Central Governance services logs to monitor usage or diagnose problems. There is one summary log for all core services and one log for each service.

View log

1. Select the **Administration** tab to view the list of Central Governance services.
2. Locate the service whose log you want to view and click **View log**.

The log page is displayed where you can:

- Click **Refresh** anytime to update the log entries.
- Sort the entries by newest or oldest.
- Filter the entries, saving filters for future use.
- Use the Log drop-down list to view the log of a selected service.

The following describes all log table columns. The columns apply to the service logs as noted.

Date/time

The server date and time of the log entry. Format: YYYY-MM-DD hh:mm:ss.

Applies to logs for all services.

Service

Identifies the internal service associated with the log entry.

Applies to log for Core Services.

Level

Level of the log entry. Levels, from highest to lowest verbosity, are DEBUG, INFO, WARNING, ERROR.

Applies to logs for Core Services, Access and Security, Visibility and Transfer CFT connector.

Message

Actual log message.

Applies to logs for all services.

Filter log

You can filter a log by one or multiple conditions. The filters you add are saved until deleted or the browser cache is cleared. You can, for example, filter by level, leave the page and return, and the displayed log entries are filtered by level.

Click **Filter** and select a filter type to add. You can add multiple filters.

Not all filter types are available for all logs. For example, the service filter is available only for the Core Services log.

Date/time

You can filter log entries by age in hours or generated within a date range.

Service

You can filter log entries by full or partial service names. This filtering is not case sensitive. Only one filter can be set for the service column.

Level

You can filter log entries by severity. This filtering provides cumulative verbosity. If you filter by Info level, Info messages and all message levels above the Info level are displayed. If you filter by Fatal level, only fatal log entries are displayed.

Message

You can filter log entries by full or partial messages. This filtering is not case sensitive. You can filter by one or more messages.

Added filters are displayed at the top of the page. Click a filter to change it. Click the appropriate X icon to delete a single filter or to clear all filters.

Configuration and startup

You can change the configuration of Central Governance that was set initially after installation. You can change any settings except:

- You cannot change the application database type (internal or external). If Central Governance has been installed with the embedded database, it is not possible to switch to an external one. Likewise, if Central Governance has been installed with an external database, it is not possible to activate the embedded one and switch to it afterwards.
- If the database type is external:
  - You cannot change from one database to another. For example, you cannot change from Oracle to MySQL.
  - You cannot use a fresh database. You must use the same database or an exact duplicate of the old one. For example, if the database is MySQL, do nothing to keep using it. Otherwise, you must dump the database and install the dump on a new MySQL database. The new database must have the same tables and data as the old database. You must ready the duplicate database before starting the configuration process.

Only change the configuration using the configuration user interface. Do not instead change property values in configuration files.

Although except for the database exclusion you can change values of any fields, additional tasks are required when values of some fields are changed. See *Editing some fields requires follow-up actions* on page 48.

Start configuration web server

You must stop Central Governance and then run the `cgcmd configure` command from the installation directory to start an internal web server that hosts the configuration user interface.

Once the web server has started, the command lists the URL for opening the web page in a browser. If the computer on which the command was executed has a default browser, the page opens automatically. Otherwise, open the page with the provided URL.

By default the web server runs on port 8082. But you can change the port when invoking the command. For example:

```
cgcmd configure -p <port>
```

See *cgcmd command* on page 74 for more information about the `cgcmd` command.

Editing some fields requires follow-up actions

After using the configuration web page the first time to initially configure Central Governance, you can use the page again to edit field values. However, changing some fields requires performing additional tasks to make sure Central Governance continues running properly.

General > FQDN

Changing the FQDN field also requires you to update all registered products manually with the new host value. If you do not, Central Governance cannot reach the registered products. The status of registered products becomes unreachable.

Access and Security > HTTPS client authentication port

Changing the HTTPS client authentication port field also requires you to redeploy the configurations of all registered Transfer CFTs after restarting Central Governance. Redeploying makes the port change effective on the Transfer CFTs. Only Transfer CFTs are affected and not any other types of registered products.

Access and Security > Shared secret

Changing the shared secret in Central Governance requires also changing the shared secret used by registered products. The shared secret is used to establish connections between Central Governance and registered products.

Change the shared secret in Central Governance and SecureTransports at the same time. To change:

1. Stop Central Governance, start the configuration web server and change the shared secret on the configuration page.
2. Stop the Central Governance agent in SecureTransport and change the shared secret on its Central Governance configuration page.
3. Restart Central Governance.
4. Restart the SecureTransport Central Governance agent.

For Transfer CFT you don't have to change the shared secret immediately. Transfer CFT communicates with Central Governance differently than SecureTransport. If the shared secret is changed in Central Governance, Transfer CFT Copilot can still connect to Central Governance without changing the shared secret. However, if a change is made in Central Governance that affects its communication with Transfer CFT, such as changing a CA, the shared secret in Transfer CFT must be current. See *If CAs change after Transfer CFT registration* on page 101 for more information.

To change the shared secret in Transfer CFT, stop the product, run the installer in configure mode and change the shared secret.

Access and Security > Business certificate authority

Changing the business CA also requires you to make changes in the registered Transfer CFTs. See *Change CAs after Transfer CFT registration* on page 147 for more information.

Access and Security > Governance certificate authority

Changing the governance CA also requires you to make changes in the registered Transfer CFTs. See *Change CAs after Transfer CFT registration* on page 147 for more information.

Visibility > Front-end port

Changing the Front-end port field also requires you to redeploy the configurations of all registered Transfer CFTs after restarting Central Governance. Redeploying makes the port change effective on the Transfer CFTs. In the case of registered SecureTransports, you must change the front-end port in the SecureTransport administration user interface.

Transfer CFT connector > Secured communication port

Changing the Secured communication port field also requires you to update all registered Transfer CFTs manually with the new port value. If you do not, Central Governance cannot reach the registered products. The status of registered products becomes unreachable.

Secure external database connections

If you choose to use an external database, instead of the default embedded database, to store data in the application database, you can have secure JDBC connections for one or more of the following:

- Access and Security service
- Visibility service
- Dashboards service

This option is available for all supported database types.

To have secure connections you must:

1. Obtain valid server certificates and configure your database system to use them.
2. Select **Use secured JDBC connection** for a service on the Central Governance configuration page.
3. Click **Browse** and select the public certificate file to upload for the secure connection. This file contains the CA or trust chain for the SSL certificate used by the external database server. The imported file must contain only one certificate. Supported keystore formats are PKCS#12 and Java KeyStore (JKS).
4. Enter a password for the certificate. This is required to enhance security even though the uploaded file does not contain a private key.

Complete configuration

Change fields as you require on the configuration page.

When appropriate, the user interface provides default values in the fields and as tooltips.

Many of the fields are for port values. See *Default ports and firewall requirements* on page 58 for the list. One reason to use your own rather than a default value is port conflicts. A default port assignment could conflict with a port used by another application or process on your system. Ports already in use are detected when you submit the configuration page, which enables you to select other values. However, you also can use a command to discover and resolve port conflicts. See *Resolving port conflicts* on page 60.

In addition, when firewalls are present, some ports must be opened to enable communications with remote systems.

General

FQDN

The name used by systems outside your network to connect to Central Governance. This can be a fully qualified domain name (FQDN), IPv4 address or a load-balancer URL. FQDN example: `myhost.domain.com`.

You can use an IP address in this field only when it can be resolved to a valid FQDN.

Host name

The host name for Central Governance. This can be the host name only or the same as the FQDN field value. This also can be a virtual name for running Central Governance in an active-passive cluster.

This is not necessarily the machine where Central Governance is installed, but the machine where it will run. Technically, the name refers to the network card where Central Governance will bind the sockets for all the ports in use.

UI port

The SSO port for connecting in a browser to the Central Governance user interface.

License

Click **Browse** and select the Central Governance license file in the file system. You must have a valid license file to run Central Governance. See *License file* on page 64 for more information.

Log level

The level of events written to log files for Central Governance and its services. The log levels, from lowest to highest verbosity, are:

- Error
- Warning
- Info
- Debug
- All

Selecting the highest verbosity level might slow the performance of Central Governance.

SMTP server

Central Governance requires a connection to an external SMTP server to send notifications and alert messages to its users. You might have to consult with your network administrator to configure this.

SMTP server host

The name of the SMTP server for outbound email messages.

SMTP server port

The port for outbound messages typically is **25**. Outbound messages include alerts and notifications to users of Central Governance.

Authentication

Requiring authentication for outbound messages is uncommon. If your server requires authentication, click **Yes** and complete the user name and password fields.

Agent

Central Governance name

Unique name of the Central Governance agent. This name is used to identify this instance of Central Governance in communications with other instances of Central Governance and with registered products.

Port

Agent cluster infrastructure listening port. External agents use this port to register products with Central Governance and communicate with it.

Core services

HTTPS port

User interface (non-SSO) HTTP over SSL. This is the internal port on which the SSO server connects to Central Governance core services.

Application database

Type

Indicates whether to use the embedded internal database or an external database for storing application data. You can use the embedded database only if your user license allows. If not, you only can use an external database. Note that you cannot change the type once Central Governance has been fully installed. So it is not possible to switch from an embedded database to an external one afterwards, and vice-versa.

Note The embedded MySQL application database is for use by Central Governance only. Do not try to use this database with any other Axway or third-party product.

If you select **External**, you are choosing to use an external database. Complete the database connection fields in the database sections under:

- *Access and security* on page 52
- *Visibility* on page 54
- *Dashboards database* on page 56

The following fields apply only when you have selected to use the internal MySQL database for storing application data.

Port

Port for the embedded MySQL database.

Root and confirm root password

Password for the embedded MySQL database. The root user can create other users of the service.

Access and security

Executive port

Internal administrative port that runs and monitors the Access and Security service.

HTTP port

User interface and API server port for HTTP plain connections.

HTTPS port

User interface and API server port for HTTP connections over SSL.

PKI port

PKI legacy socket server. Central Governance does not use this port by default.

PKI SSL port

PKI legacy secure socket server. Central Governance does not use this port by default.

HTTPS client authentication port

Client authentication for HTTP over SSL. This port is used when Transfer CFTs register with Central Governance.

Component authentication

Shared secret and confirm shared secret

The value you set for the shared secret is used by products when registering with Central Governance. You must provide this value to operators of products before they attempt to register.

The shared secret, like passwords, is encrypted in the database.

Encryption

Key for encrypting and decrypting passwords in the database and encrypting when exported. Also, encrypting private certificates and keys when exported. This key is the default when exporting. The value must be at least 8 characters.

Confidential information such as passwords and private certificates used in Central Governance are encrypted to enhance security. The encryption algorithm is based on the key you enter.

Database

Fields to complete depend on whether you selected to use an internal or external database for the application database.

If internal, the fields are:

User

User of the database schema.

Password and confirm password

User password.

If external, select the database type and complete the fields for connecting to the database.

The database user must have rights to create tables.

For Oracle, you can define the URL using one of the following methods:

- Using the SID of the database. For example:

  ```
  jdbc:oracle:thin:@{host}:{port}:{SID}
  ```

- Using the service name of the database. For example:

  ```
  jdbc:oracle:thin:@{host}:{port}/{serviceName}
  ```

If using Oracle RAC, the URL must include the service name.

You only have to create the database or schema in the database application. Central Governance creates the tables when you start the server the first time. However, you must use a different database or schema for access and security, visibility and dashboards.

If the database is external, you can click **Check database connection** to verify the values for connecting to the database.

See *Secure external database connections* on page 49 if you want a secure JDBC connection.

Business certificate authority

Generates end-entity certificates used by products to secure transfers. Use the default Central Governance intermediate certificate used by the internal CA to generate end-entity certificates. Or, select the custom option and import a password-protected JKS or P12 certificate authority file.

If you choose the custom option, the imported file must contain only one certificate. In the certificate the Basic constraint **isCA** must be set to **true**, indicating the certificate is a self-signed root certificate or an intermediate certificate.

Best practice is to change the default business CA with a CA certificate signed by a known CA.

After registering Transfer CFTs in Central Governance, changing the business CA might affect flows. See *If CAs change after Transfer CFT registration* on page 101 for more information.

Governance certificate authority

Generates end-entity certificates used by Central Governance to secure communications internally and with other products. Use the default Central Governance intermediate certificate used by the internal CA to generate end-entity certificates. Or, select the custom option and import a password-protected JKS or P12 certificate authority file.

If you choose the custom option, the imported file must contain only one certificate. In the certificate the Basic constraint **isCA** must be set to **true**, indicating the certificate is a self-signed root certificate or an intermediate certificate.

Best practice is to change the default governance CA with a CA certificate signed by a known CA.

After registering Transfer CFTs in Central Governance, changing the governance CA might result in Transfer CFTs becoming unavailable. See *If CAs change after Transfer CFT registration* on page 101 for more information.

Visibility

Front-end port

Listening port for events such as flow monitoring, alert notifications and auditing. The port is used by Central Governance and registered products that use the Visibility service.

RMI port

Communications port between the user interface and the server.

HTTP port

Port for connecting in a browser to the Visibility service correlation user interface.

HTTP stop port

Port for administrating the web server that runs the dashboards option.

HTTPS port

Port for connecting in a browser to the dashboards user interface over SSL.

SSO port

Port for connecting in a browser to the dashboards user interface over SSO.

Database

Fields to complete depend on whether you selected to use an internal or external database for the application database.

If internal, the fields are:

User

User of the database schema.

Password and confirm password

User password.

If external, select the database type and complete the fields for connecting to the database.

The database user must have rights to create tables.

For Oracle, you can define the URL using one of the following methods:

- Using the SID of the database. For example:

  ```
  jdbc:oracle:thin:@{host}:{port}:{SID}
  ```

- Using the service name of the database. For example:

  ```
  jdbc:oracle:thin:@{host}:{port}/{serviceName}
  ```

If using Oracle RAC, the URL must include the service name.

You only have to create the database or schema in the database application. Central Governance creates the tables when you start the server the first time. However, you must use a different database or schema for access and security, visibility and dashboards.

If the database is external, you can click **Check database connection** to verify the values for connecting to the database.

See *Secure external database connections* on page 49 if you want a secure JDBC connection.

Dashboards database

Fields to complete depend on whether you selected to use an internal or external database for the application database.

If internal, the fields are:

User

User of the database schema.

Password and confirm password

User password.

If external, select the database type and complete the fields for connecting to the database.

The database user must have rights to create tables.

For Oracle, you can define the URL using one of the following methods:

- Using the SID of the database. For example:

  ```
  jdbc:oracle:thin:@{host}:{port}:{SID}
  ```

- Using the service name of the database. For example:

  ```
  jdbc:oracle:thin:@{host}:{port}/{serviceName}
  ```

If using Oracle RAC, the URL must include the service name.

You only have to create the database or schema in the database application. Central Governance creates the tables when you start the server the first time. However, you must use a different database or schema for access and security, visibility and dashboards.

If the database is external, you can click **Check database connection** to verify the values for connecting to the database.

See *Secure external database connections* on page 49 if you want a secure JDBC connection.

Transfer CFT connector

Registration port

Port on which Central Governance listens for initial connections from Transfer CFTs when the Transfer CFTs are registering with Central Governance.

Secured communication port

Port used for mutually authenticated communications between Central Governance and Transfer CFTs.

Internal storage

Port for the embedded MongoDB NoSQL database for storing configuration data.

Root and confirm root password

Password of root user for the embedded NoSQL database. The root user can create other users of the service.

User

User of the NoSQL database. This is the user Central Governance uses to communicate with the internal storage database.

Password and confirm password

User password.

Save and start

Review the values on the configuration page. Click **Save and start** when you are sure the values are correct.

Do not install or run Central Governance as root on Linux. Use a common user.

After clicking **Save and start**, a page is displayed showing the startup status. If all goes well, green check marks are displayed for the following nodes:

- Application database
- Access and Security
- Visibility
- Internal storage
- Core services
- Transfer CFT connector

When Central Governance has started, you are prompted to click a link for opening the log-on page in a browser. See *Logging on* on page 61.

An X within a red circle indicates a problem with a node. If this occurs, the system rolls back any nodes that had started before encountering the problem node. The rollback stops and deletes any nodes that had been added successfully. After the rollback, you can click **Edit configuration**, check values and try again to start.

You can review the `cgcmd.log` file in the Central Governance `logs` directory for troubleshooting clues. You also can review the `cg_support_yyyy-mm-dd_hh-mm-ss` file that writes to the Central Governance install directory when a system start fails and rolls back. This compressed file contains a copy of the `initial-settings.properties` file and copies of the Central Governance logs, config and scripts directories. It also contains log files for nodes and other node files. You can send the file as an email attachment to Axway support when working with them to troubleshoot an issue.

Default ports and firewall requirements

The following are the default ports used in Central Governance, except when noted for external systems, to listen for connections. All ports are configurable during the configuration process after installation or later. You can place the cursor over a port field to display the default value on the configuration web page.

If a firewall is in use, open the ports marked as required or optional, when applicable, in the following table. This enables communications with remote systems.

- Required - Port is needed to enable communication between Central Governance and registered products.
- Optional - Port must be opened only when the functionality is used.
- Not required - Communications are internal to Central Governance and opening ports does not apply.

When **reason = external server** the port is used by an external system to listen for connections, not internally by Central Governance, and must be opened on the remote computer. Port 444, used by SecureTransport for REST API, also is an external port.

Default port	Use	Used by	Reason	Allow port through firewall
External				
25	External SMTP server to send outbound messages		External server	Required
389	External LDAP server		External server	Optional
444	REST API	SecureTransport Administrator	Product communication	Required
1433	External SQL Server application database	SQL Server	External server	Optional
1521	External Oracle application database	Oracle	External server	Optional
3306	External MySQL application database	MySQL	External server	Optional
Internal				

Default port	Use	Used by	Reason	Allow port through firewall
1305	Visibility event server for monitoring, alerting and auditing events	Registered products	Visibility events	Required
1308	Visibility service RMI			Not required
1309	Visibility HTTP service (for example, correlation user interface)	End user	UI administration	Optional
1766	Web Services	Registered products	Product communication	Required
3307	Embedded MySQL application database			Not required
5117	Access and Security service executive			Not required
5701	Agent cluster for communication with product agents	SecureTransport	Product communication	Required
6090	Access and Security service HTTP	End user		Not required
6453	Access and Security service HTTPS	End user	UI administration	Optional
6666	Access and Security service HTTPS client authentication	Transfer CFT	Product communication	Required
6667	Visibility service dashboards SSO			Not required
6900	Central Governance user interface.	End user	UI administration	Required
7000	Access and Security service PKI socket server			Not required
7101	Access and Security service PKI secure socket server			Not required

Default port	Use	Used by	Reason	Allow port through firewall
8005	Visibility service dashboards HTTPS			Not required
8081	Core Services HTTPS			Not required
8082	Central Governance configuration user interface	End user	UI administration	Required
8085	Visibility service dashboards HTTP			Not required
12553	Transfer CFT connector registration	Transfer CFT	Product communication	Required
12554	Transfer CFT connector communication	Transfer CFT	Product communication	Required
27017	MongoDB NoSQL database for internal storage			Not required

Resolving port conflicts

If you suspect a port conflict, use the **netstat** command to generate a list of ports in use on your system. You can resolve conflicts by changing the port used by Central Governance or by another application or process.

The command can be executed in the following ways.

Windows

In a command prompt or DOS window, type **netstat -a -n** or **netstat -an** to display a list of ports in use. You can instead type **netstat -a -n | more** to page through the list.

Unix and Linux

On a command line, type **netstat -a -n** or **netstat - an** to display a list of ports in use. Or, to find whether a specific port is in use, type **netstat -a | grep [port number]**.

Processes

The following table lists the processes that are running when all Central Governance nodes are started. Some nodes have more than one process.

The Agent and some nodes are Java virtual-machine processes.

Process	Description
java	Agent
mongod	Internal storage node
mysqld	Application database node *
java	Operating node
java	Access and security node JVM 1
java	Access and security node JVM 2
java	Visibility node JVM 1
java	Visibility node JVM 2
java	Visibility node JVM 3

* When Central Governance is used with the embedded MySQL database.

The processes are the same on all supported operating systems, but on Windows have the extension **.exe**.

Logging on

You are ready to log on the user interface after Central Governance has started.

Default credentials

Use the temporary password for the default credentials only when logging on the first time. Use your own assigned credentials if you have them.

- **Org** is the organization
- **admin@first.use** is the user ID
- **Initial01** is the temporary first-time password

You are prompted to change the temporary password when logging on as this user the first time.

Open log-on page

You can open the Central Governance log-on page in a browser by:

- Clicking the link at the bottom of the configuration status page after Central Governance has started.

  or

- Opening the log-on page with a URL in the following format:

  ```
  https://<host>:<UI port>
  ```

Where:

- `<host>` is the fully qualified domain name or IP address of the computer running Central Governance
- `<UI port>` is the port Central Governance listens for connections. The default is 6900.

If a message is displayed about an untrusted certificate, you must accept the certificate to continue to the log-on page. The message is normal with some browsers; you can ignore the warning.

Log on

Log on with your assigned credentials or the default credentials with the temporary password if logging on the first time. You can click **Help** at the top right of the page after logging on to open the online help.

Audit reports

Many events related to executed actions are tracked, typed and stored in the database. These can be actions by users, the server, organizations or managed products. Central Governance enables you to search for and display this data.

Select **Administration > Audit** to generate an audit report in the Visibility service dashboards user interface.

After executing a search, you can use controls at the top to perform other tasks on the search results page, such as filtering. You can get more information by selecting **Help** in the Visibility user interface.

Supported browsers and requirements

Central Governance supports the following web browsers for navigating the user interface and help. Although all of these are supported, tests have indicated the UI performs best in Chrome.

Client OS	Browser	Browser version
Windows 7 - 32 and 64 bit	Internet Explorer	11
Windows 7 - 32 and 64 bit	Chrome	Latest

Client OS	Browser	Browser version
Windows 7 - 32 and 64 bit	Firefox	Latest
Windows 7 - 32 and 64 bit	Firefox Extended Support Release (ESR)	Latest
Windows 8.1 - 64 bit	Internet Explorer	11
Windows 8.1 - 64 bit	Chrome	Latest
Windows 8.1 - 64 bit	Firefox	Latest
Windows 8.1 - 64 bit	Firefox Extended Support Release (ESR)	Latest

Browsers must support the following:

- Browsers must accept cookies from the Central Governance user interface.
- On any supported version of Internet Explorer disable compatibility view if you encounter display issues while using the help.
- Local storage must be activated in all supported browsers. It is active in all browsers by default, but the following is how to verify:
    - In Chrome, go to **Settings | Show advanced settings | Privacy | Content settings | Cookies**. Make sure the following is not selected: **Block sites from setting any data**.
    - In Firefox, enter the address **about:config** and click **I'll be careful, I promise** if prompted. Scroll down to **dom.storage.enabled** and make sure the value is **true**.
    - In Internet Explorer, go to **Tools | Internet options | Advanced**. Under **Settings | Security**, make sure the following is selected: **Enable DOM storage**.
- Central Governance has an HTML5-enabled user interface. The JavaScript option must be activate on your browser:
    - In Chrome, go to **Settings | Show advanced settings | Privacy | Content settings | JavaScript**. Make sure the following is selected: **Allow all sites to run JavaScript**.
    - In Firefox, go to **Tools | Options**. Make sure the following are selected: **Block pop-up windows**, **Load images automatically**, and **Enable JavaScript**.
    - In Internet Explorer, go to **Tools | Internet options | Security | Security level for this zone**. Under **Settings | Scripting**, make sure the following is selected: **Enable scripting of Java Applets**.

The recommended screen resolution is 1200x800. The minimum supported screen resolution is 800x600.

Tips for using the user interface

Standard website usability guidelines apply to the Central Governance user interface.

- Add bookmarks to the pages you use often. Each page has a unique URL you can use to access it directly.
- Refresh each page using the browser's page refresh or reload option.
- Navigate through the pages using the browser's back and forward buttons.
- Open a new page in a different browser tab. Best practice is opening several browser tabs to access pages quickly.

Some Central Governance context parameters are stored in the browser. If you connect to the UI using the same browser on the same machine, the same context is available. Note that context does not depend on the user connected, but is relative to the URL domain. Central Governance context parameters include:

- The filters applied on each grid
- The grid customization (columns displayed, columns order and size)

If you have problems viewing or navigating the help, accessed via help links throughout the user interface, refresh or reload the page. Or clear your browser's history or cache, restart the browser and try again.

License file

An XML license file controls the functionality you are entitled to use. For example, it specifies whether you can use an embedded or external database for storing application data.

Before installing or upgrading, make sure you have obtained a license file for Central Governance from Axway. Verify it enables the features you want. The license specifies:

- Hosts where Central Governance can be installed. The license must support multiple hosts if you plan to run Central Governance in a cluster environment of multiple computers.
- The supported operating system.
- The supported embedded or external database.
- An expiration date. If there is not a date, the license is perpetual.

After installing, during the initial configuration of Central Governance, you are prompted to enter the location of the license file. After starting the server the first time, the file is stored in:

```
<install directory>\runtime\com.axway.nodes.ume_
<UUID>\conf\license\license.xml.
```

Do not move, rename or delete the file. Any attempt to change the contents makes the product inoperable. The file is hashed and signed to protect it from tampering. You can, however, open the file and review its contents.

If you receive a new license file, you can stop Central Governance and run `cgcmd configure` to replace the old license file. For example, you may receive a new license to replace one that has expired.

If the license expires while Central Governance is running, it keeps running but once stopped cannot be restarted.

Flow and transfer monitoring

Many events related to flow execution are tracked, typed and stored in the database. You can search for and retrieve the data via the Central Governance or Axway Sentinel user interfaces. The Central Governance Visibility service is based on Sentinel.

After retrieving data, you can use toolbar controls to perform other tasks on the displayed data.

Options to retrieve data

In Central Governance you can select **Flows > Flows Report** to generate a report of all flows that have executed in the past 7 days. You also can select **Administration > Dashboards** to open the dashboards page and then select **My documents > View all flows** to generate the same report. You can use the **Filter document** control to find specific data or generate more data.

Another option is to select **Flows > Monitoring** to open the Visibility UI, which is based on the Sentinel UI. Select **View_Flows > View all flows** to display fields for filtering data. Only start and end dates are required. All other filtering fields are optional. Click **Execute** to perform a search.

Actions

Once data are displayed, you can use toolbar icons to perform actions, such as running commands.

For example, select one or more transfer records and click **Perform action** to display available actions. Depending on the state of the transfer, for Transfer CFT you can:

- Restart the transfer
- Cancel the transfer
- Pause the transfer

The following describes each of these actions.

Restart the transfer

This command reactivates a suspended or held transfer. It corresponds to the Transfer CFT START transfer control command. The Transfer CFT requesting the transfer initiates the restart. Eligible transfer statuses are:

AVAILABLE
CANCELED

INTERRUPTED
SUSPENDED
PRE_PROC_ABORT

After restarting, transfers in the Transfer CFT H or K phasestep in the catalog change to the D phasestep. These transfers are restarted after scanning the catalog for availability of the required resources.

Cancel the transfer

This command suspends one or all of the send or receive transfers with selected partners and puts them in Transfer CFT KEEP status. It corresponds to the Transfer CFT KEEP transfer control command. Eligible transfer statuses are:

TO_EXECUTE
AVAILABLE
RECEIVING
SENDING
PRE_PROC

Suspended transfers are set to the Transfer CFT K phasestep. Transfer CFT ensures the integrity of the suspended data. Depending on the protocol, it authorizes restarting the transfer from the last synchronization point set before the pause or from the beginning of the file.

Pause the transfer

This command interrupts a transfer in process and puts it in Transfer CFT HOLD status. It corresponds to the Transfer CFT HALT transfer control command. Eligible transfer statuses are:

TO_EXECUTE
RECEIVING
SENDING
PRE_PROC

The halted transfers are set to the Transfer CFT H phasestep in the catalog. They can be reactivated:

- By an operator START command
- On receiving a transfer reactivation request from the partner

Transfer CFT ensures the integrity of the data. Depending on the protocol, it authorizes restarting the transfer from the last synchronization point set before the interruption or from the beginning of the file.

Dashboards and reports

Dashboards and reports transform file-transfer data into meaningful graphical displays. These are a feature of the Central Governance Visibility service. You can add, edit, view and delete them in the user interface.

Dashboards are containers of reports. There also can be reports that are not within dashboards.

Run dashboards and reports

1. Select **Administration > Dashboards** to open the dashboards UI.
2. Select **My documents** on the menu to display a list of available dashboards and reports.
3. Select an entry to run it.

Default dashboards and reports

You can use the default dashboards and reports as-is or customize them. In the dashboards user interface, select **Help > Help** to open a help system with information about dashboard configuration.

The role assigned to your user governs the reports you can access and run.

Users assigned to the IT Manager role can run and customize the following:

- IT Manager dashboard, which contains:
  - Transfer status by product
  - Product statuses
  - Transfer status overview
- Audit report
- Transfer error rate history report

Users assigned to the Middleware Manager role can run and customize the following:

- Middleware Manager dashboard, which contains:
  - Transfer status by application
  - Transfer status by flow
- Protocols usage dashboard, which contains:
  - Protocols used
  - Protocols used (ratios)
- Transfer trends dashboard, which contains:
  - Average transfer rate
  - Exchanged data volume
  - File transfer count
  - Transfers per file size

- Trend summary dashboard, which contains
  - Day trend - Bytes (MB)
  - Day trend - Files
  - Day trend - Transfer rate
  - Week trend - Bytes (MB)
  - Week trend - Files
  - Week trend - transfer rate
  - Month trend - Bytes (MB)
  - Month trend - Files
  - Month trend - Transfer rate
- And the following reports:
  - SecureTransport - File processing
  - Today's flows analysis
  - Transfer error per hour today
  - View all flows report

User privileges for dashboards and reports

The default roles IT Manager and Middleware Manager contain privileges that enable users with these roles to manage dashboards and reports, as outlined in *Default dashboards and reports* on page 67. Other users who need such permissions must be assigned to roles containing one or both of the following predefined privileges or user-defined privileges based on the same Sentinel Web Dashboard resource as these:

View Web dashboards and reports

This predefined privilege enables viewing dashboards and reports.

Manage Web dashboards, access, reports and database

This predefined privilege enables managing dashboards and reports.

Caution about audit and flow reports

Just as with other default reports, you can change or delete the audit and view all flows reports. However, deleting these reports disables the ability to run them using the toolbar options **Flows > Flows Report** and **Administration> Audit**. Best practice is to keep these reports and not delete them.

Database administration 3

The following topics provide information about managing Central Governance databases, internal and external. This is for database administrators or other users responsible for database maintenance and performance, including backing up and restoring data.

Central Governance has two embedded databases:

- Internal storage is a NoSQL MongoDB database for Central Governance configuration data, including policies, flows and partner definitions.
- Application database is a MySQL database for transfer tracking data, user roles and privileges, certificates and dashboards. Central Governance can be configured to use an external database instead for this purpose.

Although you can perform maintenance, do not attempt to install updates for the embedded databases. Database updates are applied when installing Central Governance upgrades, service packs or patches.

Internal storage database maintenance

The following topics describe maintenance tasks for the embedded MongoDB database. The database contains configuration data for Central Governance. A different database, the application database, stores transfer tracking data, user roles and privileges, certificates and dashboards.

The commands to use are in the in the MongoDB `bin` directory at:

```
<install directory>\runtime\com.axway.nodes.mongodb_<UUID>\mongodb-<data string>\bin
```

Prerequisites

You need the user name and password for the MongoDB database to back up and restore data. You can look up the user name by stopping Central Governance and running `cgcmd configure` to open the Central Governance configuration page. The database user name is in the User field under the Internal Storage section of the page.

There also is a Password field, but the value is hidden. If you have forgotten the password, you can set a new one on the configuration page. See *Configuration and startup* on page 47 for more information about the page.

Silent mode option

When backing up and restoring data, you can enter the password with `-p <password>` or use `-p` only for silent mode. In silent mode, you are prompted to enter the password after executing the command, but the value is hidden.

Backing up data

You can export all data in binary format to a backup file with the `mongodump` command in the MongoDB `bin` directory. You can back up data when the database is running or stopped.

The following commands create a backup named `dump/` in the current directory. It contains a file in BSON format for each exported collection.

When running

```
mongodump --db umcft -u <user> -p <password>
```

When stopped

```
mongodump --dbpath {dbpath_value} --db umcft -u <user> -p <password>
```

The value of `dbpath` is specified in the database configuration file. The file is named `mongo.ini` and is at:

```
<install directory>\runtime\com.axway.nodes.mongodb_<UUID>\mongo
```

The default value is `./mongo/data`.

Once you have created the backup file, copy it to a directory outside of the Central Governance installation directory. This makes sure the file is available when you want to use it to restore data.

Restoring data

You can restore data in a backup file to a new or existing database with the `mongorestore` command in the MongoDB `bin` directory. You can restore data when the database is running or stopped.

When running

```
mongorestore -u <user> -p <password>
```

When stopped

```
mongorestore –dbpath {dbpath_value} --journal -u <user> -p
<password>
```

Before restoring data, you can first purge the current database by running the following command:

```
mongorestore --drop -u <user> -p <password>
```

These commands restore the database dump in the `dump/` directory. However, you can define the path to the directory where the dump files are located. For example, if the database is running:

```
mongorestore {dump_path} -u <user> -p <password>
```

More information

Visit http://www.mongodb.org/ for more information about the MongoDB database.

Embedded application database maintenance

The following topics describe maintenance tasks for the embedded MySQL application database.

Data directory

Data are stored in the following directory:

```
<install directory>\runtime\com.axway.nodes.mysql_<UUID>\data
```

The directory location is not configurable.

When Central Governance is running on Linux, you must execute the following command from the `<install directory>\runtime\com.axway.nodes.mysql_<UUID>\mysql\bin` directory to connect to the database before running other commands:

```
mysql -S ../../data/axwayDB.socket -u root -p
```

Backing up data

You can export all data to a backup file with the `mysqldump` command in the MySQL `bin` directory. The database must be running to back up data.

Run the following command to generate a `dumpfile.sql` file in the current directory. The file contains all queries for restoring the database.

```
mysqldump -u {conf.db.root.login} -p{conf.db.root.password} -P
{conf.db.port} --all-databases > dumpfile.sql
```

Once you have created the backup file, copy it to a directory outside of the Central Governance installation directory. This makes sure the file is available when you want to use it to restore data.

Restoring data

You can restore data in a backup file to a new or existing database with the `mysql` command in the MySQL `bin` directory. The database must be running to restore data.The following command overwrites all existing data.

```
mysql -u {conf.db.root.login} -p{conf.db.root.password} <
dumpfile.sql
```

Make sure there is not a space between the `-p` password parameter and the value.

More information

Visit http://www.mysql.com/ for more information about the MySQL database.

Flow monitoring and audit data maintenance

Whether you use the embedded application database or an external database for this purpose, regular purging or archiving or both is recommended for some types of data to avoid disk space issues:

- Data related to audit reports (see *Audit reports* on page 62).
- Data related to monitoring of flow transfers, which are data stored in the XFBTransfer Tracked Object.

These records are in the database for the Central Governance Visibility service, which is based on Axway Sentinel.

Never purge other data in the application database. This includes other data for the Visibility service and any data in the Central Governance Access and Security service and Dashboards databases. Also, never purge data in the internal MongoDB database.

Best practices:

- Define the rules and interval for purging and archiving data. These decisions are at your discretion, according to your organization's policies and practices. You can review database sizing recommendations in the prerequisites section of the Central Governance Installation Guide.
- Define the data to purge or archive or both. This should include only the audit and flow transfer monitoring data.
- Apply a data purge and archive procedure.

The Visibility service has tools for archiving and purging data in the following directory:

```
<install directory>\runtime\com.axway.nodes.sentinel_
<UUID>\sentinel\tools
```

The following are examples of archiving the historic and current flow transfer monitoring data in the XFBTransfer Tracked Object:

- Archive of the current data:

```
trkcmd archive -tabname "TrkTable(XFBTransfer, current)"
```

- Archive of the historic data:

```
trkcmd archive -tabname "TrkTable(XFBTransfer)"
```

The result of these archive commands is two XML files in the following directory:

```
<install directory>\runtime\com.axway.nodes.sentinel_
<UUID>\sentinel\archive
```

The following are examples of purging the historic and current flow transfer monitoring data in the XFBTransfer Tracked Object:

- Purge of the current data:

```
trkcmd purge -tabname "TrkTable(XFBTransfer, current)" -delay 0
```

- Purge of the historic data:

```
trkcmd purge -tabname "TrkTable(XFBTransfer)" -delay 0
```

You can also restore some archived data with the following command:

```
trkcmd restore -tabname "TrkTable(XFBTransfer)" -file my_archived_
data.xml
```

For more details, see the Sentinel user documentation.

Tools 4

Central Governance has tools for performing routine and advanced tasks.

cgcmd

The `cgcmd` command starts and stops Central Governance, displays system status and performs basic configuration.

Command line interface

The command line interface (CLI) enables you to perform operations on Central Governance services and products.

The following topics provide more details.

cgcmd command

The `cgcmd` command starts and stops Central Governance, displays system status and performs basic configuration.

The `cgcmd` command is in the Central Governance install directory. You must run it from that directory. The syntax is `cgcmd <parameter>`.

Parameters

The following are the `cgcmd` command parameters.

Note If you want to stop or start an individual product or start a Central Governance service, see *Command line interface* on page 78.

You can run any parameter, except `help`, with a `--verbose` option to display more information when the command executes. For example:

```
cgcmd status --verbose
```

configure

Starts an internal web server that hosts a web page for configuring Central Governance. Central Governance must be stopped before you can run configure mode.

Once the web server has started, the command lists the URL for opening the web page in a browser. If the computer on which the command was executed has a default browser, the page opens automatically. Otherwise, open the page with the provided URL.

By default the web server runs on port 8082. But you can change the port when invoking the command. For example:

```
cgcmd configure -p <port>
```

When the configuration page opens, complete configuration fields as needed. Click **Save and start** when done. The system starts and the settings are applied.

See *Configuration and startup* on page 47 for more details.

help

Displays a list of all parameters and descriptions. It also lists the return codes and descriptions of all parameters. Invoking `cgcmd` without a parameter also displays the list of parameters.

repair

Restores the Default User admin@first.use to its initial password, user ID, role and organization. If the Default User has been deleted, it also re-adds the user to its original state. See *About the Default User* on page 118 for more information about this user.

Central Governance must be running to use this parameter.

restart

Stops and then starts Central Governance and all of its services.

If you run Central Governance on Windows as a service, start or stop the service, automatically or manually. Do not use `cgcmd restart`.

start

Starts Central Governance and all of its services, which also are called nodes. The initial configuration must be completed before Central Governance can be started.

If you run Central Governance on Windows as a service, start or stop the service, automatically or manually. Do not use `cgcmd start`.

See *Startup behavior* on page 76 for more information.

status

Displays the current started or stopped status of Central Governance. Use the `verbose` option to also display statuses of all nodes.

stop

Stops Central Governance and all of its nodes.

If you run Central Governance on Windows as a service, start or stop the service, automatically or manually. Do not use `cgcmd stop`.

support

Packages Central Governance log files and other files in a compressed file. You can send the file as an email attachment to Axway support when working with them to troubleshoot an issue.

When you run `cgcmd support`, the file is added to the Central Governance install directory. The file name is in the format `cg_support_yyyy-mm-dd_hh-mm-ss`. The file type is ZIP on Windows and TGZ on Linux.

Included in the package are a copy of the `initial-settings.properties` file and copies of the Central Governance logs, config and scripts directories. It also contains log files for nodes and other node files.

If Central Governance is on Linux, make sure that the interface configuration (ifconfig) system administration utility is in the system PATH variable and that the user has permissions to use it. The ifconfig utility must be available to enable the `cgcmd support` command to collect the maximum amount of data.

version

Displays the version of Central Governance and of all applied service packs and patches.

Also displayed are versions of all nodes, components and node archive files.

If node versions are not reported, Central Governance is not fully installed or initial configuration has not been completed.

Startup behavior

Starting Central Governance can result in a fully or partially running system, depending on whether the agent and all of the nodes are started successfully.

Central Governance is fully running when the agent and the following nodes, or services, are started:

- Application database
- Access and Security
- Visibility
- Internal storage

- Core services
- Transfer CFT connector

All are core nodes except Transfer CFT connector and Visibility.

If a core node cannot be started, the `cgcmd start` command is stopped and startup fails.

If the Transfer CFT connector or Visibility or both cannot be started, Central Governance is partially started. When partially started, users can connect to the user interface, but might be unable to use all features.

Troubleshoot start or stop failures

The following are troubleshooting guidelines when Central Governance fails to start or stop.

- Check the log files in the Central Governance `logs` directory. You might have to manually stop the Agent process.
- If you are working with Axway support, you can send them an archive file containing logs and other system files helpful in troubleshooting. Central Governance generates the archive automatically on start and stop failures or you can generate it manually. See *support* on page 76.
- A timeout is reached when stopping. Wait until Central Governance has stopped completely and check with `cgcmd status`.
- Central Governance fails to stop. Check with `cgcmd status`. You might have to manually stop processes.
- Expected number of nodes is not correct. For example, only three of six nodes were detected. Review the node logs, try to resolve the issue, and start the system again.
- Some node types are missing in the Central Governance installation. Review the agent and cgcmd logs, try to resolve the issue, and start the system again.
- Some node types are in error. Check their log files. Try to stop Central Governance and start it again.
- Some nodes are still starting. Wait until all nodes have started.
- Some nodes have not started. Try to start Central Governance again.

Crashed or StartingInError recovery

If you run `cgcmd status --verbose` and one or more nodes have a status of `Crashed` or `StartingInError`, run `cgcmd restart` to stop and then start Central Governance and all of the nodes. Alternatively, you can run `cgcmd stop`, manually kill any hung processes, then run `cgcmd start`.

Command line interface

The command line interface (CLI) enables you to perform operations on Central Governance services and products.

Prerequisite

Central Governance must be running.

Usage

The CLI utility is in the Central Governance `cli` directory. The tool is named `cgcli.bat` on Windows or `cgcli.sh` on Unix and Linux.

You must enter user credentials to run CLI commands or log on to a console mode. Credentials are user ID, organization name and password. However, users only must supply passwords the first time they use CLI. Thereafter, users only need enter user ID and organization name.

For example, from the `cli` directory, a first-time CLI user executes the following command to list CLI user credential rules and all commands and descriptions:

```
cgcli -u <user ID> -o <user organization name> -p
```

The `-p` parameter initiates a prompt to enter the user password. Passwords are hidden when entered. After typing the password and pressing **Enter**, all rules and commands and descriptions are listed.

The first time you use CLI, your password is saved. Subsequently, you only need enter your user ID and organization name. You can use the `-p` parameter, but it is optional unless your password has changed. For example:

```
cgcli -u <user ID> -o <user organization name>
```

If a user ID, organization name or password contains spaces, enclose the value in quotes "like this". Quotes also are required for values of any other parameters containing spaces.

New users must log on to the Central Governance user interface and change their initial passwords before using CLI. The tool does not accept the temporary passwords assigned to new users.

There are short and long forms of command parameters. For instance, the **user** parameter can be used in the following forms:

- Short: `-u <user ID>`
- Long: `--user <user ID>`

So in the previous example to list credential rules, commands and descriptions, you can execute the following long forms to generate the same list:

First-time user

```
cgcli --user <user ID> --organization <user organization name> --
password
```

User with saved password

```
cgcli --user <user ID> --organization <user organization name>
```

You invoke CLI with commands. Commands have required, optional or no parameters. The following are examples of short-form syntax:

First-time user

```
cgcli -u <user ID> -o <user organization name> -p <command>
```

User with saved password

```
cgcli -u <user ID> -o <user organization name> <command>
```

See *CLI commands* on page 84 for descriptions of the CLI commands and their uses.

CLI displays help whenever there is a syntax error in command-line use. Details of commands are displayed only when user authentication works.

CLI modes

The utility has two modes: normal and console.

Normal mode

You must enter user credentials each time you want to run a command in normal mode. For example:

First-time user

```
cgcli -u <user ID> -o <user organization name> -p <command>
```

User with saved password

```
cgcli -u <user ID> -o <user organization name> <command>
```

Console mode

You log on with your user credentials to run CLI in console mode. Once logged on, you can run commands without entering credentials each time.

Use the **console** command to log on. For example:

First-time user

```
cgcli -u <user ID> -o <user organization name> -p console
```

User with saved password

```
cgcli -u <user ID> -o <user organization name> console
```

Once logged on, user credentials are not required to invoke commands. Just run a command and any parameters. For example, to display the status of a specified Central Governance service:

```
serviceStatus -n <service name>
```

Type `exit` to log off console mode.

Permissions enforcement

Users must have roles with correct privileges to execute CLI commands. Without proper privileges, access-denied error messages are displayed and commands fail to run. The only exception is CLI help is not filtered against privileges and is accessible even to users who do not have roles.

The following table shows the resources and enabled actions that must be in privileges associated with roles for users to run CLI commands successfully. See *Roles and privileges* on page 114 for more information.

CLI command	Resource	Action	Comment
appExport	Application	View	
appImport	Application	View Modify Create	Create is required when the command is used without parameters. Modify is required for using the overwrite parameter. The View action is required in both cases.
flowDeploy	Flow	Deploy	
flowExport	Flow	View	

CLI command	Resource	Action	Comment
flowImport	Flow	Create Modify	Create is required when the command is used without parameters. Modify is required for using the overwrite parameter. The View action on the Flow resource also is required in both cases. A user also must have the Create action on the Application resource to use the import applications parameter. A user also must have the Create action on the Unmanaged Product resource to use the import unmanaged products parameter.
partnerExport	Partner	View	
partnerImport	Partner	View Modify Create	Create is required when the command is used without parameters. Modify is required for using the overwrite parameter. The View action is required in both cases.
policyExport	Policy	View	
policyImport	Policy	Create	
productConfigurationDeploy	Product Configuration	Deploy	
productList	Product	View	
productRestart	Product	Stop Start	
productStart	Product	Start	
productStop	Product	Stop	
serviceList	Service	View	
serviceLog	Service	View Configure	

CLI command	Resource	Action	Comment
serviceStart	Service	Start	
serviceStatus	Service	View	

The following is an example of console output when a user attempts to run the productList command without rights for the Product resource view action.

```
CLI> productList
CLI> Access denied. Contact your administrator if you require access.
CLI>
```

Use CLI remotely

You can use CLI on a computer that is not running Central Governance. This requires copying some Central Governance system files and having the remote computer access them. The remote computer also must set a path to a local instance of Java Runtime Environment (JRE) 1.8.

Central Governance must be running on the host computer for the remote computer to use CLI.

1. Create a directory on the computer running Central Governance. This directory is for putting copies of Central Governance files needed to run CLI. Give the directory a meaningful name. For example, `cli_remote`.
2. Copy the following files to the new directory:
   - The Central Governance `cli` directory
   - The Central Governance `data` directory.
3. Copy the directory to the remote computer.
4. In the `cli` directory open the `cgcli.properties` file for editing. Make sure the port and host properties are for Central Governance on the host computer.
5. In the `cli` directory open open the `profile` file for editing. Make sure the `JVM_EXECUTABLE` property is set to the JRE on the local machine. For example, on Windows, `C:\Program Files\Java\jre8\bin\java`.
6. Use CLI in normal or console mode. See *CLI modes* on page 79.

Typographical conventions

The following typographical conventions are used in command syntax and examples.

Symbols	Description
<text>	User-defined value is displayed inside angle brackets.
[parameter]	Optional parameters are displayed inside square brackets.
{parameter | parameter}	Required parameters, one of which must be selected, are displayed inside braces and separated by vertical lines.
[parameter | parameter]	Optional parameters, one of which can be selected, are displayed inside square brackets and separated by vertical lines.
-	Precedes the short form of the parameter.
--	Precedes the long form of the parameter.

Errors

The following table lists general errors that may result from the execution of any command.

Context	Error Output
Command contains incorrect parameters.	Unrecognized parameter: <parameter>. Command help is displayed.
Mandatory parameter is missing.	Missing parameter(s): <parameter1, parameter2>. Command help is displayed.
Parameter value is missing. This error occurs only for parameters that require a value. For example, on the `serviceStop` command, `--name` requires a value, but `-force` does not.	Missing parameter value for: <parameter_value>. Command help is displayed.
The server does not respond	Web server is not available.
An unexpected error occurred during command execution.	Unexpected error. The text of the actual error is displayed.

CLI commands

The following are the CLI commands and their uses.

console

Start CLI in console mode. See *Console mode* on page 79 for details.

appExport

Export a list of applications filtered by name or group.

You can use this command to export applications defined in Central Governance for testing and import them in a production Central Governance environment.

When exporting applications, the reference to products (not all the definition or configuration of the product) linked to applications also are exported. If the application is contained in groups, the group definition is exported, too.

```
appExport [-f <file path> -format <JSON|XML> -g <group> -n <name> -s]
appExport [--file <file path> --formatfile <JSON|XML> --group <group> --
name <name> --silent]
```

`<file path>` - The name of the exported file.

`<name>` – One or more application names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

`<group>` – One or more application group names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

If you do not use the optional **file** parameter, applications are exported to the Central Governance `cli` directory. The files have the prefix `export_app`.

If you do not use the optional **format** or **formatfile** parameter, applications are exported to JSON files. If you use it, you can specify JSON or XML as the file type.

The **group** and **name** parameters are optional. All applications are exported if you do not use them.

The optional **silent** parameter is for overwriting the target file, if any, without confirmation.

appImport

Import a list of applications or groups from a file.

See *Command usage details* on page 92 for more information.

```
appImport { -f <file name> } [ -o ]
```

```
appImport { --file <file name> } [ --overwrite ]
```

The optional **overwrite** parameter enables overwriting of applications by identifier. Be careful that groups are not overwritten.

`<file name>` - Path and file name of the application to import.

The command tries to find XML and JSON files to import. If neither format is found, importing is aborted and CLI reports the file has an invalid format.

flowExport

Export a list of flows filtered by name.

You can use this command to export flows defined in Central Governance for testing and import them in a production Central Governance environment.

When exporting flows, the definition of applications or groups of applications used in flows also are exported. The products linked to the flow directly or via applications and application groups are exported, too. Definitions of any unmanaged products are exported as well.

```
flowExport [-ek <key> -f <file path> -format <JSON|XML> -n <flow name> -
s ]
flowExport [--encryptionkey <key> --file <file path> --formatfile
<JSON|XML> --name <flow name> --silent]
```

`<key>` - The key for encrypting passwords and private certificates and keys. Although optional, Central Governance uses a default value if you do not specify one. The default value is set in the Encryption field on the Central Governance configuration page (see *Encryption* on page 53). Any value you specify must conform to the Central Governance password policy.

`<file path>` - The name of the exported file.

`<flow name>` - One or more names of flows separated by commas. Use an asterisk (*) as a wildcard to filter the results.

If you do not use the optional **file** parameter, flows are exported to the Central Governance `cli` directory. The files have the prefix `export_flow`.

If you do not use the optional **format** or **formatfile** parameter, flows are exported to JSON files. If you use it, you can specify JSON or XML as the file type.

The **name** parameter is optional. All flows are exported if you do not use them.

The optional **silent** parameter is for overwriting the target file, if any, without confirmation.

See *Prerequisites for promoting flows* on page 398 for more information.

flowImport

Import a list of flows from a file.

See *Command usage details* on page 92 for more information.

```
flowImport [ -ai | -app | -dk <key> ] { -f <file name> } [ -o | -up ]
flowImport [ --allowincomplete | --importapplications | --decryptionkey
<key> ] { --file <file name> } [ --overwrite | --importunmanagedproducts
]
```

The optional **allowincomplete** parameter enables import of flows where applications or products are missing.

The optional **importapplications** parameter enables import of applications.

The optional **overwrite** parameter enables overwriting of flows by name. Existing applications and unmanaged products are not overwritten.

The optional **importunmanaged products** parameter enables importing of unmanaged products. If used, definitions of the unmanaged products in the file must also be recorded in Central Governance.

`<key>` - The key for decrypting passwords and private certificates and keys. Although optional, Central Governance uses a default value if you do not specify one. The default value is set in the Encryption field on the Central Governance configuration page (see *Encryption* on page 53).

`<file name>` - Path and file name of the flow to import.

The command tries to find XML and JSON files to import. If neither format is found, importing is aborted and CLI reports the file has an invalid format.

See *Prerequisites for promoting flows* on page 398 for more information.

flowDeploy

Deploy flow definitions.

```
flowDeploy {-n <names> }
flowDeploy {--name <names> }
```

You must specify the names.

`<names>` - One or more flow names. Use commas to separate multiple names. Use an asterisk (*) as a wildcard to filter the results.

partnerExport

Export a list of partners filtered by name.

You can use this command to export partners defined in Central Governance for testing and import them in a production Central Governance environment.

When exporting partners, all related credentials and communication profiles of partners also are exported. Both communication profile types, client and server, are exported if present.

Server communication profiles are editable in partners, and client communication profiles are editable in flow protocols.

If partners contain public PGP keys, the keys also are exported.

```
partnerExport [-ek <key> -f <file path> -format <JSON|XML> -n <name> -s]
partnerExport [--encryptionkey <key> --file <file path> --formatfile
<JSON|XML> -- name <name> --silent]
```

`<key>` - The key for encrypting passwords and private certificates and keys. Although optional, Central Governance uses a default value if you do not specify one. The default value is set in the Encryption field on the Central Governance configuration page (see *Encryption* on page 53). Any value you specify must conform to the Central Governance password policy.

`<file path>` - The name of the exported file.

`<name>` – One or more partner names separated by commas. Use an asterisk (*) as a wildcard to

filter the results.

If you do not use the optional **file** parameter, partners are exported to the Central Governance `cli` directory. The files have the prefix `export_partner`.

If you do not use the optional **format** or **formatfile** parameter, partners are exported to JSON files. If you use it, you can specify JSON or XML as the file type.

The optional **silent** parameter is for overwriting the target file, if any, without confirmation.

partnerImport

Import a list of partners from a file, including all credentials of type certificates, SSH keys, PGP keys and logins, and all communication profiles of partners. Both types of communication profile types, client and server, are imported if present.

See *Command usage details* on page 92 for more information.

```
partnerImport [ -dk <key> ] { -f <file name> } [ -o ]
partnerImport [--decryptionkey <key>] { --file <file name> } [ --
overwrite ]
```

`<key>` - The key for decrypting passwords and private certificates and keys. Although optional, Central Governance uses a default value if you do not specify one. The default value is set in the Encryption field on the Central Governance configuration page (see *Encryption* on page 53).

`<file name>` - Path and file name of the partner to import.

The optional **overwrite** parameter enables overwriting partners by identifier.

The option to overwrite can be applied even if a partner is used in flows. Changes are applied to the partner, but you must check whether the flow is correct regarding new partner changes before deploying.

If communication profiles are already used by flows, Central Governance checks for communication profile compatibility based on type (client or server) and applies protocols before updating profile information. In case of an incompatibility, the whole partner is not imported.

A server communication profile or a credential of whatever type that is already used by flows is never removed by overwrite mode.

The command tries to find XML and JSON files to import. If neither format is found, importing is aborted and CLI reports the file has an invalid format.

policyDeploy

Deploy policies.

```
policyDeploy {-n <name>} [-s]
policyDeploy {--name <name>} [--silent]
```

You must specify the name parameter.

`<name>` – One or more policy names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

`s` or `silent` – Disables confirmation prompts. No value is required.

policyExport

Export a list of policies filtered by name. You can export policies defined in Central Governance for testing and import them in a production Central Governance environment.

```
policyExport [-f <file path> -format <JSON|XML> -n <policy name> -s]
policyExport [--file <file path> --formatfile <JSON|XML> --name <policy
name> --silent]
```

`<JSON|XML>` is the name of the file to export. If not specified, the exported file is saved to the current directory with a name in the format `export_policy_<date and time>`. The extension is JSON or XML depending on the file format.

`<file format>` is the format of the file to export. You can specify XML or JSON. If not specified, the default is JSON.

`<policy name>` is one or more policy names. Use commas to separate multiple names. Use an asterisk (*) as a wildcard to filter the results.

The optional **silent** parameter is for overwriting the target file, if any, without confirmation.

If the file to export already exists, the system prompts for overwrite confirmation.

policyImport

Import policies.

The command tries to find XML and JSON files to import. If neither format is found, importing is aborted and CLI reports the file has an invalid format.

See *Command usage details* on page 92 for more information.

```
policyImport [-o] {-f <file name>}
policyImport [--overwrite] {--file <file name>}
```

The optional **overwrite** parameter enables overwriting existing policies.

`<file name>` is the path and file name of the file to import.

productList

List all products registered in Central Governance. The name, status and version of products are listed.

```
productList [-n <name>] [-g <group>]
productList [--name <name>] [--group <group>]
```

`<name>` – One or more product names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

`<group>` – One or more group names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

productConfigurationDeploy

Deploy configurations for a registered product.

```
productConfigurationDeploy {-g <product group> | -n <product name>} [-
norestart -s]
productConfigurationDeploy {--group <product group> | --name <product
name>} [--noproductrestart --silent]
```

You must supply the group or name parameter.

`<product group>` - One or more product groups separated by commas. Use an asterisk (*) as a wildcard to filter the results.

`<product name>` – One or more product names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

`norestart` or `noproductrestart` – The product is not restarted after deployment. You must restart the product later for the changes to become effective.

`s` or `silent` – Disables confirmation prompts. No value is required.

productStart

Start a registered product (for example, Transfer CFT).

```
productStart {-n <name> | -g <group>}
productStart {--name <name> | --group <group>}
```

You must supply the name or group parameter.

`<name>` – One or more product names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

`<group>` – One or more group names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

productStop

Stop a registered product (for example, Transfer CFT).

```
productStop {-n <name> | -g <group>} [-s] [-mode]
productStop {--name <name> | --group <group>} [--silent] [-mode [normal
| quick | force]]
```

You must supply the name or group parameter.

`<name>` – One or more product names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

`<group>` – One or more group names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

`s` or `silent` – Disables confirmation prompts. No value is required.

`mode` – Specifies the manner of stopping. To stop normally, do not specify this parameter, since normal is the default behavior.

- `quick` – Can be used on a "started" or "in error" system.
- `force` – Can only be used on an "in error" system.

productRestart

Restart a registered product (for example, Transfer CFT).

```
productRestart {-n <name> | -g <group>} [-s]
productRestart {--name <name> | --group <group>} [--silent]
```

You must supply the name or group parameter.

`<name>` – One or more product names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

`<group>` – One or more group names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

`s` or `silent` – Disables confirmation prompts. No value is required.

serviceList

List all the Central Governance services. The name and status of services are listed.

```
serviceList
```

No parameters are required.

serviceStatus

Display the current status of the specified Central Governance service.

```
serviceStatus (-n <name>}
serviceStatus {--name <name>}
```

`<name>` – A valid service name. Execute the `serviceList` command to display a list of valid service names.

serviceStart

Start the specified Central Governance service.

```
serviceStart {-n <name>}
serviceStart {--name <name>}
```

`<name>` – A valid service name. Execute the `serviceList` command to display a list of valid service names.

serviceLog

List or change the levels of events written to log files for Central Governance services. The log levels, from lowest to highest verbosity, are:

- OFF
- ERROR
- WARNING
- INFO,
- DEBUG
- ALL

Verbosity refers to the quantity of events written to log files. For example, at the DEBUG level, many more events are written to log files than when the level is set to WARNING. However, the severity of events is the opposite. Severity from highest to lowest is: ERROR, WARNING, INFO, DEBUG, ALL, OFF.

Run `serviceLog` command without parameters to list the current logging level of all services.

The following is the syntax for changing log levels.

```
serviceLog [-l <log level>] [-n <name>]
serviceLog[--level <log level>] [--name <name>]
```

Omit the name parameter to change the logging level of all services.

`<log level>` - Log level to apply.

`<name>` – One or more service names separated by commas. Use an asterisk (*) as a wildcard to filter the results.

If you change a service log level to a severity higher than Central Governance, the Central Governance log level changes to that level, too. If you change the log level of Central Governance, the log level of all services changes to that level, too. For example:

1. If you set Central Governance to INFO, the log level of all services becomes INFO.
2. If you set the Visibility service to the lower DEBUG level, Central Governance remains at the higher INFO level.
3. If you set the application database service to the higher WARNING level, the Central Governance level changes from INFO to WARNING.

Command usage details

This topic provides details about using some of the more complex CLI commands.

The usage examples in this topic are independent of each other, except when a linkage between examples is noted.

applImport

You can use this command to import to Central Governance a list of applications in a file. The file can be exported by another instance of Central Governance or generated with an external tool.

If products linked to applications are not registered in Central Governance, the import of applications succeeds, but the command output reports the products were not found.

If an application has no products when imported, Central Governance check whether there are other products on the same host that are registered in Central Governance. If products are found, they are linked to the imported application.

If an application belongs to a group that is not defined in Central Governance, the group also is imported.

The option to overwrite is not applied to applications that are used in flows in Central Governance.

Usage example

The file `impApps.json` contains definitions of the following applications.

- The application Product Catalog Application is linked to the Transfer CFT CFT_PCA on host `host.product.catalog.application`.
- The application Store_001 is linked to the Transfer CFT CFT_Store_001 on host `host.store.001`.
- The application Store_002 is linked to the Transfer CFT CFT_Store_002a on host `host.store.002`.

The following Transfer CFTs are registered in Central Governance:

- CFT_PCA and CFT_Store_001
- CFT_Store_002b on `host.store.002`

User executes the command:

```
appImport --file c:/imports/impApp.json
```

The results of the import are:

- The three applications are imported: Product Catalog Application, Store_001 and Store_002.
- The Transfer CFT CFT_Store_002a was not found, but Transfer CFT CFT_Store_002b was found on the same host, `host.store.002`. CFT_Store_002b is linked to the application Store_002
- In the command output , a message reports that CFT_Store_002a was not found and that CFT_Store_002b was linked to the application.

flowImport

You can use this command to import to Central Governance a list of flows in a file. The file can be exported by another instance of Central Governance or generated with an external tool.

You also can import the applications defined in the file.

If the participants used in flows are not found in Central Governance or cannot be imported from the file, the flow is not imported unless the **allowincomplete** parameter is used.

Usage examples

The file `impFlows.json` contains a flow with name PL001 that has:

- As source the application, Product Catalog Application linked to Transfer CFT CFT_PCA.
- As target the application, Store_001 linked to Transfer CFT CFT_Store_001 and the application Store_002 linked to Transfer CFT CFT_Store_002.

The following Transfer CFTs are registered in Central Governance:

- CFT_PCA
- CFT_Store_001.

In example 1, two optional parameters are not used in the executed command: **importapplications** and **allowincomplete**. Examples 2-4 show what happens when the parameters are used.

Example 1

User executes the command:

```
flowImport --file c:/imports/impFlows.json
```

The results of the import are:

- The three applications are not imported.
- The flow PL001 is not imported. In the command output there is a message that applications Product Catalog Application, Store_001 and Store_002 cannot be added to the flow because they are not found in Central Governance.

Example 2. importapplications

User executes the command:

```
flowImport --importapplications --file c:/imports/impFlows.json
```

The results of the import are:

- The three applications are imported. In the command output for the application Store_002 there is a message that CFT_Store_002 was not found.

- The flow PL001 is not imported. In the command output there is a message that application Store_002 cannot be added to the flow because the CFT_Store_002 was not found.

Example 3. allowincomplete

User executes the command:

```
flowImport --allowincomplete --file c:/imports/impFlows.json
```

The results of the import are:

- The three applications are not imported. In the command output for the application Store_002 there is a message that CFT_Store_002 was not found.
- The flow PL001 is imported with status **Saved** and:
  - Source is empty
  - Target is empty
- In the command output there is a message that applications Product Catalog Application, Store_001 and Store_002 cannot be added to the flow because they are not found in Central Governance.

Example 4. allowincomplete and importapplications

User executes the command:

```
flowImport --allowincomplete --importapplications --file c:/imports/
impFlows.json
```

The results of the import are:

- The three applications are imported. In the command output for the application Store_002 there is a message that CFT_Store_002 was not found.
- The flow PL001 is imported with status **Saved** and with:
  - Source, the application Product Catalog Application is linked to the Transfer CFT CFT_PCA
  - Target, the application Store_001 is linked to Transfer CFT CFT_Store_001 and the application Store_002 is not linked to any products

Example 5. overwrite

After running and saving Example 4, there is a new version of the export file from another instance of Central Governance. The flow PL001 in the file is changed with target linked to only application Store_001.

User executes the command:

```
flowImport --overwrite --file c:/imports/ impFlows.json
```

The results of the import are:

- The three applications are not imported.
- The flow PL001 is imported and overwrites the existing flow PL001 flow with the status **Saved not deployed** and with:
  - Source, the application Product Catalog Application is linked to Transfer CFT CFT_PCA
  - Target, the application Store_001 is linked to Transfer CFT CFT_Store_001

Only the flow is overwritten in all cases. No applications are overwritten, even if you use the **importapplications** option in combination with **overwrite** mode.

partnerImport

You can use this command to import to Central Governance a list of partners in a file. The file can be exported by another instance of Central Governance or generated with an external tool.

Usage examples

The following examples illustrate using the command.

Example 1

The file `impPartners.json` contains definitions of the following partners.

- The partner BreadSupplier with a server communication profile HTTP
- The partner WineSupplier with a server communication profile SFTP

User executes the command:

```
partnerImport --file c:/imports/impPartners.json
```

The results of the import are:

- The two partners are imported and created: BreadSupplier and WineSupplier.
- Each partner has the correct server communication profiles imported, without SSL/TLS.

Example 2

The file `impPartners-v2.json` has a new version of partner WineSupplier, which now can also communicate with FTP over SSL/TLS.

User executes the command:

```
partnerImport --file c:/imports/impPartners-v2.json -o
```

The results of the import are:

- Partner WineSupplier is updated.
- A new server communication profile FTP in SSL/TLS mode with its credential of type certificate is added to WineSupplier.
- The existing communication profile with SFTP is still there.

Example 3

The file `impPartners-v3.json` has a new version of partner WineSupplier, which now can communicate only with FTP over SSL/TLS.

This partner is already used by the flow with SFTP in Central Governance.

User executes the command:

```
partnerImport --file c:/imports/impPartners-v3.json -o
```

The result of the import is the update of partner WineSupplier is not imported because the SFTP server communication is already used by a flow and cannot be removed.

policyImport

You can use this command to import a list of policies from a file. Reasons for importing policies are:

- Put policies in production after testing them in a testing or staging environment.
- Import policies already assigned to Transfer CFTs to overwrite and update their configurations.

Policies are determined to be unique by name. Transfer CFT assignments are retained, but status is affected depending on whether changes are detected:

- **No changes detected**. The original policy parameters have identical values as the imported policy, even if pin or lock replacements are detected. Policy status, parameters status and deploy status are not affected.
- **Changes detected**. Policy status becomes Saved, not deployed. The possible changes are: parameters that were originally disabled are now pinned or locked, values are updated, and parameters that were pinned or locked originally are now disabled.

Policies with incorrect values for at least one parameter are not imported.

Usage examples

The file `impPolicy.json` contains four policies with the following conditions in relation to the import target environment:

- The policy POLICY1 is new.
- The policy POLICY2 already exists and has no Transfer CFTs assigned.
- The policy POLICY3 already exists, has at least one Transfer CFT assigned, and is deployed.
- The policy POLICY4 has at least one invalid parameter.

Example 1

User executes the command:

```
policyImport --file c:/imports/impPolicy.json
```

The results of the import are:

- POLICY1 is imported and has the Saved status.
- The existing policies POLICY2 and POLICY3 are not imported. In the command output a message says each policy exists.
- POLICY4 is not imported. In the command output a message specifies the invalid parameters.

Example 2

User executes the command:

```
policyImport --overwrite --file c:/imports/impPolicy.json
```

The results of the import are:

- POLICY1 is imported and has the Saved status.
- The existing policies POLICY2 and POLICY3 are updated successfully. POLICY2 status is Saved and POLICY3 status changes to Saved, not deployed if changes are detected.
- POLICY4 is not imported. In the command output a message specifies the invalid parameters.

Certificates 5

Central Governance comes with default certificates for securing browser connections and communications with registered products. Best practice is to replace the default certificates with your own. The security service of the Central Governance Access and Security node supports CA services for generating custom certificates. The following topics describe the security service, CA services and methods for replacing and updating certificates.

Security service

The Security service of the Central Governance Access and Security node manages an internal PKI where SSL certificates are stored.

Access and Security:

- Stores the SSL certificates used between the Central Governance nodes and the user interface.
- Manages certificate authorities for certificate signing request (CSR) validation and signing as part of the product registration process.
- Performs chain building and validation for each certificate in use.
- Notifies of certificate expiration in advance so certificates can be replaced before functionality is affected.

The following describes options on menus used for security tasks in the Access and Security user interface. The menus are Security and Administration. Your user must be assigned to a specific role to perform security tasks in the UI (see *Roles for managing certificates* on page 102).

Security menu

The Security menu has options for all managed PKI objects.

Entities option

- Lists and manages all entities, which are password-protected containers of private keys.
- The entity trust level applies to all active certificates it contains;

Certificates option

- Lists and partially manages certificates.
- Active certificates are ready for use. Non-active certificates are viewed as drafts and cannot be selected.
- A certificate is considered trusted (root of a certificate chain) if it is marked as trusted or the entity is trusted.
- Versioning is supported for automatic replacement when the principal certificate expires. Under the same alias, multiple versions can coexist.

CA Services option

- Lists and manages certificate authorities, which are responsible for issuing certificates.

Certificate Signing Requests (CSR) option

- Lists CSRs that can be imported or generated. CA services validate and sign CSRs.
- A certificate obtained from a CSR is meant to belong to an entity or to another CA service.

Administration menu

Server Security Settings is the only applicable security option on the Administration menu. It is used for listing and delegating SSL certificates for various purposes.

CA services

The Central Governance Access and Security service supports CA services. When products like Transfer CFT register with Central Governance, a CA service has a private key for signing the SSL certificates used for securing connections between Central Governance and the products. Access and Security stores and manages all certificates.

There are two default CA services:

- PassPort CA has a certificate named PassPort Component CA for signing all certificates to secure transfers between the registering Transfer CFT and all other Transfer CFTs. This CA service corresponds to the business CA (see *Business certificate authority* on page 54).
- PassPort Product CA has a certificate named PassPort Product CA for signing all certificates to secure connections between the Central Governance internal nodes and the registering Transfer CFTs. This CA service corresponds to the governance CA (see *Governance certificate authority* on page 54).

You can change the CAs on the Central Governance configuration page (see *Configuration and startup* on page 47). Best practice is to replace the defaults with your own certificates.

If CAs change after Transfer CFT registration

After registering Transfer CFTs in Central Governance, changing any of the Central Governance certificate authorities requires resubmitting certificate registration:

1. Transfer CFT Copilot requests a new SSL certificate signed by the new CA.
2. Central Governance sends the requested certificate to Copilot.

Also see *CA services* on page 100 for more information.

Governance CA

Changing the governance CA affects registered Transfer CFTs. You must import the new CA in Transfer CFT and schedule the certificate registration.

Transfer CFT 3.1.2

For Transfer CFT 3.1.2, stop Copilot and Transfer CFT and do the following:

Replace the PassPort CA by running the following Transfer CFT command:

```
PKIUTIL PKICER ID = 'PassPortCA', ROOTCID = 'PassPortCA', ITYPE =
'ROOT', INAME = '<GovernanceCACertificateFile>', IFORM = 'DER', MODE
= 'REPLACE'
```

Then trigger the certificate registration by resetting the `cg.registration_id` to **-1** with the following command:

```
CFTUTIL UCONFSET ID=cg.registration_id, VALUE=-1
```

Restart Copilot.

Transfer CFT 3.1.3

For Transfer CFT 3.1.3, import the new CA by doing one of the following:

- Configure the CA by setting the CA Certificate by using the installer for Transfer CFT in configure mode. You must stop Copilot and Transfer CFT before starting the installer in configure mode. You can run the configure command in the Transfer CFT installation directory to start the installer in configure mode.

  or

- If you do not want to stop Transfer CFT, use the following commands:
  - `PKIUTIL PKICER ID = '<CG CA new alias>', ROOTCID = '<CG CA new alias>', ITYPE = 'ROOT', INAME = '<GovernanceCACertificateFile>', IFORM = 'DER', MODE = 'CREATE'`
  - `CFTUTIL UCONFSET ID=cg.ca_cert_id, VALUE='<CG CA alias>,<CG CA new alias>'`

Then set the parameter `cg.certificate.governance.renewal_datetime` (format: YYYYMMDDHHMMSS + GMT) to schedule the request at first heartbeat after the specified date and time.

The heartbeat interval is specified in seconds in the `cg.periodicity` parameter (default value 600). For example, schedule the certificate request to start December 23, 2014, at 14:30:00 + GMT by running the following command:

```
CFTUTIL UCONFSET ID=cg.certificate.governance.renewal_datetime,
VALUE=20141223143000
```

Transfer CFT becomes unreachable until the new certificate is received.

Business CA

If the business CA is changed, a new business SSL certificate can be requested. The new certificate is signed by the new CA and used in secured flows.

Schedule a new certificate request starting with the time specified in the Transfer CFT parameter `cg.certificate.business.renewal_datetime` (format: YYYYMMDDHHMMSS + GMT).

Make sure the new business CA is known by all Transfer CFT flow partners before the certificate is renewed.

Roles for managing certificates

Users who perform certificate management need appropriate roles for working in the Access and Security user interface. They can be associated with a system administrator role or a certificate role with narrower privileges.

Add system administrator role

Use this procedure to add a system administrator role for Access and Security. This role enables associated users to perform all functions.

1. In Central Governance, select **Access > Roles** to open the Roles page in Access and Security.
2. Select the PassPort role **System administrator** and click **Copy**.

3. Click **Paste** and type a unique name for the copied role. For example, **AS system admin**. Click **OK** to add the role.
4. Go to *Assign role* on page 103.

Add certificate role

Use this procedure to add a certificate role for Access and Security. This role enables associated users to perform certificate management functions.

1. In Central Governance, select **Access > Roles** to open the Roles page in Access and Security.
2. Click **New Role** to open the New Role wizard.
3. Do the following to add the role.
    a. On the General Information page, type a unique name for the role. For example, **AS certificate manager**. Entering a description is optional. Leave **Active** as the status. Click **Next**.
    b. On the Select Privileges page, select the following PassPort Available Privileges and add them to the Selected Privileges section:

    Manage CA Services
    Manage certificates and ssh keys
    Manage entities
    Manage server settings
    View domains, organizations and users

    c. Click **Finish** to add the role.
4. Go to *Assign role* on page 103.

Assign role

Use this procedure to assign a role to a user in Central Governance.

1. In Central Governance, select the **Access** tab.
2. Select a user and click **Select roles**.
3. Select a role and click **Apply**.

If the role enables permissions for Access and Security and the user already has a UI session running for the service, the session must be refreshed or reopened for the role to become effective.

Replace SSO certificate

Central Governance provides a default certificate for securing browser connections for SSO. This certificate is signed by a default CA. You can replace the default CA with a trusted CA, which automatically customizes the SSO certificate and all SSL certificates used by Central Governance. See *CA services* on page 100 for information on replacing the CA.

Alternatively, you can replace only the SSO certificate with a custom certificate. The replacement must not be a self-signed certificate. HSTS rejects self-signed certificates. Use the following procedure to change the SSO certificate.

Prerequisites

- A user needs permissions to manage certificates to replace the SSO certificate. See *Roles for managing certificates* on page 102 for details.
- Make sure the replacement certificate file is available on the file system. The certificate file must contain a public-private key pair.

Add entity

1. In Central Governance, select **Access > Roles** or **Access > Privileges** to open the Roles or Privileges page in the Access and Security user interface.
2. Select **Security > Entities** to open the Entities page. You are going to add an entity for the new SSO certificate.
3. Click **New Entity** to open the Create Entity window. Do the following:
    a. Type a unique name for the entity.
    b. Select **Synchrony** as the domain.
    c. Type a password. This must be used in future whenever you change the entity contents.
4. Click **OK** to add the entity.

Import certificate

5. On the Entities page, click the name of the entity to open its details page.
6. In the Certificates section, click **Import** to open the Import Certificate window. Do the following:
    a. Type the password of the entity.
    b. Type an alias to identify the certificate to be imported. For example, **SSO SSL cert**.
    c. Select the P12 file to import.
    d. Type the password for the file.
    e. Click **OK** to import.
7. Once imported, select the **Active** check box on the entity details page.
8. If available, import the public issuer certificate or certificate chain and trust the root. Alternately, the SSL certificate can be trusted directly, and certificate validation does not include the rest of the chain.
9. Click **Save** and **Cancel** to save and close the entity details page.

Replace certificate

10. Select **Administration > Server Security Settings** to open the Server Security Settings page.
11. Scroll down the page and find **Default_SSO (Entity: SSL)**.
12. Click **Change** to open the Change HTTPS Certificate wizard.
13. Select the entity where the replacement certificate is located and click **Next**.
14. Select the replacement certificate and click **Next**.
15. Type the entity password and click **Finish**.

Update SSO certificate before expiration

You can replace the SSO certificate before it expires by changing the governance CA. See *CA services* on page 100 for more information.

Alternately, use the following procedure to update the SSL certificate for SSO before its expiration date.

For uninterrupted HTTPS connections, you can open the SSL entity, or the entity with the current SSL certificate, and import a version of the SSL certificate. The version becomes effective when the current SSL certificate expires.

This procedure applies not only to updating the SSO certificate before expiration, but to any certificate used for SSL communications that is stored in an entity.

1. Make sure your user account has a role with privileges for managing certificates. See *Roles for managing certificates* on page 102.
2. In Central Governance, select **Access> Roles** or **Access > Privileges** to open the Roles or Privileges page in the Access and Security user interface.
3. Select **Security > Entities** to open the Entities page.
4. Open the SSL entity, or the entity with the current SSL certificate, and import or generate a version of the current SSL certificate. The password of the SSL entity is **ssl**.
5. Mark the certificate as **Active**.

If the new SSL certificate version has a different issuer or is self-signed, either its root or the certificate itself must be marked as trusted to be validated properly.

Certificates for HTTP, FTP, PeSIT

This topic describes the credentials used by participants in server and client communication profiles for SSL mutual authentication in flows using HTTP, FTP and PeSIT.

To achieve mutual trust, each party must have and trust the CA of the other party. However, in some cases, one party might decide to trust an intermediate CA rather than the root certificate, or even directly the end-user certificate, which is not good practice.

Partners and unmanaged products provide only the public part of their certificate or key for mutual trust. When importing a new certificate, you can select one of the following options:

- Import only the root CA, which is a single certificate.
- Import the full public chain, which includes the root self-signed CA, the intermediate CAs and the end user SSL certificate.
- Import a public sub-chain, which starts with an intermediate CA followed by other intermediate CAs and the end user SSL certificate.

When Central Governance deploys configurations, the registered products must receive certificates, which include private keys, representing their own certificates to be used for SSL in flows. When importing a certificate, you must import the full chain of the SSL private certificate. However, you can import a self-signed certificate for SSL for testing purposes.

Object	Has private key	File format
Middleware client communication profiles (in flows)	Yes	Private chain of certificates: PKCS#12 (*.p12) password protected Self-signed private SSL certificate: PKCS#12 (*.p12) password protected
Middleware server communication profiles (on the static configuration page)	Yes	Private chain of certificates: PKCS#12 (*.p12) password protected Self-signed private SSL certificate: PKCS#12 (*.p12) password protected
Partner client communication profiles (in flows)	No	Single public certificate: DER (*.der, *.cer), PEM (*.pem) Public chain of certificates: PKCS#7 (*.p7b)
Partner server communication profiles (on the partner page)	No	Single public certificate (CA): DER (*.der, *.cer), PEM (*.pem) Public chain of certificates: PKCS#7 (*p7b)
Unmanaged product PeSIT SSL certificate	No	Single public certificate (CA): DER (*.der, *.cer), PEM (*.pem) Public chain of certificates: PKCS#7 (*p7b)

The following are guidelines for managing certificates.

- You can upload a new certificate or select among existing ones.
- When uploading a new certificate, you must provide an alias.
- The alias must be unique at the participant level. For instance, if certificates are uploaded for a Partner, the alias must be unique at the Partner level (no matter if it is uploaded from the Partner page or from the flow page in a client communication profile).
- If the same certificate exists and it is used by another Partner or Middleware respectively, it will be linked to the existing one while keeping the newly provided alias as a reference. In other words, two aliases can point to the same certificate content
- Private certificates always require a password. This is also mandatory in order to display the certificate information
- Certificates are imported in the Access and Security Public Key Infrastructure (PKI)
- You cannot import the same certificate both for a Partner (only the public part) and a Middleware (the private part as well). This will end up in error.
- At selection, certificates are validated for their trust and validity before returned. Selection may happen when:
    - certificates are deployed during flow deployment
    - certificates are viewed on the Partner/Middleware/Flow page
    - Partner/Middleware is removed, so certificates are also removed along with it.
- When importing a certificate or a chain of certificates, the top-most will be imported as trusted. This influences how the certificate chain is constructed. A counter-example of what should not be done is:
    - Have a Partner1 with chain: CARoot -> CAInt -> Partner1.
    - Have a Partner2 with chain: CARoot -> CAInt -> Partner2.
    - Import on the Partner1 the chain: CARoot -> CAInt -> Partner1 => CARoot will be imported as trusted.
    - Import on the Partner2 the chain: CAInt -> Partner2 or just CAInt => CAInt will be imported as trusted. This will not only impact Partner2 but also Partner1.
    - Next time selection for Partner1 is done, the chain will stop at CAInt because it is the first certificate to be found as trusted, while it is recommended that the root-most be trusted.
    - The recommendation in this case is:
        - Either import only CARoot for both Partner1 and Partner2 (it is sufficient that the Middleware exchanging with this Partner know only the Root-most CA).
        - Or, import the full chain for both Partner1 and Partner2.
- When importing a certificate, make sure it is not expired. Central Governance will reject an expired certificate.

Keys for SFTP

This topic describes the credentials used by participants in server and client communication profiles for SSH key authentication in flows using SFTP.

For keys there is no chain of certificates, but a public-private key pair. For SSH the client might ask for the server key verification based on fingerprint or key content.

When the client authenticates with a public key, the client must be provided a private key while the server must have the public key corresponding to the client's private key.

Just as for certificates, partners only deal with the public SSH keys. However, products need the private key to use for authentication as clients or the private key to serve as SSH key for the server. The private key is on the partner's side and is not managed by Central Governance.

Object	Has private key	File format
Partner server communication profiles (on the partner page)	No	public key: DER (.der), PEM (*.pem)
Partner client communication profiles (in flows)	No	public key: PEM (*.pem)
Middleware server communication profiles (on the static configuration page)	Yes	private key: PKCS#8 (*.p8) password protected
Middleware client communication profiles (in flows)	Yes	private key: PKCS#8 (*.p8) password protected

The following are guidelines for managing keys.

- You can upload a new key or select among existing ones.
- When uploading a new key, you must provide an alias.
- The alias must be unique at the participant level. For instance, if keys are uploaded for a Partner, the alias must be unique at the Partner level (no matter if it is uploaded from the Partner page or from the flow page in a client communication profile).
- If the same key exists and it is used by another Partner or Middleware respectively, it will be linked to the existing one while keeping the newly provided alias as a reference. In other words, two aliases can point to the same key content (although functionally it should not be the case in production).
- Private keys always require a password. This is also mandatory to display the key information
- Keys are imported in the Access and Security public key infrastructure (PKI).
- You cannot import the same key both for a Partner (only the public part) and a Middleware (the private part as well). This will end up in error.

User management 6

The following topics are about managing users in Central Governance.

Managing users is limited to users who are assigned to roles with the Central Governance Manage User privilege or a user-defined privilege with similar properties. If the Access tab is not available on the top toolbar, you cannot manage users.

List users

The User List page displays a list of all users managed by Central Governance. Once the list is displayed, you can add users and perform maintenance such as changing or removing users. This topic provides cross-references for related user-management actions.

Managing users is limited to users who are assigned to roles with the Central Governance Manage User privilege or a user-defined privilege with similar properties.

Click **Access** on the top toolbar to open the User List page. You can perform the following tasks.

Add user

Click **Add user** to add a user. See *Add a user* on page 110.

View user details

Click a user ID to open a page where you can review user details or edit or remove the user. See *View, edit, remove a user* on page 112.

Assign or unassign roles

Select one or more users and click **Select roles** to change assigned roles. See *Roles and privileges* on page 114 and *Manage roles* on page 116.

Unlock user

Select one or more locked users and click **Unlock** on the User List page. You also can click the name of a locked user on the User List page and click **Unlock** on the user details page. See *User lockouts* on page 113.

Remove

Select one or more users and click **Remove** to remove from Central Governance.

Add a user

Use this procedure to add a user in Central Governance. Managing users is limited to users who are assigned to roles with the Central Governance Manage User privilege or a user-defined privilege with similar properties.

Note Users are unique by organization and not globally by user ID. This means two users can have the same user ID provided they belong to different organizations.

Steps

1. Click **Access** on the top toolbar to open the User List page.
2. Click **Add user** to open the Add User page.
3. Complete at least the required fields.

   You must select an organization for the user. If the organization you want is not available, cancel, add the organization and then add the user. See *User organizations* on page 119 for details.

   Specifying an email address is optional. However, Central Governance can send notifications to the user only if you provide a valid email address. If you add an email address, Central Governance notifies the new user of the URL for the log-on page and credentials for logging on. Without an email address, you must notify the new user yourself.

   You should select a role for the user. Users must be assigned to a role to enable them to perform actions. Users without roles can log on to Central Governance, but can access only the Help Center tab. See *Roles and privileges* on page 114 for information.
4. Click **Save user** to add the user.

Notifying user of new account

After adding a user, what happens next depends on whether you specified a valid email address for the user.

With email address

If you specified an email address, Central Governance notifies the user of the URL for connecting to the log-on page and credentials for logging on. A randomly generated password is provided that the user must change after logging on the first time. This information is contained in two email messages to the user. The first contains the organization name and user ID. The second contains the password and the link to the Central Governance log-on page.

If the email address is valid but Central Governance cannot send messages because of an SMTP server connection error or other problem, the user is added, but the password is set to Initial01.

If the email address is invalid, the user is not notified. The user must inform the Central Governance administrator of failure to receive credentials. The administrator must enter a correct email address for the user and inform the user of their organization. The user then must use the forgot password feature on the log-on page to receive a password.

Without email address

If you did not specify an email address, inform the user they have been added to Central Governance and provide:

- The user ID and password Initial01. The password is valid for logging on the first time only. The system prompts the first-time user to change the password.
- The name of the organization associated with the user. The user must select this organization when logging on.
- The URL for connecting in a browser to Central Governance.

Customizing email templates

Central Governance has email template files with content you can use as-is or customize. The templates are the models for email messages the server sends to users with valid email addresses in their Central Governance user accounts.

The template files are at `<install directory>\data\mail`. The following table lists the templates and their uses.

Template file	Description
UserAccountCreation1_en	First message sent to new user. It contains the user's organization name and user ID.
UserAccountCreation2_en	Second message sent to new user. It contains the user's password and a link to the Central Governance log-on page.
UserAccount_Locked_en	Message informs user their account has been locked after the user exceeded the allowed number of consecutive attempts to log on with an invalid password. See *User lockouts* on page 113 for more information.
UserAccount_Unlocked_en	Message informs user their account has been unlocked.
UserPassword_Reset_en	Message contains a new password for a user who forgot their password. See *Password recovery* on page 114 for more information.

The templates use a combination of text and variables. The variables are replaced with values before the server sends messages. For example, the variable `%FirstName%` is replaced with the user's first name. The variables you can use are listed and defined in the template files.

The templates contain messages in HTML and plain-text formats. This is for email clients that support plain text but not HTML. When customizing messages, make sure to make identical changes for both formats.

After editing templates, restart Central Governance for the changes to become effective.

You can customize the content of the following parts:

Section	Description
subject	Message subject line
content.html.body	Message in HTML format
content.text.body	Message in plain-text format

The default sender of the messages is **cg.noreply@axway.com**, an invalid email address. You can change the sender to a different valid or invalid address. To change the address, edit the value of the `mail.sender.address` property in the `com.axway.cmp.mail-default.cfg` file at `<install directory>\runtime\com.axway.nodes.ume_<UUID>\conf`.

View, edit, remove a user

Use this procedure to view, edit or remove a user in Central Governance. Managing users is limited to users who are assigned to roles with the Central Governance Manage User privilege or a user-defined privilege with similar properties.

Note To add a user see *Add a user* on page 110.

View user

1. Click **Access** on the top toolbar to open the User List page.
2. Click the name of the user to open the details page for the user. You can view the user name, organization, email address, user ID, roles and address.

Edit user

Click **Edit** to open a page for editing the user details. If you need information about changing roles, see *Roles and privileges* on page 114.

A user editing their own account cannot change the organization or user ID. Only another user with user management rights can make such changes.

Remove user

Do one of the following to remove a user:

- On the User List page, select one or more users and click **Remove**.
- On the User List page, click the name of a user to open the details page. Click **Remove**. Or, click **Edit** and then click **Remove**.

User lockouts

Users who make repeated, consecutive attempts to log on with invalid passwords are blocked from logging on even with valid passwords. This occurs after users exceed the allowed limit of unsuccessful attempts. Locked-out users can log on again only after an administrator unlocks their accounts.

By default users can make three consecutive attempts to log on with invalid passwords before the lock-out engages. The number is configurable.

Lock-outs affect only users managed by Central Governance. Users on external LDAP identity stores are not affected.

Users with valid email addresses associated with their accounts receive messages when they are locked out. Users without email addresses do not receive notifications.

Unlock users

Lock-outs do not expire. Users are locked-out until an administrator unlocks them.

Users who are locked out are identified by a lock icon on the User List page under Access > Users.

To unlock, select one or more locked users and click **Unlock** on the User List page. You also can click the name of a locked user on the User List page and click **Unlock** on the user details page.

When a user is unlocked:

- If an email address is defined for the user, the user receives a message containing a new randomly generated password. If Central Governance cannot send the user a message because of an SMTP server connection error or other problem, the user is unlocked but the password is set to Initial01.
- If an email address is not defined for the user, the user's password is reset to Initial01.

Configure lock-out threshold

You can change the number of consecutive times a user can fail to log on with an invalid password before being locked out. To change the threshold, edit the value of the `number.accepted.failures` property in the `com.axway.cmp.participant-default.cfg`

file at `<install directory>\runtime\com.axway.nodes.ume_<UUID>\conf`.

Password recovery

Central Governance enables users with valid email addresses who have forgotten their passwords to get new ones. This only applies to users managed by Central Governance.

A user who clicks the **Forgot password** link on the log-on page is prompted to select their organization and enter their user ID. After submitting, there can be different results.

- If the user has a valid email address defined in their account and is not locked out, the user receives a message containing a new randomly generated password.
- If Central Governance cannot send a message to a user with a valid email address because of an SMTP server connection error or other problems, a new password is not set.
- If the user has an invalid email address, no address or is locked out, an error message displays and a new password is not set.
- If the user is managed by an external LDAP identity store, an error message displays and a new password is not set. Central Governance cannot reset passwords of such users.

Roles and privileges

Roles and privileges grant or limit users' permissions to perform actions and ordain the areas of the user interface they can access.

Roles

A role is based on one or more privileges, and a privilege is based on a resource. There are two types of roles: predefined and user-defined. Predefined roles are available by default to assign to users. User-defined roles are custom roles that an administrator creates. Predefined roles cannot be changed or deleted, but you can copy and rename them, and the copies can be managed just like user-defined roles.

Users can be assigned to one or more roles. Typically, users with multiple roles have more privileges than users with fewer roles. However, you could build a single role that grants unlimited privileges.

Users must be assigned to a role to enable them to perform actions. Users without roles can log on to Central Governance, but can access only the Help Center tab.

Default roles

Central Governance has default roles that grant different levels of access. Users might have full or partial access to features on the tabs enabled by their assigned roles.

A user with user management authority can assign or unassign roles when adding or editing users in the Central Governance user interface. However, you must access the Access and Security UI to view role details and add or edit roles. See *Manage roles* on page 116 for how to display a list of Central Governance roles and descriptions.

The following describes the default Central Governance predefined roles.

CG Admin

This administrator role grants users unlimited access to all areas of the Central Governance user interface.

Access Manager

- Access tab - Full access.
- Administration tab - Full access to services, but no access to deployments, audit or web dashboards.

IT Manager

- Products tab - Full access.
- Alert Rules tab - Full access.
- Access tab - Full access.
- Administration tab - Full access to services, but limited access to deployments. The user can view and deploy or redeploy configurations, but not flows. The user also can view and redeploy policies. In addition, this user can run and customize the IT Manager dashboard and reports, run the Audit report and monitor product updates.

Middleware Manager

- Products tab - Partial access to products and unmanaged products, and no access to policies and product updates. Role allows viewing product details and configuration, but not editing.
- Applications tab - Full access. This is the only role that gives access to the Applications tab.
- Flows tab - Full access. This is the only role that gives access to the Flows tab, which is for managing and monitoring flows. The user can run the Flows Report.
- Alert Rules tab - Full access.
- Administration tab - No access to services and audit, but limited access to deployments. The user can deploy or redeploy flows, but not policies and configurations. In addition, this user can run and customize the Middleware Manager dashboards and reports.

Partner Manager

- Applications tab - Full access.
- Partners tab - Full access.
- Access tab - Access to Organization List page. The user can view, add, edit and remove organizations.

Default roles of registered products

Axway products that register in Central Governance also can have default predefined roles. You can do the following to view details of default roles for registered products.

1. Select **Access > Roles** on the top toolbar in the Central Governance user interface to open the Access and Security Roles page.
2. Click **Search**, type the name of a product and click **Go** to display the roles for the product.
3. Click the name of a role to open its details page. The details include the privileges in the role.

Privileges

Privileges give users authorization to access and perform actions in the user interface. Privileges are assigned to roles that in turn are assigned to users.

Central Governance has a number of predefined privileges. Predefined means the privileges are available by default to assign to roles. It also means the privileges cannot be changed or deleted. You can, however, make a copy of a predefined privilege and edit the copy. See *Manage privileges* on page 118 for how to display a list of Central Governance privileges and descriptions.

Privileges are based on resources, and a single privilege is based on a single resource. Each resource has available actions. The privilege inherits the actions, which can be enabled or disabled individually.

For example, Central Governance has a predefined privilege named Manage User. The privilege is built on a resource named User. The User resource has the following actions: view, create, modify, delete, assign, reset. All of these actions are enabled in the predefined privilege.

Although you cannot change the Manage User predefined privilege, you can make a copy and enable only some actions. For example, in the copy you can enable only the view action, which enables viewing users in Central Governance, but forbids all other user-management actions, including adding and deleting.

Manage roles

Use this procedure to view a list of Central Governance roles and descriptions and perform other tasks.

Note This topic is an introduction to managing roles in the Access and Security UI. For more details, select **Help > Topic Help** or **Help > Help** in the Access and Security UI.

View roles, descriptions

There are two ways to view roles and descriptions.

Central Governance user interface

Click **Access** on the top toolbar to open the User List page. Select a user and click **Select roles** to open a pop-up that lists available roles. Place your cursor over a role to display a description of it. You can assign or unassign roles or click **Cancel** when done viewing.

You can open the same pop-up listing available roles when adding or editing a user.

Access and Security user interface

Select **Access > Roles** to open the Roles page in the Access and Security UI. If the list of roles exceeds one page, use the paging controls at bottom right.

Click the **Product** heading to sort the list of roles by product. Move the cursor over the descriptions in the Description column to display the full descriptions of roles. Descriptions are optional and only display when available.

A blank Product field identifies a user-defined role that contains privileges for multiple products.

There are two types of roles: predefined and user-defined. Only user-defined roles can be edited or deleted. Predefined roles cannot. But you can copy a predefined role, rename it and edit the copy. The Central Governance default roles are predefined; see *Default roles* on page 114 for a list.

Click the name of a role to open its details page. Notice that a role can contain privileges and sub-roles.

Add, edit, delete a user-defined role

Only user-defined roles can be added, edited or deleted in the Access and Security UI. But you can copy a predefined role, rename it and edit the copy. Select **Help > Help Topic** in the Access and Security UI for details about actions for managing roles.

Click the name of a user-defined role to open its details page and perform changes.

Any role you add is a user-defined role and is added to the list of roles that can be assigned to users in Central Governance. Conversely, any user-defined role you delete is no longer available to assign. Moreover, if you delete a role that has been assigned, it is removed from the users.

For example, select the CG Admin predefined role and click **Copy** and then **Paste**. Enter a name when prompted for the copied role (for example, **CG Admin Copy**). This copied role is now a user-defined role. Return to the Central Governance UI and click **Access** on the top toolbar. Select any user and click **Select roles**. Notice the CG Admin Copy role is listed as an available role to assign.

Manage privileges

Use this procedure to view a list of Central Governance privileges and descriptions and perform other tasks in the Access and Security user interface.

Note This topic is an introduction to managing privileges in the Access and Security UI. For more details, select **Help > Topic Help** or **Help > Help** in the Access and Security UI.

Select **Access > Privileges** to open the Privileges page in the Access and Security UI. If the list of privileges exceeds one page, use the paging controls at bottom right.

View privileges, descriptions

Click the **Product** heading to sort the list of privileges by product. You can identify the privileges for Central Governance or any product by the Product column.

There are two types of privileges: predefined and user-defined. Predefined privileges are available by default. User-defined privileges are added by users.

Most predefined privileges have descriptions. However descriptions are not required, and some predefined and user-defined privileges might not have descriptions.

By default Central Governance has only predefined privileges. You can view details of predefined privileges, but you cannot edit or delete them.

Click the name of a privilege to open its details page. The name of the resource on which the privilege is based is displayed in the Resource field. The available actions for the privilege are displayed below the resource name.

Add, edit, delete a user-defined privilege

A privilege created by a user is a user-defined privilege. Only user-defined privileges can be edited. But you can copy a predefined privilege, rename it and edit the copy, which becomes a user-defined privilege. The Type column identifies privileges as predefined or user-defined. Select **Help > Help Topic** for details about adding a user-defined privilege.

Click the name of a privilege to open its details page. Select **Help > Help Topic** for details about the changes you can make to user-defined privileges.

If you delete a user-defined privilege that has been added to a role, it is removed from the role.

About the Default User

Central Governance has a single Default User with privileges to log on to the user interface when the system is started the first time. This user can add other users.

The user ID of the Default User is admin@first.use. The initial password is Initial01, but must be changed when the user logs on the first time. The user has the Access Manager role, which grants privileges for managing users on the Access tab in the user interface.

The Default User can be edited just like any other user. You can change its name, user ID, roles, and so on. However, there is a safeguard for restoring the user to its original configuration should the need arise. For details see the repair parameter in *cgcmd command* on page 74.

Password policy

Users managed by Central Governance are subject to its password policy.

New users are given a temporary password. Central Governance forces users to change it the first time they log on. Thereafter, passwords do not expire.

Each password must have:

- 8 characters minimum
- At least 1 lower-case letter
- At least 1 upper-case letter
- At least 1 numeric character
- Not be equal to user ID
- Not be equal to initial password

In addition, passwords are case sensitive and can be any combination of:

- Upper- and lower-case alpha characters
- Numeric characters
- Special characters

User organizations

An organization is an object containing users who are managed by the same identity store.

Each user must be associated with an organization. The default in Central Governance is to assign users to Org. This organization uses the Central Governance internal identity store.

An organization must have a unique name and be associated with the internal identity store or an external identity store on an LDAP server. Optionally, an organization can have a description and an address and phone number.

Note Users are unique by organization and not globally by user ID. This means two users can have the same user ID provided they belong to different organizations.

Manage organizations

Use the following procedures to manage organizations.

If you use an external identity store

If you associate an organization with an external identity store and are mapping roles between Central Governance and the external LDAP server:

- Central Governance adjusts when an LDAP server has pagination enabled for roles. Central Governance can page through all pages and select all roles as needed. For example, if the LDAP server has 1,000 roles per page, Central Governance iterates through all pages.
- When Central Governance encounters an LDAP server with more than 100,000 roles, it returns an error message and does not display any of the LDAP roles in its user interface. This requires adjusting the filter in the LDAP server to return fewer roles.

View list of organizations

Select **Access > Organizations** to open the Organization List page. The page lists all organizations, their identity stores and number of users associated with the organizations.

You can:

- Click the name of an organization to view or edit its details.
- Add or remove an organization.

Add organization

When adding an organization, you must associate it with the internal Central Governance identity store or an external identity store on an LDAP server. Using the internal identity store means Central Governance manages users and their roles and credentials. Using an external identity store means users are managed by an LDAP server.

To associate an organization with an external identity store, you first must add the identity store in the user interface. See *Use Identity Store List page* on page 134 for details.

1. Select **Access > Organizations > Add organization** to open the Add Organization page.
2. Complete at least the required fields.

   If you select the internal identity store, no other action is required except to save the organization.

   If you select an external identity store, you must map internal Central Governance user roles with roles on the LDAP server. Click **Map roles** to open the Role Mapping page. The top left of the page lists the available internal roles. The bottom half of the page lists the available roles on the LDAP sever.

Select an internal role and the available external roles you want to map to it. Use the add and remove buttons to map or unmap. Click **Apply** when done.

3. Click **Save organization** to add it.

If you associated the organization with an external identity store and mapped internal and external roles, the LDAP users can log on with their LDAP credentials to Central Governance. See *Log on as LDAP user* on page 141 for more information.

View, edit organization

1. Select **Access > Organizations** to open the Organization List page.
2. Click the name of an organization to view its details. If the organization is associated with an external identity store, click **View role mapping** to review mapping of internal and external roles. Move the cursor to the top right of the Role Mapping page and click **X** to close it.
3. Click **Edit** on the details page.
4. Enter changes as needed. If the organization is associated with an external identity store, you can change identity stores or role mapping.

   You cannot change the identity store of Org, but you can change other details (description, address). Org uses the Central Governance internal identity store.
5. Click **Save changes**.

Remove organization

Do one of the following to remove an organization:

- Select **Access > Organizations** to open the Organization List page, select one or more organizations and click **Remove**.
- Select **Access > Organizations** to open the Organization List page, click the name of an organization to open its details page and click **Remove**.

If any users are associated with the removed organization, those users also are removed.

You cannot remove the default Org organization.

Fine-grained access control 7

Central Governance supports fine-grained access control (FGAC) to manage instances of objects specified users can view or change in the Central Governance user interface or when using CLI. Central Governance supports FGAC for:

- Applications, application groups, products, product groups and flows
- Dashboards and reports generated by the Visibility service

You can set up FGAC-enabled privileges in the Access and Security UI. Once created, you also use the Access and Security UI to assign the privileges to roles. Lastly, you assign the roles to users in the Central Governance UI.

The following are examples where FGAC might be useful:

- Differentiate between file transfers that are managed externally and internally.
- Differentiate access to managed file transfers by region or business unit.
- Give access to application groups to specific users. For example, if you have group A and group B, you can grant specific users access to group A and other specific users access to group B.

Objects, resources and actions for FGAC

The following table describes:

- The Central Governance objects you can manage with FGAC.
- The technical name of the resources to use in privileges for enforcing FGAC.
- The name of the product that owns the resource. The Visibility service is based on the Sentinel product, which is the name used in the Access and Security user interface.
- The resource properties that can be set as conditions for using FGAC privileges.
- The actions each resource supports. You can enable one or more actions per resource.

Object	Resource	Resource owner	Resource property	Resource property description	Resource actions
Application	Application	Central Governance	Name	Application name	View, create, modify, delete
Application group	Application Group	Central Governance	Name	Application group name	View, create, modify, delete

Object	Resource	Resource owner	Resource property	Resource property description	Resource actions
Flow	Flow	Central Governance	Name	Flow name	View, create, modify, delete, deploy
Product	Product	Central Governance	Name	Product name	Logs, stop, start, delete, modify, view
Product group	Product Group	Central Governance	Name	Product group name	Start, logs, stop, delete, modify, view, create
Dashboards	HTML Dashboard	Sentinel	Name	Dashboard name	Only the view action is used. The view design and manage actions are not used.
Reports	HTML Report	Sentinel	Name	Report name	View

The resource actions have the following meanings for supported actions on objects:

- **View** is permission to view objects.
- **Create** is permission to add objects.
- **Modify** is permission to edit objects.
- **Delete** is permission to remove objects.
- **Logs** is permission to view product logs.
- **Start** is permission to start products.
- **Stop** is permission to stop products.
- **Deploy** is permission to deploy flows to registered products.

Here's how you can view resources for products in the Access and Security UI. In the Central Governance UI, select **Access > Roles** or **Access > Privileges** to open the Access and Security UI. Then select **Administration > Products** to open the Access and Security Products page. Click the name of a product to open its details page. Click the **Resources** tab to view the resources for the product. Click the name of a resource to view details about it.

FGAC-enabled predefined privileges

The Central Governance Access and Security service has the following predefined privileges that are based on resources that support FGAC. You cannot edit conditions of predefined privileges. But you

can make copies of the predefined privileges, which makes the copies user-defined privileges, and then customize actions and set the name property in the privilege condition editor. The name property is the key to FGAC support.

You also can create your own user-defined privileges based on FGAC-enabled resources. You can only enable FGAC for user-defined privileges and not predefined privileges.

Predefined privilege	Privilege owner	Resource	Enabled actions
Administrate Product	Central Governance	Product	Logs, stop, start, delete, modify, view
IT Manager - Execute reports in Web Dashboards	Sentinel	HTML Report	View
IT Manager - Execute dashboards in Web Dashboards	Sentinel	HTML Dashboard	View
Manage Application	Central Governance	Application	View, create, modify, delete
Manage Application Group	Central Governance	Application Group	View, create, modify, delete
Manage Flow	Central Governance	Flow	View, create, modify, delete, deploy
Manage Product Group	Central Governance	Product Group	Logs, stop, start, delete, modify, view
Middleware Manager - Execute dashboards in Web Dashboards	Sentinel	HTML Dashboard	View
Middleware Manager - Execute reports in Web Dashboards	Sentinel	HTML Report	View
View Application	Central Governance	Application	View
View Application Group	Central Governance	Application Group	View
View Flow	Central Governance	Flow	View
View product	Central Governance	Product	View

The name condition for the Application resource applies to individual applications, but the name condition for the Application Group resource applies to all applications within the group. For efficiency, you could group applications and then have a single Application Group privilege. The specified name condition in the privilege would apply to all applications within the group.

Steps to enable FGAC

Use the Access and Security service user interface to manage FGAC privileges and associated roles. In the Central Governance UI, select **Access > Privileges** to open the Access and Security Privileges page.

The following are guidelines for enabling FGAC.

1. Create a user-defined privilege that is based on a FGAC resource. See *Objects, resources and actions for FGAC* on page 122.
2. When adding or editing a user-defined privilege, use the condition editor to set a value or property for the **name** resource property.
3. Once the privilege is configured, associate it to one or more roles.
    - For users managed by an LDAP identity store, map custom roles to external LDAP role definitions (user groups or roles) in the external organization.
    - For users managed by Central Governance, assign the privilege to one or more user-defined roles.

See the Access and Security help for details about managing privileges and roles. In the Access and Security UI, select **Help > Help** and see the following topics:

- Access menu > User privileges
- Access menu > User roles

The Central Governance documentation also has topics about roles and privileges managed in the Access and Security UI. See *Roles and privileges* on page 114.

Guidelines for creating FGAC privileges

The following tables provide guidelines for creating user-defined privileges that are FGAC enabled.

Any FGAC-enabled object

The following table provides guidelines for creating user-defined privileges for any FGAC-enabled object. See *Objects, resources and actions for FGAC* on page 122 for the names of FGAC-enabled objects and their related resources.

If you want to	You need a privilege with	A user role with the privilege can
Create objects	View and Create actions enabled for the object's resource and FGAC filter conditions set on the name property in the privilege.	View and create objects.
Edit objects	View and Modify actions enabled for the object's resource and FGAC filter conditions set on the name property in the privilege.	View and edit objects.
Remove objects	View and Delete actions enabled for the object's resource and FGAC filter conditions set on the name property in the privilege.	View and remove objects.
View objects	View action enabled for the object's resource and FGAC filter conditions set on the name property in the privilege.	View lists of objects and object details.

Product, Product Group, Product Configuration and Update Package resources

The following tables provide guidelines for creating user-defined FGAC-enabled privileges with the following resources:

- Product
- Product Group
- Product Configuration
- Update Package

Product and Product Group are FGAC-enabled resources, but Product Configuration and Update Package are not. The latter two resources can be added to privileges in roles that also have privileges based on FGAC-enabled resources.

Product resource

If you want to	You need a privilege with	A user role with the privilege can
Edit products	View and Modify actions enabled for the Product resource and FGAC filter conditions set on the name property in the privilege.	View and edit products. Also applies to using the productList CLI command.
Remove products	View and Delete actions enabled for the Product resource and FGAC filter conditions set on the name property in the privilege.	View and remove products. Also applies to using the productList CLI command.

If you want to	You need a privilege with	A user role with the privilege can
Start products	View and Start actions enabled for the Product resource and FGAC filter conditions set on the name property in the privilege.	View and start products. Also applies to using the productStart and productList CLI commands.
Stop products	View and Stop actions enabled for the Product resource and FGAC filter conditions set on the name property in the privilege.	View and stop products. Also applies to using the productStop and productList CLI commands.
View products	View action enabled for the Product resource and FGAC filter conditions set on the name property in the privilege.	View lists of products and product details. Also applies to using the productList CLI command.

Product Group resource

If you want to	You need a privilege with	A user role with the privilege can
Edit products in a product group	View and Modify actions enabled for the Product Group resource and FGAC filter conditions set on the name property in the privilege.	View and edit products in product groups. Also applies to using the productList CLI command on product group members.
Remove products in a product group	View and Delete actions enabled for the Product Group resource and FGAC filter conditions set on the name property in the privilege.	View and remove products. Also applies to using the productList CLI command on product group members.
Start products in a product group	View and Start actions enabled for the Product Group resource and FGAC filter conditions set on the name property in the privilege.	View and start products in product groups. Also applies to using the productStart and productList CLI commands on product group members. Also can restart products if the role also has stop product group privilege.
Stop products in a product group	View and Stop actions enabled for the Product Group resource and FGAC filter conditions set on the name property in the privilege.	View and stop products in product groups. Also applies to using the productStop and productList CLI commands on product group members.

If you want to	You need a privilege with	A user role with the privilege can
View products in a product group	View action enabled for the Product Group resource and FGAC filter conditions set on the name property in the privilege.	View lists of products in product groups. Also applies to using the productList CLI command on product group members.

Product Configuration resource

If you want to	You need a role with	A user with the role can
Deploy product configurations	Privilege 1: View action enabled for the Product or Product Group resource and FGAC filter conditions set on the name property in the privilege. Privilege 2: View and Deploy actions enabled for the Product Configuration resource.	View and deploy configurations and view deployments on products or product group members. Also applies to using the productList CLI command.
Edit product configurations	Privilege 1: View action enabled for the Product or Product Group resource and FGAC filter conditions set on the name property in the privilege. Privilege 2: View and Modify actions enabled for the Product Configuration resource.	View and edit configurations and view deployments of products or product group members. Also applies to using the productList CLI command.
View product configurations and deployments	Privilege 1: View action enabled for the Product or Product Group resource and FGAC filter conditions set on the name property in the privilege. Privilege 2: View action enabled for the Product Configuration resource.	View configurations and deployments of products or product group members. Also applies to using the productList CLI command.

Update Package resource

If you want to	You need a role with	A user with the role can
Update a product	Privilege 1: View action enabled for the Product or Product Group resource and FGAC filter conditions set on the name property in the privilege. Privilege 2: View and Deploy actions enabled for the Update Package resource. or Privilege 1: View action enabled for the Product resource and FGAC filter conditions set on the name property in the privilege. Privilege 2: Manage Update Package predefined privilege.	Deploy updates on products or product group members. Also applies to using the productList CLI command.

Application and Application Group resources

The following tables provide guidelines for creating user-defined FGAC-enabled privileges with the Application and Application Group resources.

Application resource

For all of the following options, permissions for application groups supersede permissions for applications.

If you want to	You need a privilege with	Comment
Edit applications	View and Modify actions enabled for the Application Group resource and FGAC filter conditions set on the name property in the privilege.	Permissions for application groups supersede permissions for applications.

If you want to	You need a privilege with	Comment
Group applications in application groups	Privilege 1: View action enabled for the Application resource and FGAC filter conditions set on the name property in the privilege. Privilege 2: View , Create and Modify actions enabled for the Application Group resource and FGAC filter conditions set on the name property in the privilege.	Permissions for application groups supersede permissions for applications.
Link products to applications	View action enabled for the Application resource and FGAC filter conditions set on the name property in the privilege.	If the user has rights to view an application, the user can link it to every product even without rights on products.
Remove applications	View and Delete actions enabled for the Application resource and FGAC filter conditions set on the name property in the privilege.	Permissions for application groups supersede permissions for applications.
View applications and their details	View action enabled for the Application resource and FGAC filter conditions set on the name property in the privilege.	Permissions for application groups supersede permissions for applications.

Application Group resource

If you want to	You need a privilege with	A user role with the privilege can
Create application groups	View and Create actions enabled for the Application Group resource and FGAC filter conditions set on the name property in the privilege.	Create application groups that met FGAC conditions.
Create application groups and add applications to the groups	View , Create and Modify actions enabled for the Application Group resource and FGAC filter conditions set on the name property in the privilege.	In the grouping action, for example, add an application group and link an application to the group.

If you want to	You need a privilege with	A user role with the privilege can
Edit application groups	View and Modify actions enabled for the Application Group resource and FGAC filter conditions set on the name property in the privilege.	View application members and their details and edit them.
Remove application groups	View and Delete actions enabled for the Application Group resource and FGAC filter conditions set on the name property in the privilege.	View application members and their details and remove application groups.
View applications as members of an application group and view the application group and its details	View action enabled for the Application Group resource and FGAC filter conditions set on the name property in the privilege.	View applications as members of application groups and application details. If the user has rights on all application groups, the user can view all applications belonging to groups.

Flow resource

The following table provides guidelines for creating user-defined FGAC-enabled privileges with the Flow resource.

If you want to	You need a privilege with	A user with the privilege can
Create flows	Privilege 1: View and Create actions enabled for the Flow resource and FGAC filter conditions set on the name property in the privilege. Privilege 2: View action enabled for the object's resource and FGAC filter conditions set on the name property in the privilege.	Create flows that use objects — applications, application groups, partners, unmanaged products, products — when the user has view rights on the respective objects.
Deploy flows	View and Deploy actions enabled for the Flow resource and FGAC filter conditions set on the name property in the privilege.	View and deploy allowed flows and view flow deployments.

If you want to	You need a privilege with	A user with the privilege can
Edit flows	Privilege 1: View and Modify actions enabled for the Flow resource and FGAC filter conditions set on the name property in the privilege. Privilege 2: View action enabled for the object's resource and FGAC filter conditions set on the name property in the privilege.	For sources and targets, edit flows when the user has view rights on applications, application groups, partners and unmanaged products. For relays, edit flows when the user has view rights on products and unmanaged products. The user can edit flows without rights on objects provided the user does not change the objects in flows. A user without rights on an object cannot edit an object.
Remove flows	View and Delete actions enabled for the Flow resource and FGAC filter conditions set on the name property in the privilege.	Remove the allowed flows.
View flows and their details	View action enabled for the Flow resource and FGAC filter conditions set on the name property in the privilege.	View details of flows and view flow deployments.

HTMLDashboard and HTMLReport resources

The following table provides guidelines for creating user-defined FGAC-enabled privileges with the HTMLDashboard and HTMLReport resources.

If you want to	You need a role with	Comment
View web dashboards	Privilege 1: View action enabled for the HTMLDashboard resource and FGAC filter conditions set on the name property in the privilege. Privilege 2: View action enabled for the HTMLReport resource and FGAC filter conditions set on the name property in the privilege. Privilege 3: Manage Web dashboards, access, reports and database Sentinel predefined privilege.	User needs privileges on all used subcomponents of dashboards (that is, HTMLReport). The Manage Web dashboards, access, reports and database predefined privilege gives permission to access dashboards in the Central Governance user interface. Without the privilege the user can access web dashboards directly with a URL or has dashboard rights.

Design web dashboards

A user with a role with the following Sentinel predefined privileges can design dashboards:

- Manage Web dashboards, access, reports and database
- Manage dashboards in Web Dashboards

The Manage Web dashboards, access, reports and database predefined privilege gives permission to access dashboards in the Central Governance user interface. Without the privilege the user can access web dashboards directly with a URL or has dashboard rights.

Identity stores

8

Central Governance supports internal and external identity stores. There is one internal identity store. You can set up multiple external identity stores on LDAP servers.

All users are associated with an organization, and each organization is associated with an identity store. Multiple organizations can be associated with the same identity store.

Internal and external identity stores

The internal identity store is the default mode for user identification. It uses the Central Governance Access and Security service as the authentication source. To use this mode, select the internal identity store option when adding an organization. Set up and manage users within the Central Governance user interface and associate the users with an organization that uses the internal identity store.

When you use an external identity store for an organization, you must set up and manage users externally and not within the Central Governance user interface. However, you must map roles between Central Governance and the external system.

LDAP identity store

To use LDAP authentication, you must have a running LDAP server and the knowledge to configure the server and create and manage users, passwords, roles and groups.

The Central Governance password policy does not apply to the externally managed users on the LDAP server. Also, you cannot change passwords of external users from within Central Governance.

You can use any LDAP V3 compliant server. Set up users, roles and groups on the LDAP server instance. For example, you could add entities with the following names:

- User: CGuser
- Role: CGrole
- Group: CGgroup

Use Identity Store List page

Use the Identity Store List page to add, view, edit or remove an external identity store. Select **Access > Identity stores** to open the page.

Add identity store

Click **Add identity store** to open the Add Identity Store page. Complete the configuration and click **Save identity store** when done. See *LDAP identity store fields* on page 135 for descriptions of the fields.

View identity store

Click the name of an identity store to open its details page.

Edit identity store

Click the name of an identity store to open its details page, and then click **Edit**. Click **Save changes** after editing fields. See *LDAP identity store fields* on page 135 for descriptions of the fields.

Remove identity store

Do one of the following to remove an identity store:

- Select one or more identity stores on the Identity Store List page and click **Remove**.
- Click the name of an identity store to open its details page, and then click **Remove**.
- Click the name of an identity store to open its details page, and then click **Edit** and **Remove**.

LDAP identity store fields

The following are the fields for configuring an LDAP identity store in Central Governance.

Refer to these fields when managing identity stores. See *Use Identity Store List page* on page 134.

Name

Name of the identity store. This can be any unique name you want.

Description

Optionally, a description of the identity store.

Connection

Server

Host

Fully qualified domain name or IP address of the computer running the LDAP server.

Port

Port the server listens on for connections.

Encryption mode

Security level to use for the connection between Central Governance and the LDAP server. Options are:

None - Clear communication

StartTLS - Transport Layer Security (TLS) secured connection

Certificate

Click **Browse** to select a public-key certificate file in the format DER or PEM or a public certificate chain file in the format P7B (PKCS#7). A certificate is required when StartTLS encryption is selected, representing the certificate authority of the LDAP server. When a certificate is selected, click **Display** to show certificate details.

Authentication

Login

User ID for logging on to the LDAP server to retrieve user roles and user groups. This data enables the administrator to map roles and groups between Central Governance and the LDAP server.

Password

Password for logging on to the LDAP server.

Authentication Mode

Authentication mode for logging on to the LDAP server.

Simple - Use the user's relative distinguished name (RDN) to authenticate

Advanced settings

Connection timeout

Timeout limit in seconds for the LDAP connection.

Number of retries

Number of times Central Governance attempts to re-connect after the connection fails.

Enable connection pooling

Enables connection pooling for user login and filter searches.

Click **Check connection** to verify whether the values are valid for the LDAP server. If the connection fails, Central Governance displays failure reasons returned by the LDAP server.

LDAP tree

Active directory

Indicates whether the LDAP server is a Windows Active Directory implementation. Active Directory enables users to log on with the notation **user@domain**. If this is an Active Directory and the login does not include the @ character, Central Governance adds **@domain** to the login.

If you specify the server is Active Directory, you optionally can provide the value for the domain in the following field.

Active directory domain

For Active Directory LDAP servers, the domain to be added to the user login if the domain is absent.

This field is optional when Active Directory is enabled. If you leave the field blank, nothing is appended to the user name.

Base DN

The base Distinguished Name (DN) to authenticate on the connected LDAP server. The top level of the LDAP directory tree is the base DN.The base DN defines which node of the LDAP tree to use as the root node. Example: `ou=system`

Prefix

Prefix to add to the user login for connection to the LDAP server. Example: `cn=username`.

Suffix

Suffix to add after the user login to the LDAP server. Example: `,ou=users`

Prefix and suffix are optional. If you provide both, Central Governance can use the values to derive a full user DN based on the SubjectDN X500principal:

```
prefix + user login + suffix + baseDN
```

This allows users to enter only their user name at login.

Authorization

The values of the following fields specify the LDAP search queries, telling Central Governance how to retrieve objects from the LDAP structure.

Central Governance uses LDAP queries at run-time to populate fields in the mapping wizard table, and also to evaluate login requests.

To complete these fields it is important to carefully define which LDAP object class controls your access control.

Query syntax must match the target LDAP structure, and use the same object class names as used on the server. Default values that appear in the fields reflect standard naming conventions. If your LDAP server structure includes non-standard naming, you must indicate the customized names in these fields.

Cache timeout

Indicates how long in hours the response to an LDAP query is considered valid.

User DN

Returns the user searched DN from the LDAP server. If this filter is not set, the user searched DN is replaced by the user Full DN. In other filters this will be the userSearchedDN.

Role list

Returns all roles on the LDAP server.

Filtered roles

Returns roles matching the specified filter on the LDAP server.

User roles

Returns all roles of a user on the LDAP server.

Group roles

Returns all roles of a group on the LDAP server.

User groups

Returns all groups of a user on the LDAP server.

Mapping role attribute

Attribute for identifying roles in mapping process.

User mapping

Select a user object class and map values of user attributes available on the LDAP server.

User Filter

Returns all users in a domain on the LDAP server.

First name attribute

Value to filter for a specific user first name.

Last name attribute

Value to filter for a specific user last name.

Email attribute

Value to filter for a specific user email address.

Example LDAP setup for AD

This example provides LDAP identity store default values for any Microsoft Windows Active Directory (AD) having out-of-the-box configuration.

AD is the Microsoft implementation of a directory service. The AD is used to authenticate and authorize all users in a Windows domain-type network.

Connection

Host

```
<ldap.company>.com
```

Port

```
389
```

Login

```
<user email>
```

Password

```
<user password>
```

LDAP tree

Active directory

```
Yes
```

Active directory domain

```
<company>.com
```

Base DN

```
DC=company,DC=com
```

Authorization

Cache timeout

```
12
```

User DN

```
(&(objectClass=organizationalPerson)(sAMAccountName=:userLogin:))
```

Optionally, you can append the string with `(!(userAccountControl=<code>))` to deny access to users who have been disabled. In the following example, 514 is the code for a disabled account:

```
(&(objectClass=organizationalPerson)(sAMAccountName=:userLogin:)(!
(userAccountControl=514)))
```

A list of codes for disabled users is at the following URL:

http://www.netvision.com/ad_useraccountcontrol.php

Role list

```
(objectClass=group)
```

Filtered roles

```
(&(objectClass=group)(cn=:filter:))
```

User roles

```
(&(objectClass=group)(member=:userSearchedDN:))
```

Group roles

```
(&(objectClass=group)(memberOf=:groupFullDN:))
```

User groups

```
(&(objectClass=group)(member=:userFullDN:))
```

Mapping role attribute

```
cn
```

User mapping

User filter

```
(objectClass=organizationalPerson)
```

First name attribute

```
givenName
```

Last name attribute

```
sn
```

Email

```
mail
```

Log on as LDAP user

A user managed on an LDAP server in an external identity store uses this procedure to log on to Central Governance.

Prerequisites

- An external identity store has been added. See *Use Identity Store List page* on page 134.
- The identity store is associated with an organization and external and internal roles are mapped. See *Add organization* on page 120.
- The user knows their LDAP user ID and password and the name of the Central Governance organization associated with the identity store.
- The user knows the URL for connecting to the Central Governance log-on page in a browser.

Steps

1. Open the Central Governance log on page in a browser.
2. On the log-on page, select your organization from the drop-down list and enter your LDAP user ID and password.
3. Click **Sign in** to log on.

Once logged on, the actions the user can perform depends on the mapping of internal to external roles within the user's organization.

Product registration 9

The following topics are about registering products for Central Governance to manage.

Central Governance must be installed and running before products can register and be managed.

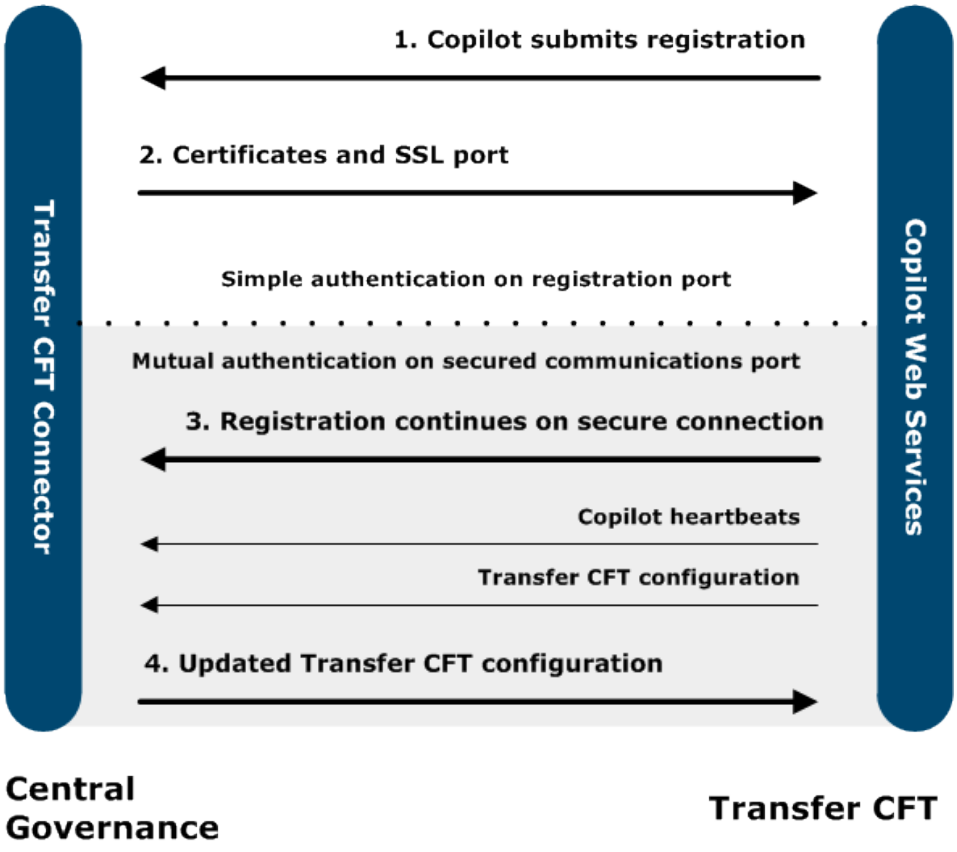
This version of Central Governance is compatible with Axway Transfer CFT 3.1.2 and 3.1.3 and Axway SecureTransport 5.3.1 and later.

Transfer CFT registration

The following topics describe the tasks to register Transfer CFT in Central Governance.

The parameters for connecting Transfer CFT and Central Governance are specified during installation of Transfer CFT. After installation, the Transfer CFT Copilot is started, and Copilot connects to Central Governance to begin the registration process. Only Copilot, and not Transfer CFT server, must be started to begin the registration process.

The following graphic illustrates the Transfer CFT registration process with Central Governance.



Use the numbers in the graphic to correlate with the text describing each phase.

As noted in *4. Transfer CFT configuration updated* on page 144, Central Governance pushes configurations to Transfer CFTs upon registering. By default Central Governance sets Transfer CFTs to use the Visibility service within Central Governance. If you want Transfer CFTs to use Axway Sentinel instead, see *Use local settings for Sentinel* on page 146 for details.

1. Registration request

Copilot must be started to begin the registration process. Transfer CFT also can be started, but this is optional for registration. Copilot makes a simple-authentication connection to Central Governance.

A registration request is sent that contains:

- Information about Transfer CFT, including its instance name, host, port, operating system and version.
- The Central Governance shared secret. The shared secret, obtained from the Central Governance administrator, is set in Transfer CFT when Transfer CFT is installed.
- Certificate signing requests (CSRs) for Central Governance to process, and the names of the certificate authority (CA) services that process the CSRs.

Products must use the Central Governance shared secret to register successfully. The shared secret was set during initial configuration of Central Governance, immediately after it was installed but before the server was started the first time.

2. Certificates for Transfer CFT

The following describes the certificates Central Governance sends to Transfer CFT, depending on your choice of certificate authorities.

Default CAs

When the default CAs are used, the Central Governance Access and Security service creates a security entity for Transfer CFT. It uses as the entity name the concatenation of the Transfer CFT InstanceId and InstanceGroup, which both were set during installation of Transfer CFT.

Central Governance then generates a signed business certificate to be used for securing file transfers between the registering Transfer CFT and all other Transfer CFTs. The Transfer CFT name is used to identify the business certificate. The business certificate authority is the internal CA that generates and signs the certificate, which is known as the PassPort CA in the Access and Security service.

Along with the business certificate, Central Governance generates and sends a governance certificate to Copilot. The governance certificate is used to secure communications between Central Governance and Transfer CFT and its Copilot. The governance certificate authority is the CA that generates this certificate, which is known as PassPort Product CA in the Access and Security service.

Custom CAs

You can customize the business and governance CAs on the Central Governance configuration page. See *Access and security* on page 52 for details.

Before Transfer CFT is registered in Central Governance, the custom governance CA must be known and trusted by the registering Transfer CFT.

Customizing the business CA affects the secured transfers between the registering Transfer CFT and other Transfer CFTs. All other Transfer CFTs must know and trust the new business CA.

3. Mutually authenticated connection

Once Copilot has the certificates, it connects to Central Governance on the secured communications port using the Central Governance certificate.

All heartbeat connections between Copilot and Central Governance are mutually authenticated on the same port. Only Copilot sends heartbeats during registration.

Every time Copilot starts it checks the validity of the certificates. A renewal request is scheduled when a certificate has expired or is about to. The Transfer CFT parameter `cg.renewal_period`, with a default value of 60 days, is used to identify certificates about to expire.

For certificates about to expire, renewals are through the mutually authenticated connection. However, if the certificate used for mutually authenticated connections has expired, the simple authentication connection is used. Before Transfer CFT receives a new certificate, the product status is Unreachable in the Central Governance user interface.

4. Transfer CFT configuration updated

Completing the registration process, Central Governance gets the current configuration of Transfer CFT and changes it.

- Transfer CFT is configured to use the Central Governance Access and Security service for access management.
- Transfer CFT is configured to use the Central Governance Visibility service for transfer monitoring.

These changes create two security profiles (CFTSSL) on the Transfer CFT. The profiles are named SSL_DEFAULT; one profile is of type client and one is of type server. Their SSL version is TLSV1COMP. The configured cipher list is CIPHLIST= ('47', '53') for Transfer CFT 3.1.2 and 3.1.3. The values represent the following cipher suites:

- 47: TLS_RSA_WITH_AES_128_CBC_SHA
- 53: TLS_RSA_WITH_AES_256_CBC_SHA

The client and server security profiles must be mutually authenticated. However, by default, a registered Transfer CFT does not have a protocol with a security profile. You must edit the Transfer CFT configuration to support protocols with mutual authentication.

Central Governance sends the updated configuration to Transfer CFT. The following are the Transfer CFT parameters updated in this process.

Parameter	Value
am.passport.cg.organization	Org
am.passport.domain	CG
am.passport.hostname	<Central Governance host name >
am.passport.instance_id	$(cft.instance_ group).$(cft.instance_id)
am.passport.port	6666
am.passport.use_ssl	Yes
am.passport.userctrl.check_permissions_on_transfer_ execution	No
am.type	passport
cft.purge.rt	10D
cft.purge.rx	10D
cft.purge.st	10D
cft.purge.sx	10D
cft.server.bandwitdth.cos	4
cft.server.bandwitdth.cos.0.max_rate_in	-1
cft.server.bandwitdth.cos.0.max_rate_out	-1
cft.server.bandwitdth.cos.1.weight_in	80
cft.server.bandwitdth.cos.1.weight_out	80
cft.server.bandwitdth.cos.2.weight_in	15
cft.server.bandwitdth.cos.2.weight_out	15
cft.server.bandwitdth.cos.3.weight_in	5
cft.server.bandwitdth.cos.3.weight_out	5

Parameter	Value
cft.server.bandwitdth.enable	No
cg.mutual_auth_port	<secured communications port>
copilot.misc.createprocessasuser	No
pki.type	cft
sentinel.trkipaddr	<Central Governance host name >
sentinel.trkipport	1305
sentinel.xfb.enable	Yes

In addition to the changes in the preceding table, Transfer CFT becomes linked to the policy configured in the Transfer CFT `cg.configuration_policy` parameter. If the policy does not exist in Central Governance at registration, no operation is performed even if the policy is created after registration.

The policy configuration might overwrite the default parameters Central Governance updates. For instance, if the policy contains access management set to **None**, Transfer CFT is configured to use None rather than the Central Governance Access and Security service.

See *Manage policies* on page 210 for information about adding policies for Transfer CFTs in Central Governance.

Use local settings for Sentinel

When Transfer CFTs register, the default is for Central Governance to configure Transfer CFTs to use the Visibility service within Central Governance. Outside of Central Governance, the Visibility service is Axway Sentinel. There might be occasions when Transfer CFTs are using Sentinel and must continue using it after registering with Central Governance. In these cases you can disable Central Governance from overwriting existing configurations in Transfer CFTs for using Sentinel.

Set the following Central Governance property to **false** before registering Transfer CFTs:

```
setSentinelAtRegistration=true
```

The property is in the `com.axway.cmp.cgcft-default.cfg` file at `<install directory\runtime\com.axway.nodes.ume_<UUID>`. Restart Central Governance for the change to become effective.

The following are the Sentinel properties in Transfer CFTs whose values are untouched when the registration property in Central Governance is disabled.

- sentinel.trkipport
- sentinel.trkipaddr
- sentinel.xfb.enable

Registration results

Registration is successful when one of the following statuses is reported on the Central Governance Product List page:

- Started. Registration succeeded, and Transfer CFT is running.
- Stopped. Registration succeeded, and Transfer CFT is not running.

If a problem occurs during registration, Transfer CFT is registered with an error. See *Transfer CFT registration troubleshooting* on page 148 for information on registration errors.

Change CAs after Transfer CFT registration

After registering Transfer CFTs in Central Governance, changing any of the Central Governance certificate authorities requires resubmitting certificate registration:

1. Transfer CFT Copilot requests a new SSL certificate signed by the new CA.
2. Central Governance sends the requested certificate to Copilot.

Also see *CA services* on page 100 for more information.

Governance CA

Changing the governance CA affects registered Transfer CFTs. You must import the new CA in Transfer CFT and schedule the certificate registration.

Transfer CFT 3.1.2

For Transfer CFT 3.1.2, stop Copilot and Transfer CFT and do the following:

Replace the PassPort CA by running the following Transfer CFT command:

```
PKIUTIL PKICER ID = 'PassPortCA', ROOTCID = 'PassPortCA', ITYPE =
'ROOT', INAME = '<GovernanceCACertificateFile>', IFORM = 'DER', MODE
= 'REPLACE'
```

Then trigger the certificate registration by resetting the `cg.registration_id` to **-1** with the following command:

```
CFTUTIL UCONFSET ID=cg.registration_id, VALUE=-1
```

Restart Copilot.

Transfer CFT 3.1.3

For Transfer CFT 3.1.3, import the new CA by doing one of the following:

- Configure the CA by setting the CA Certificate by using the installer for Transfer CFT in configure mode. You must stop Copilot and Transfer CFT before starting the installer in configure mode. You can run the configure command in the Transfer CFT installation directory to start the installer in configure mode.

  or

- If you do not want to stop Transfer CFT, use the following commands:
  - `PKIUTIL PKICER ID = '<CG CA new alias>', ROOTCID = '<CG CA new alias>', ITYPE = 'ROOT', INAME = '<GovernanceCACertificateFile>', IFORM = 'DER', MODE = 'CREATE'`
  - `CFTUTIL UCONFSET ID=cg.ca_cert_id, VALUE='<CG CA alias>,<CG CA new alias>'`

Then set the parameter `cg.certificate.governance.renewal_datetime` (format: YYYYMMDDHHMMSS + GMT) to schedule the request at first heartbeat after the specified date and time.

The heartbeat interval is specified in seconds in the `cg.periodicity` parameter (default value 600). For example, schedule the certificate request to start December 23, 2014, at 14:30:00 + GMT by running the following command:

```
CFTUTIL UCONFSET ID=cg.certificate.governance.renewal_datetime,
VALUE=20141223143000
```

Transfer CFT becomes unreachable until the new certificate is received.

Business CA

If the business CA is changed, a new business SSL certificate can be requested. The new certificate is signed by the new CA and used in secured flows.

Schedule a new certificate request starting with the time specified in the Transfer CFT parameter `cg.certificate.business.renewal_datetime` (format: YYYYMMDDHHMMSS + GMT).

Make sure the new business CA is known by all Transfer CFT flow partners before the certificate is renewed.

Transfer CFT registration troubleshooting

The following are guidelines for troubleshooting Transfer CFT registration.

Transfer CFT does not display in UI

Transfer CFT has registered and is running, but is not listed in the Central Governance user interface. To ensure that Transfer CFT has registered properly:

- Verify Central Governance is fully started. See *Status of Central Governance and services* on page 42.
- Verify the Central Governance IP address or FQDN you entered in Transfer CFT.
- On the computer running Transfer CFT, verify Central Governance is reachable at the IP address or FQDN specified in the Transfer CFT configuration.
- Check the Central Governance logs. If the Transfer CFT does not appear, typically the Transfer CFT was unable to contact Central Governance.

Transfer CFT registered in error following a start

Transfer CFT is started and has a "registered in error" status:

- Verify Central Governance is fully started. See *Status of Central Governance and services* on page 42.
- Check the Central Governance logs for additional information.
- Check whether the shared secret provided by the Central Governance administrator matches the shared secret Transfer CFT submitted when registering.
- Check whether the governance certificate authority exists in the Transfer CFT PKI. See *CA services* on page 100 for more information.
- Check whether Transfer CFT has the maximum allowed number of CRONTABs within an active configuration (CFTPARM). Central Governance creates and deploys a dedicated CRONTAB named CGCRONTAB during registration and attaches it to the active configuration for use in CRONJOBs. Reduce CRONTABs to less than 32 in the Transfer CFT configuration to allow the CGCRONTAB creation.
- If the Transfer CFT parameter `cg.configuration_policy` is set and the policy exists in Central Governance, check whether the policy deployment status is in error.

Next steps

If registration fails:

1. Stop Transfer CFT and its Copilot.
2. Remove Transfer CFT from Central Governance. See *Remove products* on page 166.
3. Check the registration settings in Transfer CFT for accuracy. Check that:

   ```
   cg.registration_id = -1
   ```

   ```
   cg.port = <Central Governance registration port>
   ```

   See *Check registration settings* on page 149 for details.
4. Restart Copilot to start the registration process.

Check registration settings

Use one of the following methods to check the registration settings in Transfer CFT.

Method 1

Run the following command to display a list of properties and values for Central Governance:

```
CFTUTIL LISTUCONF ID=cg*
```

Check the values and change if needed. For example, if the registration ID is not `-1`, change the value with the command:

```
CFTUTIL UCONFSET ID=cg.registration_id, VALUE=-1
```

Check the Transfer CFT PKI and add the custom governance CA if missing:

```
PKIUTIL PKICER ID = '<CG CA alias>', ROOTCID = '<CG CA alias>',
ITYPE = 'ROOT', INAME = '<GovernanceCACertificateFile>', IFORM =
'DER', MODE = 'REPLACE'
```

Method 2

In the Copilot user interface, go the UCONF table, search for the Central Governance settings, and check the registration ID and port. Change the values if needed.

If the governance CA was changed in Central Governance, go to the Public Key Infrastructure table and check whether the new CA exists. Import it if needed.

SecureTransport registration

The following topics describe what happens when SecureTransport registers in Central Governance and the prerequisites and process for registering.

Unique and duplicate server communication profiles

Before registering you should understand how unique and duplicate server communication profiles are determined when SecureTransport has SecureTransport Edge servers deployed within its infrastructure.

One or more SecureTransport Edge servers can be grouped in a network zone in SecureTransport. An edge can belong to one or more network zones. During Central Governance registration, for each network zone, protocol servers activated on each edge are imported as server communication profiles. If multiple edges in the same network zone have the same protocol server activated with the same configuration, a single server communication profile is declared for all edges. In this case the Edges field of the server communication profile lists the edge server titles as declared in the SecureTransport server.

A server communication profile is considered a duplicate when the following configuration is identical:

- The exchange protocol
- The exchange port
- Whether SSL/TLS is enabled, and if enabled:
    - The client authentication mode
    - The certificate used to secure the connection is the same in terms of content
- FIPS transfer mode

If any of these settings differs, separate server communication profiles are imported in the static configuration of SecureTransport. However, server communication profiles can be identical across network zones, in which case they are imported separately in each network zone.

See *SecureTransport network zones* on page 200 for more information.

PeSIT services

If PeSIT services are activated on SecureTransport, client communication profiles — corresponding to each SecureTransport server communication profile — are created in Central Governance during registration. The following information is retrieved from the server communication profiles:

- The network protocol.
- The PeSIT login which, as for the server communication profile, represents the SecureTransport product name.
- If SSL is activated, the SSL certificate to be used when SecureTransport is client. The certificate must have both server and client authentication key usage extensions to be usable as both the client and server certificate.
- FIPS mode.

If there is no server communication profile created in the private network zone during registration, no client communication profile is created.

The generated client communication profiles can be used in flows. However, you can create other client communication profiles. See *PeSIT client communication profile* on page 285.

Prerequisites

Before registering SecureTransport, make sure the configuration is correct for transferring files via preferred protocols. This includes services, ports, SSL certificates for services and protocol-specific configuration.

After registering, if you need to change the SecureTransport configuration, change it first on SecureTransport and then make parallel changes for the SecureTransport in the Central Governance user interface.

Central Governance and SecureTransport must be configured and running for SecureTransport to register successfully. The SecureTransport Transaction Manager must be running; SecureTransport Edge, if installed, must be configured correctly.

See the SecureTransport user documentation for more information about configuring SecureTransport properly for file transfers.

The following steps must be completed in the SecureTransport administration user interface.

Network zones

Network zones and streaming configuration between the SecureTransport server and edges must be defined in SecureTransport Administration before starting the registration process. In Central Governance the list of edge node addresses is set in the Hosts field of the network zone. The FQDN is set by default to the first edge node address. Registration fails when the definition of a network zone is invalid on SecureTransport.

Administrator account

1. Make sure the SecureTransport administrator can log on using client certificates.
    a. In Setup > Admin Settings, set client certificates to **optional**.
    b. Accept certificates issued by **any trusted issuer**.
2. Make sure you have an administrator user with a Master Administrator role in Accounts > Administrators.

See Registering SecureTransport in Central Governance in the SecureTransport Administrator Guide.

Static configuration

Static configuration must be setup and started on SecureTransport. Central Governance retrieves it in the registration process.

1. Configure, enable and start the server protocols to be exposed in Central Governance at Operations > Server Control.
2. Specify the Client Certificate Authentication and SSL Encryption at Access > Secure Socket Layer. Determine whether client certificate authentication is optional or mandatory depending on your security policy.
3. If applicable, configure the SSL certificates for use by server protocols supporting a secure connections. Generate or import certificates referenced in the SSL Key Alias fields.
4. Central Governance retrieves the static configuration for FTPS during the registration process. Set the Base Port, Number of Ports and Port End under FTP Passive Mode at Setup > FTP Settings in SecureTransport.

Central Governance configuration in SecureTransport

The parameters for connecting SecureTransport and Central Governance are specified on the Central Governance page at Setup > Central Governance in the SecureTransport administration user interface. The Central Governance administrator must provide many of the field values to the

SecureTransport administrator. Many of the values are set on the configuration page in Central Governance. The SecureTransport fields are:

Host

Must contain the same IP address or FQDN used on the configuration page in Central Governance. Registration fails unless the same value is used on both sides.

Port

Port of the Central Governance agent as entered in the Agent section on the configuration page in Central Governance. The default port is 5701.

Name

Name of the Central Governance agent as entered in the Agent section on the configuration page in Central Governance.

Shared Secret

The shared secret set up on the configuration page in Central Governance for products to use during registration.

Product Identifier

Central Governance uses this value to uniquely identify the instance or cluster of SecureTransport being registered.

Administrator Name

The name of the administrator user account for Central Governance to use for logging on to the SecureTransport Administration Tool.

Local Bind Address

The IP address or FQDN of the computer running SecureTransport.

Local Bind Port

The port used by the Central Governance agent to listen for connections.

Account Home Folder Prefix

Base home folder for accounts created by Central Governance.

Real User (Windows only)

If the the account home folder prefix is on a shared network; specify a real user who has access to it. You must specify the real user in a password vault file.

For more information, see the "Add a user to a password vault" topic in the SecureTransport Administrator Guide.

UID (Unix and Linux only)

UID for accounts created by Central Governance.

GID

GID for accounts created by Central Governance.

Failed login attempts before account is locked

The number of times a user can attempt to log on with invalid credentials before being locked out. This field is optional.

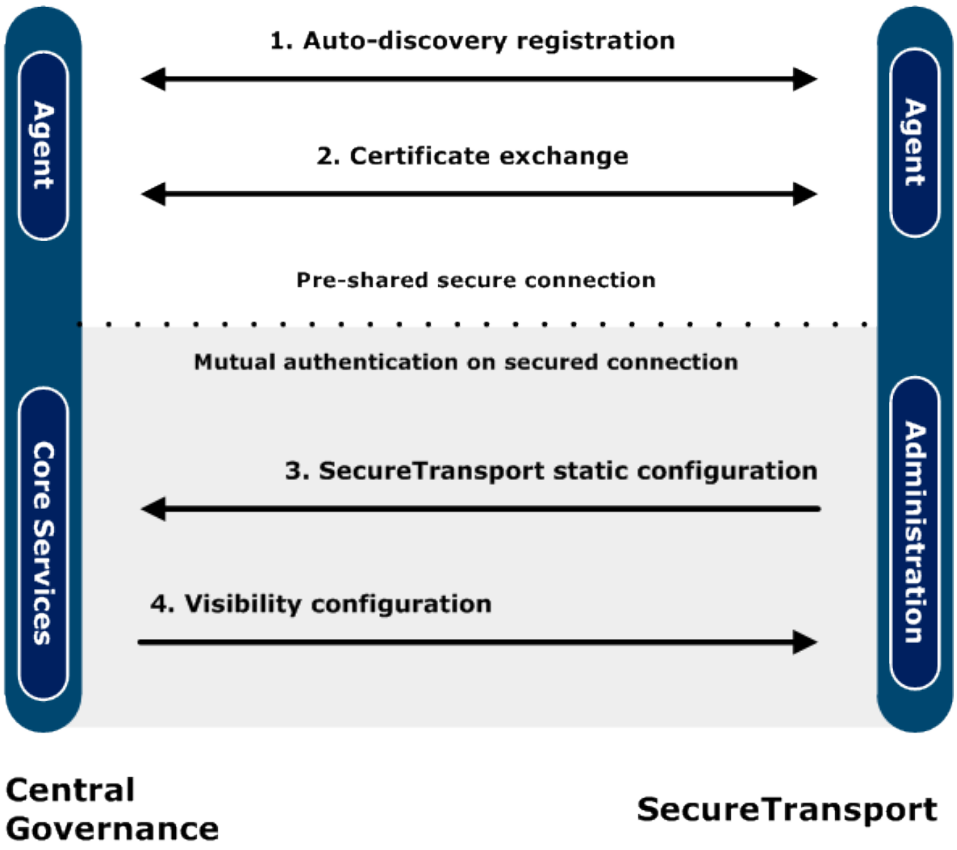
Expire password on account creation

When enabled users must change their passwords after logging on the first time.

For more information see the Central Governance configuration topic in the SecureTransport Administrator Guide for the configuration to complete in SecureTransport.

Registration process

The following graphic illustrates the SecureTransport registration process with Central Governance. The process begins when the Central Governance configuration is saved in SecureTransport. The graphic shows what happens next.



Use the numbers in the graphic to correlate with the text describing each phase.

1. Auto-discovery registration

The SecureTransport agent connects over a pre-shared channel to the agent in Central Governance. The registration request includes the name, host name, operating system and version of the SecureTransport.

Products must use the Central Governance shared secret to register successfully. The shared secret was set during initial configuration of Central Governance, immediately after it was installed but before the server was started the first time.

2. Certificate exchange

The Central Governance governance certificate authority generates and sends to SecureTransport an SSL certificate for securing connections between the two products. To establish mutual authentication, Central Governance also sends the governance certificate authority as a trusted authority. These enables the SecureTransport administrator user to have a secure connection.

Once SecureTransport has the certificates, it connects to Central Governance on the secured communications port using the Central Governance certificate.

All data connections between SecureTransport and Central Governance are mutually authenticated on the same port, via SecureTransport RESTful APIs. For instance, Central Governance pushes flow configurations over the mutually authenticated channel using the HTTP SET method of the RESTful APIs. Deployment of flows by Central Governance to SecureTransport succeeds even if the SecureTransport Agent is unavailable.

When this certificate expires, Central Governance renews it and sends it to SecureTransport.

3. SecureTransport static configuration

Central Governance receives the current configuration of SecureTransport. This includes:

- Server communication profiles corresponding to the activated and running protocol servers on SecureTransport.
- Network zone definition and server communication profiles corresponding to the activated and running protocol servers on SecureTransport Edge servers, if any.
- Certificates for the SSL certificates used by the servers supporting secured connections. Only the public key of the SSL certificate is retrieved along with the chain of certificate authorities.

4. Visibility configuration

Central Governance configures SecureTransport to use its Visibility service, which is an embedded instance of Axway Sentinel. The following SecureTransport properties are changed during registration.

Sentinel

Property	Value
Host	<Central Governance host name>
Port	Visibility port (default 1305)

Overflow file

Property	Value
Name	sentinel-overflow.db
Path	var/run/
Size	5 MB
Warning threshold	90

The following properties are not changed:

- Send Heartbeat to Axway Sentinel Every
- Events (Available Event States, Event States to Send)
- When Overflow File Exceeds Maximum Size.

Registration results

Registration is successful when the SecureTransport status is Started on the Central Governance Product List page.

SecureTransport creates the following configurations for use in Central Governance:

- CentralGovernanceApplication - An application of type AdvancedRouting to be used in all subscriptions.
- CentralGovernanceRouteTemplate – A route package template to be used for all routes of AdvanceRouting type.

See the advanced routing topic in the SecureTransport Administrator Guide for more information.

If after registering the product does not appear on the Product List page or has a registered in error status, see *SecureTransport registration troubleshooting* on page 157.

Status monitoring

Central Governance monitors the status of registered SecureTransports through their agents. SecureTransport has a status of started in Central Governance when the SecureTransport agent communicates successfully with the Central Governance agent. Even if SecureTransport is not running, the status is started in Central Governance as long the agents on both sides can communicate. If communication between agents breaks, the status of SecureTransport is unreachable on the Product List page in Central Governance.

Remove and re-register

The best practice for removing a registered SecureTransport is to first make sure the SecureTransport status is not in unreachable status in Central Governance. Once you remove the product, the Central Governance configuration in SecureTransport becomes disabled. You then can enable the configuration in SecureTransport to re-register the product.

See *Remove products* on page 166 for information about removing registered products from Central Governance.

SecureTransport registration troubleshooting

The following are guidelines for troubleshooting SecureTransport registration.

SecureTransport does not display in UI

SecureTransport has registered and is running, but is not listed in the Central Governance user interface. To ensure the product has registered properly:

- Verify Central Governance is fully started. See *Status of Central Governance and services* on page 42.
- Verify the Central Governance host matches in SecureTransport. If Central Governance uses an IP address as its host value, use the IP address in SecureTransport. If Central Governance uses an FQDN as its host value, use the FQDN in SecureTransport.
- On the computer running SecureTransport, verify Central Governance is reachable at the IP address or FQDN specified in SecureTransport.
- Check the Central Governance logs. If the SecureTransport does not appear, typically it was unable to contact Central Governance. This can occur when:
    - The Central Governance shared secret is incorrect in SecureTransport. Check the SecureTransport logs for a shared secret error.
    - The administrator account does not exist in SecureTransport or has an incorrect level of access.
    - The administrator account cannot connect using the certificate. The following option must be enabled in SecureTransport: Enable administrator login using client certificate.

SecureTransport registered in error

SecureTransport is started and has a "registered in error" status:

- Verify Central Governance is fully started. See *Status of Central Governance and services* on page 42.
- Check the Central Governance logs for additional information.
- Review the error message, which displays as a tooltip in Central Governance, for the cause. Often it is an issue in retrieving configuration. For example, if the SecureTransport Transaction Manager is stopped, Central Governance cannot retrieve configuration.

Next steps

If registration fails:

1. Remove SecureTransport from Central Governance by going to the Product List page and removing it.
2. Fix the cause of the failed registration.
3. Enable Central Governance configuration in SecureTransport to start the registration process again.

Product updates 10

Central Governance can apply updates remotely to products that have been registered successfully. Updates are files containing service packs, patches and version upgrades for products.

In this version of Central Governance, single instances of Transfer CFTs 3.1.3 and later are supported for updating remotely. The Transfer CFTs can be running on supported operating systems except z/OS and IBM i.

You cannot use the product update user interface to remotely update Transfer CFTs running in clusters of multiple nodes.

Update summary and workflow

The following summarizes the product update process and provides the workflow for updating products. A system engineer or technician responsible for installing, configuring and maintaining Central Governance can update products.

Summary

Central Governance maintains a repository to store uploaded update files at:

```
[install directory]\runtime\com.axway.nodes.ume_
[variable]\data\updateRepository
```

Access the repository only through the user interface. Do not access the directory.

Central Governance manages the updating process from start to finish. Once you select an update and the products to apply it, Central Governance sends the update to the products, which shut down, start their local installers in update mode, apply the update and restart.

The Product List page lists all products registered in Central Governance. The Version column reports the versions of the products. After applying updates, values in the column change to reflect that service packs, patches or version upgrades have been installed.

You cannot uninstall updates remotely; you must uninstall at the product level. You also cannot remove update files from the Central Governance update repository.

Workflow

1. Download an update file from the Axway Sphere support website at https://support.axway.com. Save the file in a file system Central Governance can access.

2. Select **Products > Updates** in the Central Governance user interface. Upload the file to the update repository. See *View, upload updates* on page 160 for details.
3. On the Product List page, click an update to apply it to one or more products. See *Apply update to product* on page 160 for details.

Manage product updates

Use the following procedures to manage product updates on the Update List page. You can:

- View a list of uploaded update files
- Upload update files downloaded from Axway Sphere
- Apply updates to products

Prerequisites

You must download service packs, patches or version upgrade files for products from the Axway Sphere support website before you can upload files to the Central Governance update repository. Files you want to add to the repository must be on a file system accessible to Central Governance.

View, upload updates

1. Select **Products > Updates** to open the Update List page. The page lists all updates in the Central Governance repository.
2. Click **Upload file** and select the file you want to upload to the repository. Central Governance displays a message to confirm a successful upload.

Repeat the steps to upload more updates.

Apply update to product

1. Select **Products > Updates** to open the Update List page. The page lists all updates in the Central Governance update repository.
2. Click the name of an update to open a details page. Review the information to confirm you want to apply the update to products.
3. Click **Update products** to open the Update Products dialog. The dialog lists eligible products to apply the update.
4. Select one or more products and click **Update products**. Central Governance displays a success or failure status.

5. Once you have applied updates to products, you can do the following to monitor progress or confirm changes:
    - Select **Administration > Deployments > Updates** and review the Status, Target and Item columns. Place the cursor over the status message to display the sequence of the update process.
    - Select **Products** on the toolbar and review the Version column for the products.

A status of **Updated** on the Deployment List page means the update succeeded.

If an update failed, see the next topic for guidelines to resolve.

Troubleshoot product updates

Once you launch an update, Central Governance tries to apply it to the selected products. Go to **Administration > Deployments > Updates** to view update statuses. **Updated** indicates an update was applied successfully. **Updated in error** and **Uploaded in error** indicate updates failed to install.

An update failure can be due to any number of conditions on the product server. Select the failed update on the Updates tab of the Deployment List page and click **Retry**. If the update fails again, troubleshoot the issue on the computer where the product is installed. Check the local installer and product logs for failure clues. If necessary, apply the update to the product locally.

Once the product has been updated successfully by retrying in Central Governance or resolving locally, check the Product List page to confirm the correct version is installed.

Product operations 11

The Product List page enables you to view and perform actions on registered products. The page shows all products or only those meeting your filter criteria. You can click the name of a product to display its details page and to view or change the configuration of a product.

Product statuses and operations

You can view the status of registered products and perform operations on them on the Product List page.

Statuses

The following statuses are displayed. Some special statuses apply only to products running in a cluster of multiple nodes.

Status	Description
Stopped in error	Displayed for a system in an abnormal state when it failed to stop or crashed. This status is always associated with an error message. If a product cluster: At least one node is stopped in error.
Registering	Displayed when the system is registering with Central Governance.
Registered in error	Displayed when the system failed to register with Central Governance.
In progress	Product cluster only: At least one node is starting and one node is stopping.
Started	Displayed when the system has started successfully. If a product cluster: All nodes are started.
Starting	Displayed from the time a start command occurs to the time the system returns a status, or until a timeout. If a product cluster: At least one node is starting.
Partially started	Product cluster only: At least one node is stopped and at least one node is started.

Status	Description
Stopped	Displayed when the system has stopped successfully. If a product cluster: Indicates all nodes are stopped.
Stopping	Displayed from the time a stop command occurs to the time the system returns a status, or until a timeout. If a product cluster: At least one node is stopping.
Unreachable	Displayed when Central Governance cannot get the status of the system. This can occur if network issues prevent communication. This status is always associated with an error message.

Operations

You can perform the following operations on products The user interface only allows you to perform operations compatible with the product status.

Note Central Governance cannot start or stop registered SecureTransports.

Operation	Description
Start	Starts the product.
Stop	Stops the product normally. A normal stop waits for transfers to complete and then stops the product.
Quick stop	A quick stop places all transfers in progress in ready status and then stops the product. Transfers resume when the product is restarted.
Force stop	A force stop aborts the transfers in progress and then stops the product. When the system is restarted, pending transfers are resumed, but aborted transfers remain aborted.
Restart	Restarts a product that is started.
Remove	Removes a product from Central Governance. This can be performed on a system that is in any state, except registering. It removes all information related to the system from Central Governance, but has no effect on the system's status.

Transfer CFT status monitoring

Central Governance monitors the status of registered Transfer CFTs through heartbeats.

Transfer CFT and its Copilot send heartbeats via the persistent mutually authenticated connection with Central Governance. The default interval for both to send heartbeats is every 10 minutes.

Transfer CFT has a status of unreachable in the Central Governance user interface when:

- Neither Transfer CFT or its Copilot have sent heartbeats for longer than the specified interval.

  or

- Transfer CFT sends a heartbeat that reports a stopped status for its Copilot.

SecureTransport status monitoring

Central Governance monitors the status of registered SecureTransports by monitoring the agents of the products.

Start and stop products

Use the following procedures to start and stop products on the Product List page. Click **Products** on the top toolbar to open the page.

Start products

1. Select the products to start.
2. Click **Start**. When started successfully, the Status column displays **Started**.

Stop products

1. Select the products to stop.
2. Click **Stop**. When stopped, the Status column displays **Stopped**.

View or edit product details

The product details page has more information for the product than the Product List page. If the details page has a Nodes section, the product is running in a cluster of multiple nodes.

On the details page you can perform start and stop operations on the product or click **Edit** to change details. For example, you can edit:

- The groups associated with the product. You can add multiple products to the same group for mass operations or for fine-grained access control. See *Groups* on page 218 for more information.

- Contact information.
- Other details, such as adding tags, which are useful when performing a search.

The details page has links to:

- Edit the product configuration.
- Remove the product from Central Governance. See *Remove products* on page 166.

For Transfer CFT on the details page you can also:

- Review the product log. See *Product logs* on page 165.
- Manage Transfer CFT legacy flows. See *Transfer CFT legacy flows* on page 358.

Product logs

Use logs to monitor usage or diagnose problems for registered products. You can view the log for only one product at a time.

See Transfer CFT error messages documentation for the messages Transfer CFT generates and corrective actions when applicable.

View log

1. Click the name of a product on the Product List page to open its details page.
2. Click **Logs** on the right side of the page.

   The log page is displayed where you can:

   - Click **Refresh** anytime to update the log entries.
   - Sort the entries by newest or oldest.
   - Filter the entries, saving filters for future use.

The following describes the log table columns.

Date/time

The server date and time of the log entry. Format: YYYY-MM-DD hh:mm:ss.

Level

Level of the log entry. Levels, from highest to lowest verbosity, are INFO, WARNING, ERROR, FATAL.

Code

Code associated with the log message.

Message

Actual log message.

Filter log

You can filter a log by one or multiple conditions. The filters you add are saved until deleted or the browser cache is cleared. You can, for example, filter by level, leave the page and return, and the displayed log entries are filtered by level.

Click **Filter** and select a filter type to add. You can add multiple filters.

Date/time

You can filter log entries by age in hours or generated within a date range.

Level

You can filter log entries by severity. This filtering provides cumulative verbosity. If you filter by Info level, Info messages and all message levels above the Info level are displayed. If you filter by Fatal level, only fatal log entries are displayed.

Pattern

You can filter log entries by full or partial codes or messages or both. This filtering is case sensitive.

Added filters are displayed at the top of the page. Click a filter to change it. Click the appropriate X icon to delete a single filter or to clear all filters.

Remove products

Removing products means the systems no longer are registered in Central Governance and managed by it.

Guidelines

Removing a product can affect flows. Before removing, check for defined flows that use the product. See *General concepts about flows* on page 230.

Transfer CFT

Confirm that Transfer CFT and its Copilot are stopped. Both must be stopped before removing Transfer CFT from Central Governance. You can stop Transfer CFT from Central Governance, but you must stop Copilot separately. See *Start and stop products* on page 164.

SecureTransport

The best practice for removing a registered SecureTransport is to first make sure the SecureTransport status is not in unreachable status in Central Governance. Once you remove the product, the Central Governance configuration in SecureTransport becomes disabled. You then can enable the configuration in SecureTransport to re-register the product.

Steps

1. Select the **Products** tab.
2. Select the product to remove.
3. Click **Remove**.

If the removed product is used in flows, a confirmation message is displayed. If you confirm the removal, the product is removed from the flows in Central Governance, and the status of the flows is recalculated. You must redeploy the flows manually.

Transfer CFT configuration 12

The following topics describe configuring Transfer CFTs managed by Central Governance.

When Transfer CFT registers with Central Governance, it has a default configuration that enables it to operate without changes. However, your needs might require changing the default settings. Central Governance uses the Copilot WebService to deploy configuration changes to Transfer CFT.

Change configuration

Use this procedure to change, save and deploy a configuration for Transfer CFT.

1. Select the **Products** tab to view available products.
2. Click the product name to open its details page. On this page you can click **Edit** and change details about the product, including groups tags, description and contact information. However, you must go to the next step to change the product configuration.
3. Click **Configuration** to open the Configuration page and click **Edit** to open edit mode.
4. In the sub-menu panel on the left, select the configuration section to edit.
5. Change the configuration as needed.
6. Click **Save**.
7. Click **Deploy** when you are ready to push the configuration change to Transfer CFT. Then click a deployment option. You can:
    - **Deploy configuration** to deploy the configuration and restart the Transfer CFT for the changes to become effective on it.
    - **Deploy, no restart** to deploy the configuration, but not restart Transfer CFT. You must restart the product later for the changes to become effective. After the configuration is deployed, the status message at the top of the Configuration page reminds that restarting is required. A restart reminder also is displayed for the product in the Configurations list on the Deployment List page at **Administration > Deployments**.

      You can use this option on Transfer CFTs running on z/OS and IBM i computers when `copilot.misc.cftstart.enable` is set to **false** to disable Copilot operations such as start and stop.
8. If you selected the **Deploy, no restart** option, go to the Product List page and restart the product when you are ready. Select **Products** on the top toolbar to open the page.

Network configuration

This section describes how to manage Transfer CFT network resources.

Overview

The network definition includes internal options for controlling connections, network environment settings, and general environment settings. When applicable, there are examples and tips that describe the effect on performance, the behavior for failed transfers, details about host name resolution, and so on.

Fields

A network definition is comprised of the following fields. These are set by default, but can be changed.

Protocols

Protocols are the access points to Transfer CFTs from the network. There is at least one protocol per Transfer CFT.

Only one interface can be managed in Central Governance.

For each network protocol, two protocol sections can be defined: one without security and one using the business certificate generated upon registration (SSL_DEFAULT). For each protocol section, a protocol object of type PeSIT is created on Transfer CFT; the ID is the concatenation of the network protocol identifier followed by **_<Security level>**. The security level is 1 when no security is used and 2 for SSL_DEFAULT. For instance, a protocol ID is **UDT1_PESIT2** for a protocol relying on UDT network protocol and using the SSL_DEFAULT security profile.

Interface

Indicates the entity type through which connections can be established.

- **Any**: Receive incoming calls on all available addresses.
- **Any IPv4 | Any IPv6**: Use one of these options to define interfaces that can accept requests directed exclusively to all existing IPv4 addresses or IPv6 addresses, respectively.
- **Address**: The address value allows you to define a specific address for Transfer CFT communication. For example, if your host has several IP addresses, you may want to reserve a specific address for the Transfer CFT. To do this select Address, and then enter the address to use for communication. You can enter an IP address or a host name, which is mapped to the designated IP address. Once set, Transfer CFT only accepts requests intended for the defined address.

Protocol

Indicates whether the displayed network protocol is active.

An interface can be linked to up to three network protocol definitions, one for each of the following types: TCP, pTCP and UDT. Two of these, pTCP and UDT, are used specifically for file transfer acceleration (see *About transfer acceleration* on page 174).

To disable a network protocol without losing the defined values, deselect the check-box that precedes the network protocol type (TCP, pTCP or UDT). Disabling deactivates that particular network type, but stores the values for future use.

You must link at least one network protocol to the interface to establish basic network communication.

Ports out

Define up to 16 comma-separated port ranges that can be used by Transfer CFT for outgoing calls, for example "5000-6000, 7251-8000". If you leave the field blank, the default value is used.

Connections (conn.)

Indicates the maximum number of simultaneous connections that Transfer CFT can establish on a given network resource.

PeSIT / SSL

List of available security profiles or no security.

Port in

Indicates the local port for Transfer CFT.

This setting defines the ports where Transfer CFT is listening for connections, and is a mandatory parameter in Central Governance.

General

Maximum simultaneous transfers

Indicates the maximum number of simultaneous connections Transfer CFT can have for a network resource.

You can use this parameter to optimize Transfer CFT bandwidth usage. For example, if Transfer CFT is being used as a relay or central connecting point, then you may want to set this parameter to a high value. Raising this value increases the CPU processing, disk I/O, and bandwidth consumption on the host machine, so check that the host machine has the necessary hardware capability.

Using a low value when Transfer CFT is acting as a relay can cause bottlenecks in the flow's execution. If Transfer CFT receives more transfer requests than authorized by this parameter value, the transfers are listed as described in *Transfer list* on page 189. These transfers are then acted on according to their priority, or when resources become available.

Additional conditions:

- The license for Transfer CFT might impose restrictions that lower the configurable value.
- The number of incoming and outgoing transfers is also defined at the transfer point level.

Disconnect timeout

Indicates the wait timeout for either a response to the protocol connection request, or to the transfer point connection, before disconnecting.

This parameter represents the timeout when waiting for a protocol request, which can be a connection or an interruption request.

At the protocol level, a connection request refers to the physical connection with the transfer point host. Once established, the transfer point Transfer CFT may not be responding. This could be due to the fact that the transfer point application is busy or has encountered an error.

An interruption request occurs if one of the Transfer CFT transfer points, for example CFT1, needs to interrupt the communication in response to a user action. The transfer point CFT1 can then use the time defined in the **Disconnect timeout** parameter for the network connection break. After this period of time, the originator of the abort request, CFT2, will initiate a network disconnect request.

If your Transfer CFT frequently communicates with transfer points that are slow to respond, for example due to heavy usage, you may want to increase the disconnect timeout value. Or, you may want to use a higher disconnect timeout value to allow abort requests from Transfer CFT to have enough time to be correctly processed at the transfer point. Alternately, if you interact with highly responsive transfer points, you can use a lower value. In most cases, the default value of 60 seconds is sufficient.

Attempts to restart transfer

Indicates the maximum number of times Transfer CFT attempts to restart a transfer.

This requester mode parameter indicates the maximum number of attempts for a transfer. An attempt begins at the moment the physical connection is established with the remote site.

You can tune this value in combination with the **Disconnect timeout** parameter. For example, you can set a low disconnect timeout, but define a certain number of retries if the transfer cannot be performed. See *PeSIT Tuning* on page 173.

IPv6 mode

Indicates the IPv6 resolution for hostnames when Transfer CFT is acting as a client, server, both, or none.

This setting defines how Transfer CFT manages Internet addresses and hostname resolution, and is operating system dependent.

- **Client**: The host name used by Transfer CFT to connect to a host may refer

to an IPv4 or an IPv6 address or a list of addresses of either type. If a name simultaneously refers to two different addresses, Transfer CFT will connect to the one available on the remote end.

- **Server**: The host name used by Transfer CFT to listen for incoming connections can refer to both types of addresses or to a list of addresses of either type. If the name resolution request returns a list of addresses Transfer CFT listens for the first entry in the list. The order of returned addresses is not necessarily in any particular order, and is operating system dependent.
- **None**: Because enabling IPv6 use for applications may have adverse effects if improperly configured, support for IPv6 is disabled by default. In this case, hostnames that are defined in the Transfer CFT configuration files are resolved only for IPv4 addresses. That said, you can directly enter IPv6 addresses in Transfer CFT configuration files, as the **IPv6 mode** parameter applies exclusively to hostname resolution.

We recommend **None** when unsure whether your operating system is set up for IPv6.

The operating system must support IPv4 and IPv6 (if used) addressing as defined in the POSIX Protocol Independent API specifications.

Keep alive between transfers

The client and server timeout settings represent separate idle connection parameters when Transfer CFT is acting as a client or a server.

Client

Interval in seconds to maintain an active session between transfer activity on the client.

Server

Interval in seconds to maintain an active session between transfer activity on the server.

pTCP

See *About transfer acceleration* on page 174 for information about using pTCP.

Number of parallel connections

Indicates the number of simultaneous connections that can occur in parallel.

Packet size

Indicates the pTCP packet size in bytes.

Buffer size

Indicates the internal acceleration buffer size in megabytes. This value should be greater than the number of connections multiplied by packet size. The stored buffer size value should be rounded to at most 3 decimal places, with no trailing zeros.

For example, for 16 connections and 4000 bytes the value would be 0.062 MB, if you round up to the nearest 0.001 MB.

UDT

There is one UDT instance per Transfer CFT.

See *About transfer acceleration* on page 174 for information about using UDT.

Buffer size

Indicates the internal acceleration buffer size.

PeSIT Tuning

There is one PeSIT tuning instance per Transfer CFT.

Compression

Defines the use of file compression for Transfer CFT. Files can be compressed to reduce the size of the data that is transferred.

- **Yes**: Authorizes use of compression for sending or receiving a file, which is then negotiated between the source and target. This implies the source and target use the same type of compression. Selecting compression indicates one or multiple types of PeSIT compression are used, which might include horizontal or vertical compression or both.

  With horizontal compression, data are compressed horizontally to identical consecutive characters in the same row.

  With vertical compression, data are compressed vertically to identical consecutive characters in the same column.

  With both, data are compressed horizontally and vertically.
- **No**: Do not use PeSIT file compression.

Consider using compression when the files are highly compressible, or if the network has an especially low bandwidth. Note that using compression leads to high CPU consumption.

Inactivity timeout

The network monitoring timeout, in seconds, excluding the protocol connection/disconnection break phase. The value 0 indicates an infinite amount of time.

For example, transfer points CFT1 and CFT2 are connected and multiple parallel transfers are occurring. If one transfer point, the local transfer point CFT1, does not receive an acknowledgement by the time indicated in the **Inactivity timeout**, it interrupts the connection with a diagnostic code indicating that the timeout was exceeded. This could be caused, for example, by high activity on the remote CFT2 transfer point.

You may want to adjust this parameter if you have a session that is open for an extended period of time.

The Inactivity timeout refers to the file protocol data connection that occurs after a PeSIT connection. This is not to be confused with the Disconnect timeout, which is a request during the actual PeSIT session.

Acknowledgement window size

Maximum number of synchronization points, without waiting for a response, that are authorized by the source/target. This value may be negotiated between Transfer CFTs or applications during the during connection phase.

The value 0 indicates that the synchronization point option is not used. The value 1 defines half-duplex transfers.

Data transferred between sync points

Indicate then internal size of data to transfer, in kilobytes, between two synchronization points. This value is negotiated with the transfer point timing. The value 0 indicates that there are no synchronization points.

About transfer acceleration

Transfer acceleration is a feature of Transfer CFT that enables significantly faster transfer rates for large-file transfers, traveling long distances over high bandwidth networks.

Two technologies are used to optimize transfer acceleration:

- UDT: UDT is a transport protocol that Transfer CFT can use over high-speed networks. UDT uses UDP (User Datagram Protocol), a lower transport layer, for transferring bulk data.
- pTCP: Parallel TCP is an end-to-end transport layer protocol that offers increased performance by using multiple paths, or striped connections, to optimize connections.

Transfer acceleration is not supported on Transfer CFTs running on z/OS and IBM i computers.

You can use pTCP to overcome TCP/IP limitations on high bandwidth, high latency networks, and to improve end-to-end network performance. While TCP/IP extensions can be used to overcome the standard 64KB window limitation, the result is unevenly distributed. Additionally, traditional TCP/IP does not manage packet loss well.

Note pTCP can be implemented in a simple or multi-node architecture.

To decide if you would benefit from using pTCP, multiply your current bandwidth by latency to calculate your existing BDP. The maximum theoretical gain is BDP/64KB. Implementing transfer acceleration is of interest when the BDP is greater than 512KB.

The license key must include the transfer acceleration option.

Bandwidth allocation

This section describes bandwidth allocation configuration.

Overview

You can use the bandwidth allocation fields to manage data rates and the network bandwidth used for incoming and outgoing data in your flows.

See *Transfer CFT bandwidth allocation* on page 269 for more details.

Fields

The user interface indicates default values.

Enable

Enables or disables bandwidth allocation.

Global data rates

Specifies limits on the rates of incoming and outgoing data.

- Unlimited specifies there is no maximum rate for inbound or outbound communications.
- Limited allows setting maximums, in kilobytes per second, for the rates of incoming and outgoing data.

Priority

Additionally, you can specify the priority percentages for the high, medium and low service classes. The system rounds the total of percentages to 100. The system also sorts the percentages to ensure high has the highest percentage and low has the lowest percentage. A service class defines network session parameters that control rate, borrowing and bandwidth pending the class. A service class percentage can be zero, but not less than zero. The default percentages for the service classes are high 80, medium 15 and low 5.

Transfer processing

A transfer is a set of processes that result in the exchange of data between systems, where one system is the source and the other is the target. This type of data exchange between applications is a flow.

Overview

In Central Governance, you can create and deploy flows, which are triggered at the system level.

Transfer CFT provides a complete processing workflow that includes transfer processing, pre- and post-processing scripts, error scripts, and acknowledgment script execution. For more information on defining these in your flow, see *Source processing scripts* on page 339 and *Target processing scripts* on page 354.

The transfer processing configuration fields help to define transfer operations. The following fields include information on the user executing the transfer, sending groups of files, running scripts during the flow, and other general processing options.

Multiple steps are required to define a flow. For more information, see:

- *Composition, deployment, execution* on page 230
- *Flow lifecycle* on page 233
- *General concepts about flows* on page 230

Fields

The user interface indicates default values.

User for file access

Account used to read/write the files to be transferred.

- **Transfer CFT system account** - To read and write files, the Transfer CFT uses the same system account that launched the Transfer CFT.
- **USERID variable** - The user is set at runtime when performing a transfer. For example, the user is the one defined in the CFTUTIL utility SEND or RECV command. On the sender side, the system user who initiates the SEND is used to execute the end-of-transfer procedure by default. On the receiver side, the USERID is specified in the corresponding CFTRECV object. This gives the user referenced by the user ID the rights to execute end-of-transfer procedures.

If you elect **USERID variable**, you must define rights for the account on the Transfer CFT. To enable system users in Unix or Linux, use the CFTSU process and give special rights to the CFTSU executable. Log on as root and execute the following commands:

- `chown root:root $CFTDIRINSTALL/bin/CFTSU`
- `chmod u+s $CFTDIRINSTALL/bin/CFTSU`

User for script execution

Transfer CFT can execute scripts as the system user or on behalf of another user as defined by the USERID option.

This field is not supported on Transfer CFTs running on z/OS computers.

- **Transfer CFT system account** - Transfer CFT executes the script as the system user.
- **USERID variable** - Transfer CFT executes the script on behalf of another defined user, other than the system user.

For an unknown flow

Action to take when the flow is unknown.

Use the system default - The transfer is performed using the default flow identifier.

Reject request - No transfer occurs when the flow is unknown and an error code and message is returned in response to the CFTUTIL utility SEND and RECV commands.

Transmit files individually

Indicates whether the transmission of a group of files is done individually, file by file, or grouped when possible.

The method for transmitting the group of files depends on the operating systems of the Transfer CFTs involved in the flow, and their configurations.

- **When necessary** - When the source and target systems have the same operating system, you can send multiple files as an archive file.
- **Always** - Select this option when you are certain that the operating systems are different. In this case, the transfer always sends files individually for each file in the group. Note that when using the **Always** option, Transfer CFT transmits files individually even if the operating systems are the same.

See *Transferring groups of files* on page 257.

Select **Always** if uncertain about the transfer point platform. For example, a Windows system archives a group of files as a ZIP file, but a receiving UNIX transfer point expects the TAR file format. In this case, an error occurs on both sides.

When requesting all files

Action to perform if any one of the transfers fails.

Defines the desired behavior when a Transfer CFT requests all available files from another Transfer CFT.

- **Stop on error** - The file reception stops at the first error. For example, if there are 10 files to send and an error occurs while transferring the third file in the group, then the remaining files (4 through 10 in our example) are not transferred.
- **Continue** - When using this option, Transfer CFT continues with the request for the remaining files (4 through 10 in our example) even though it did not receive the third file.

Configure default scripts

The following describes the types of processing scripts you can specify on the source and target.

When configuring default scripts you can specify an existing script on the Transfer CFT or upload a new script to it.

If you upload a script file, it is available on the Transfer CFT after the configuration is deployed successfully on it. The script is uploaded with the default path:

```
$(cft.runtime_dir)/conf/ws_upload
```

The new file overwrites an existing file if present.

Select **None** if a processing script is not needed.

Source

You can specify for the transfer source:

- Post-processing script to run after a file is sent.
- Acknowledgment script to run after an acknowledgment is received for a sent file.
- Error script to run after an error occurs when sending a file.
- No script for one or more categories.

For more information, see *Transfer CFT source fields in flows* on page 327.

Target

You can specify for the transfer target:

- Post-processing script to run after a file is received.
- Error script to run after an error occurs while a file is received on the target.
- No script for one or more categories.

For more information on defining targets, see *Transfer CFT target fields in flows* on page 345.

Folder monitoring

Transfer CFT can monitor directories for files to transfer. You can specify the directories to monitor and set up conditions for identifying the files to transfer. Transfer CFT checks at intervals for transfer candidates in specified directories. You can define conditions for triggering SEND commands for qualifying files.

Overview

You can specify directories for Transfer CFT to monitor for files to transfer. Files that meet specified conditions are submitted, or consumed, for transfer. Monitoring of a directory can include or exclude its sub-directories.

Transfer CFT tracks the size and date and time of the last change for each monitored file. If the state does not change within a specified interval, the file can be submitted for transfer.

A monitored directory is the scan directory or scan_dir. For each scan_dir there must be another directory for Transfer CFT to record the state data of each file. This directory is the work directory or work_dir.

For a submitted file, Transfer CFT issues a CFTUTIL SEND command with the flow and partner parameters (IDF and PART, respectively). After the command executes, the file is:

- Moved to a work_dir when Method = Move. A file can be renamed when moved.

  or

- Kept in the monitored scan_dir when Method = File. The file is tracked as submitted through a MET file written to the scan_dir. Files can be resubmitted when changed.

See *Move and File examples* on page 182.

Work_dir and scan_dir cannot refer to the same directory.

Fields

Enable

Specifies whether to enable directory monitoring.

You can specify one or multiple directories to monitor. Click **Add folder monitoring** to specify another directory. You can click the **X** icon to delete the configuration for an added folder.

Folder name

A friendly name for the directory to monitor. Only Central Governance uses this name; it is not deployed to Transfer CFT. Instead, Central Governance deploys folder monitoring indexes to Transfer CFT.

For example, the following monitored directories are defined in Central Governance: Folder A, Folder B, Folder C. Central Governance deploys the following parameters to Transfer CFT:

- `folder_monitoring.enable = Yes`
- `folder_monitoring.folders = 1,2,3`
- `folder_monitoring.folders.1.<parameter_name>`, where `<parameter_name>` corresponds to parameters from Folder A
- `folder_monitoring.folders.2.<parameter_name>`, where `<parameter_name>` corresponds to parameters from Folder B
- `folder_monitoring.folders.3.<parameter_name>`, where `<parameter_name>` corresponds to parameters from Folder C

Directories

Directory to scan

Path and name of the root directory to monitor. In Transfer CFT this is the scan_dir. The directory must exist before the configuration is effective on Transfer CFT.

Directory where files are tracked

Path and name of the directory where files are moved after being transferred or where meta-data tracking files are created for transferred files. In Transfer CFT this is the work_dir. Transfer CFT creates this directory on first use if it does not exist.

Transfer parameters

Flow identifier

Identifier of the flow used in the transfer. The flow identifier must exist on Transfer CFT when files are submitted.

- **Custom** indicates a fixed flow identifier to use for all SEND commands.
- **First sub-folder** indicates the flow identifier is the name of the first directory sub-level in the directory to scan.
- **Second sub-folder** indicates the flow identifier is the name of the second directory sub-level in the directory to scan.

Partner

Partner who receives the file. The partner must exist on Transfer CFT when files are submitted.

- **Custom** indicates a fixed partner name to use for all SEND commands.
- **First sub-folder** indicates the partner name is the name of the first directory sub-level in the directory to scan.
- **Second sub-folder** indicates the partner name is the name of the second directory sub-level in the directory to scan.

Scanning

Scan sub-directories

Indicates whether to only scan the specified directory or also its sub-directories.

Number of files to scan

Maximum number of files to scan for submission per scanning interval.

- **Unlimited** indicates there is no limit on the number of files to scan.
- **Limited** indicates there is a maximum number of files to scan that you specify in the provided field.

Time between scans

Seconds between scans.

Submission

Time before scanned files are submitted

Files that have not changed during this interval can be submitted. This prevents submitting files before they have been written fully to the monitored directory.

Method

Specifies handling of submitted files.

- **Move** indicates files are moved to a work directory (work_dir).
- **File** indicates files are kept in the scan directory (scan_dir) and tracked with a state file. Transfer CFT creates the `*.met` state file in the work directory. You must not delete the file while folder monitoring is enabled.

Append timestamp to submitted files

When Method = Move, indicates whether to append the date and time to the moved file.

Resubmit changed files

When Method = File, indicates whether files kept in the scan directory can be resubmitted if they have changed.

File filters

Include file template

Only files matching the specified pattern are monitored. For example, define `*.txt` to send only files with a TXT extension.

Exclude file template

Files matching this pattern are excluded. For example, use ***old*** to exclude files with **old** in the file names.

The cumulative effect of the two filtering examples is only TXT files without **old** in file names are submitted for transfer. For instance, `invoice.txt` is sent, but `invoice-old.txt` is ignored.

Minimum size

Minimum size of files that can be submitted.

- **Unlimited** indicates there is no minimum size for files.
- **Limited** indicates there is a minimum size for files that you specify in bytes in the provided field.

Maximum size

Maximum size of files that can be submitted.

- **Unlimited** indicates there is no maximum size for files.
- **Limited** indicates there is a maximum size for files that you specify in bytes in the provided field.

Move and File examples

The following examples demonstrate folder monitoring when the selected method is Move or File.

Move

Field	Value
Folder name	Folder A
Directory to scan	/dir_c/scan
Directory where files are tracked	/dir_c/work
Flow identifier (custom)	FlowSendTextFiles
Partner (custom)	MyPartner
Scan sub-directories	Yes
Number of files to scan	2
Time between scans	60 seconds
Time before scanned files are submitted	5 seconds
Method	Move
Append timestamp to submitted files	Yes
Include file template	*.txt
Exclude file template	*old*
Minimum size	1 KB
Maximum size	100 KB

Given the directory and file structure as input:

```
/dir_c/scan/f1.txt 10KB

/dir_c/scan/f2.txt 10KB
```

```
/dir_c/scan/f1-old.txt 10KB

/dir_c/scan/f2-old.txt 10KB

/dir_c/scan/subfolder/f3.txt 50KB

/dir_c/scan/subfolder/f4.txt 150KB
```

Every 60 seconds, a maximum of 2 files are scanned. Submitted files are moved to `/dir_c/`work after being renamed.

First scan

```
CFTUTIL SEND part=MyPartner, idf=FlowSendTextFiles, fname=/dir_
c/work/f1.20140515120155.txt

CFTUTIL SEND part=MyPartner, idf=FlowSendTextFiles, fname=/dir_
c/work/f2.20140515120155.txt

/dir_c/work/f1.20140515120155.txt

/dir_c/work/f2.20140515120155.txt
```

Second scan

```
CFTUTIL SEND part=MyPartner, idf=FlowSendTextFiles, fname=/dir
c/work/subfolder/f3.20140515120255.txt

/dir_c/work/subfolder/f3.20140515120255.txt
```

File

Field	Value
Folder name	Folder B
Directory to scan	/dir_c/scan
Directory where files are tracked	/dir_c/work
Flow identifier	Second sub-folder
Partner	First sub-folder
Method	File

Given the directory and file structure as input:

```
/dir_c/scan/newyork/idf1/f1.txt

/dir_c/scan/newyork/idf2/f2.txt

/dir_c/scan/paris/idf3/f3.txt
```

```
/dir_c/scan/paris/idf3/f4.txt
```

The following SEND commands are issued:

```
CFTUTIL SEND part=newyork, idf=fl1, fname=/dir_
c/scan/newyork/idf1/f1.txt

CFTUTIL SEND part=newyork, idf=fl2, fname=/dir_
c/scan/newyork/idf2/f2.txt

CFTUTIL SEND part=paris, idf=fl3, fname=/dir_
c/scan/paris/idf3/f3.txt

CFTUTIL SEND part=paris, idf=fl3, fname=/dir_
c/scan/paris/idf3/f4.txt
```

Sent files are kept in the scan directory tree, and `*.met` files are created in the work directory for tracking purposes, following the same directory tree pattern:

```
/dir_c/work/newyork/idf1/f1.txt.met

/dir_c/work/newyork/idf2/f2.txt.met

/dir_c/work/paris/idf3/f3.txt.met

/dir_c/work/paris/idf3/f4.txt.met
```

CRONJOBs

CRONJOBs in Transfer CFT are scheduled jobs for performing functions such as periodically scanning a directory for files to transfer, maintenance tasks and other operations.

Overview

The CRONJOB definition includes execution scheduling and upload of the job to Transfer CFT. All CRONJOBs that Central Governance manages are attached automatically to the active CFTPARM via a dedicated CRONTAB named CGCRONTAB created at product registration.

Transfer CFT supports CRONJOBs on all supported operating systems.

The user interface for CRONJOBs within the Transfer CFT configuration page enables you to add, edit and remove CRONJOBs.

Fields

Name

A unique name for the CRONJOB on the Transfer CFT.

Status

Indicates whether the CRONJOB is active or inactive.

Description

A description of the CRONJOB.

File

Indicates whether to use an existing file or upload a new file.

File name

If an existing file, the path to the file.

Choose file

If uploading a file, select the file to upload.

Schedule

Defines when to run the CRONJOB. See *CRONJOB schedule syntax* on page 185 for the schedule rules.

User ID

Identifies the user of the job.

Additional information

The PARM to use in the job execution.

CRONJOB schedule syntax

The following tables describe Transfer CFT CRONJOB time syntax and symbolic variables. See the script execution scheduling topic in the Transfer CFT user guide for more information.

See *CRONJOBs* on page 184 for configuring CRONJOBs in the Central Governance user interface.

Time syntax

Time syntax is case sensitive | a, b, c are integers

Rule	Syntax	Alternate syntax
time	time_element; time time_element	

Rule	Syntax	Alternate syntax
time_element	seconds= content minutes= content hours= content monthdays= content weekdays= content months= content	s= content m= content h= content D= content W= content M= content
content	content_elt, content content_elt	
content_elt	a content_set / c	
		z/OS (MVS) specific syntax
content_set	[a:b] [a:b] [a:] [:b] *	<a:b> <a:> <:b>

Time syntax examples

If you use this syntax	The job is executed
m=30	Once an hour at minute 30
m=1,5,40	3 times an hour for minutes 1, 5, and 40
m=[30:40]/2	Every 2 minutes, of every hour, between the minutes 30 and 40, this means on the minute for 30, 32, 34, 36, 38, 40
m=[:40]	Every minute until, and including, the minute 40
m=[30:]	Every minute starting at and including the minute 30
h=6;m=30	Every day at 06:30:00
D=1;h=15;m=30	Every first day of the month, at 15:30:00

If you use this syntax	The job is executed
D=*/2;h=6	Every 2 days at 06:00:00, day 1, day 3, day 5...day 31. Due to the fact that all months do not have 31 days, the following month the job will be performed 2 days from the 31st, that means on day 2, then day 4, day 6, and so on.
D=[1:7],W=0;h=4	Every first Sunday of the month at 04:00:00
D=-1;h=6	Every last day of the month at 06:00:00
D=[-7:-1];W=3;h=6	Every last Wednesday of the month at 06:00:00
W=0	Every Sunday at 00:00:00
W=1	Every Monday at 00:00:00
M=1	Every January, the first at 00:00:00

CRONJOB symbolic variables

Symbolic variable	Corresponding substituted value
&CFTEVENT	Fixed name=CRONJOB
&CFTNAME	The PART value in the command CFTPARM
&SYSDATE	System date
&SYSTIME	System time
&SYSDAY	Day of the week
&NSUB	Counter for all the Cronjob procedures launched
&ID	Cronjob Identifier
&CRONTAB	Crontab value
&USERID	User identifier indicated in CFTCRON command
&COMMENT	Comment value indicated in CFTCRON command
&PARM	Parm value indicated in CFTCRON command

Symbolic variable	Corresponding substituted value
&SCOUNT	Counter for the Cronjob procedures launched for a specific Identifier

Transfer request mode

Transfer request mode lets you configure the communications medium used by Transfer CFT when applications connect to it.

Overview

Medium refers to any data support or local communications means used by Transfer CFT. This includes media accessed by Transfer CFT server and utility and media used by interactive functions or programming interfaces. Communications can be asynchronous or synchronous.

Central Governance manages only the first Transfer CFT configuration for each mode.

When you change and deploy changes in the transfer request mode configuration, Central Governance pushes the changes to the selected Transfer CFT and removes any extra communications configurations.

Asynchronous communications are based on a file where commands are stored. Transfer CFT scans the file at specified intervals. This mode is recommended if Transfer CFT might be inactive periodically.

Synchronous communications, based on TCP medium, are possible only when Transfer CFT is running. Only clients on the Transfer CFT local host can establish connections.

Asynchronous communications is the default. If you enable both asynchronous and synchronous communications, Transfer CFT always uses asynchronous. Synchronous communications are used only when set explicitly.

Fields

The user interface indicates default values.

Asynchronous

Time between scans

The interval in seconds that Transfer CFT scans the communications file.

Synchronous

The following fields are displayed when synchronous is enabled.

Host

Transfer CFT listening host.

Port

Transfer CFT listening port.

Maximum connections

Maximum allowed number of open connections.

Secured connections

Specifies whether to use the secured version of the request/reply protocol.

Session timeout

Time in seconds before a channel opened by a client is available.

Transfer list

This section describes how to customize and manage a transfer list.

Overview

Transfer lists provide a way to view all transfers in a list format. Each day as you perform transfers, the transfer list records this activity, along with data that is associated with each transfer.

By default the transfer list is purged each day, though this value is customizable. Additionally, you can define retention periods, delete files once the list exceeds a certain size, purge the transfer list at certain times of the day, and so on.

Fields

The user interface indicates default values.

Transfer List

Number of entries in memory

Maximum number of entries in the memory buffer.

Update during transfer

Indicates whether you can update the transfer list while a transfer is occurring.

Sync points between updates

Number of synchronization points that occur during a transfer before updating the transfer list, where 1 indicates constant updates. This field is displayed only when **Update during transfer** is enabled.

Synchronize list file when written

Indicates whether to force the transfer list file to synchronize when Transfer CFT processes write to this list.

Entry Retention

You can define the transfer list purge settings. Additionally, you can refine the transfer list according to whether the transfer was a send or receive, successful or incomplete.

Tip Define retention periods that are balanced between the transfer list size and the time during which you want to be able to act on transfers.

Purge

Indicates whether Transfer CFT should periodically remove older entries in the transfer list.

Interval between purges

Frequency, in days, hours or minutes, to purge the transfer list.

Purge at startup

Indicates whether Transfer CFT should purge the transfer list at start up. The transfer list grows indefinitely when manual purge is selected and startup purge is disabled.

Purge increment ___ entries per step

Indicates how many transfers to delete from the transfer list file per step.

Keep aborted transfers

Indicates whether Transfer CFT should keep aborted transfers or delete them without waiting for a purge.

Retention period

This subsection defines the period to keep a transfer in the transfer list before purging, according to its transfer direction and status.

Completed incoming transfers before purge

Number of days, hours or minutes after which the entries of incoming transfers that were successfully executed are automatically purged.

Incomplete incoming transfers before purge

Number of days, hours or minutes after which the entries of uncompleted incoming transfers are automatically purged.

Completed outgoing transfers before purge

Number of days, hours or minutes after which the entries of outgoing transfers that were successfully executed are automatically purged.

Incomplete outgoing transfers before purge

Number of days, hours or minutes after which the entries of uncompleted outgoing transfers are automatically purged.

Access and security

This section describes access management and security configuration for Transfer CFT.

- Access management refers to the policy that allows users to perform actions on Transfer CFT.
- Security is an option to elect FIPS for Transfer CFT.

Access management options

The following are the available access types and descriptions.

Central Governance

Central Governance administrates access management. Only Central Governance users in an organization with appropriate Transfer CFT privileges can perform operations directly on Transfer CFT, through Copilot or CFTUTIL commands. Additionally, users defined as superusers are not asked for permissions checks for actions performed with CFTUTIL commands.

Transfer CFT uses a built-in cache that allows CFTUTIL users and users authenticated by Copilot to continue being authorized even when Central Governance is unavailable. The authorization-persistent cache on Transfer CFT is updated after access management configuration is changed in Central Governance, meaning, for example, updates occur after a user is created or a user's role assignment is changed. It can take up to 10 minutes before the cache is updated as specified by the Transfer CFT parameter `am.passport.persistency.check_interval`. This parameter, which has a default value of 600 seconds, represents the interval between two checks of access management updates.

The user and password for logging on are not stored in the cache, which means you cannot connect to Copilot when Central Governance is down.

Transfer CFT internal

This is an out-of-the-box access management based on predefined roles and privileges and a group database. Groups and their members are defined in this supplied database. There are two available databases:

- Operating system group database
- xfbadm database (UNIX and Linux only). Use the xfbadmusr and xfbadmgrp utils to manage users, groups and user assignment to groups.

The pre-defined roles are:

- Administrator: Provides full user access
- Application: Allows applications to request transfers and view the catalog
- Designer: Allows you to manage application flows
- Helpdesk: Enables you to view the catalog and log
- Partner Manager: Allows you to manage partners

The system or xfbadm groups are mapped to the predefined roles, so users inherit the privileges.

None

User authorization is based on local users defined in your operating system.

Fields

The user interface indicates default values.

Access Management

Access type

Type of access management to use in Transfer CFT. See *Access management options* on page 191.

The next field, **Create process as user**, applies to all access types. Check the fields for your selected access type:

- *Access type is Central Governance* on page 194
- *Access type is Transfer CFT internal* on page 194

Create process as user

Specifies whether the user who logs on to Copilot server must have system administrator rights.

When set to **No**, Central Governance controls user authentication.

When set to **Yes**, the system on which Transfer CFT is installed controls user authentication. When using Central Governance access management type, this allows system users that are declared as superusers to authenticate even when Central Governance is unavailable. All other users must be known on both the system and Central Governance.

To enable this, the user who is logging on to Copilot must have read-write rights for the Transfer CFT installation directory.

This parameter is not available for Transfer CFT on z/OS (MVS).

Additional configuration is required for system user access.

Define user rights on Windows

Some user rights must be assigned to the user who launched the Transfer CFT GUI server to permit other Windows users to log on. This is true except for the local system account when working in the service mode. The user rights to assign are:

- Adjust memory quotas for a process
- Impersonate a client after authentication (only on Windows 2008)
- Replace a process level token
- Create a token object

To define user rights:

1. In a command window, enter **lusrmgr.msc** to open the system users list. Check available users.
2. In a command window, enter **secpol.msc** to open the Local Security Policy window.
3. Select Security Settings > Local Policies > User Rights Assignment.
4. Double-click the required right.
5. Click Add user or group and define.
6. Close and re-open the Windows session for the changes to become effective.

If using Central Governance access management type, the user who wants to log on the Transfer CFT Copilot must exist both in the Windows system and Central Governance. The Windows system performs the user authentication and Central Governance checks the authorization.

Define user rights on Unix or Linux

To enable system users in Unix or Linux, use the CFTSU process and give special rights to the CFTSU executable. Copy the `cftsu` file outside of the home directory and point to the new path.

1. Log on as root.
2. Copy the file to the new location.

```
cp $CFTINSTALLDIR/bin/cftsu <new_folder>/cftsu
```

3. Change the owner of the file.

```
chown root:root <new_folder>/cftsu

chmod u+s <new_folder>/cftsu
```

4. Use CFTUTIL to set the new folder path.

```
uconfset id=copilot.unix.cftsu.fname, value=<new_
folder>/cftsu
```

Access type is Central Governance

The following fields apply when Central Governance is the access type.

Organization

Central Governance organization for Transfer CFT users. Only users who belong to the selected organization can perform actions on Transfer CFT according to their assigned rights.

Superuser

User IDs of users who have unlimited privileges for Transfer CFT for configuration and transfer execution. These are users who can use Transfer CFT when Central Governance is unavailable using the CFTUTIL command.

These are not necessarily users set up within Central Governance, for instance when using CFTUTIL. However, if you log on with a superuser to Copilot, the user must exist in Central Governance for credential check.

For Unix and Windows the superuser is the user that installs Transfer CFT.

Check permission for transfer execution

Check whether the user has permissions to execute transfers.

When set to **No** a system user is authorized to perform transfers using CFTUTIL even if the user is not declared in Central Governance.

When set to **Yes** a system user is authorized to perform transfers using CFTUTIL only if the user is declared in Central Governance.

Access type is Transfer CFT internal

The following fields apply when Transfer CFT internal is the access type.

Group database

Type of database where group members are defined:

- System is the OS group database (UNIX/Linux, Windows).
- xfbadm is the xfbadmgrp database (UNIX/Linux only).

This field is not supported on Transfer CFTs running on z/OS and IBM i computers.

Admin

Users in these groups can perform all administrative tasks.

Application

User applications from these groups can send transfers.

Designer

Users in these groups can manage flows.

Helpdesk

Users in these groups can view logs, transfers and configuration.

PartnerManager

Users in these groups can add and manage partners.

Security

FIPS

Activates FIPS security.

Visibility

The Transfer CFT visibility feature in Central Governance enables you to view and react accordingly to transfer-related events that occur in your system. Central Governance provides a set of graphical displays that you can use to generate business and technical views of transfer processes.

Using the information provided by Central Governance, you can interpret and react to event-related information. For example, a Middleware Manager can use the monitoring interface to be alerted to and act on a flow that is in error.

Overview

When you modify and deploy changes in the visibility configuration, Central Governance pushes the changes to the selected Transfer CFT.

How it works

Transfer CFT has a native agent that generates messages containing event processing data. The agent sends the messages, according to the configuration definitions, to the monitoring environment. Central Governance extracts this event tracking data and displays it at Flows > Flows Report in the Central Governance user interface.

Main and backup servers

Your configuration can use either a single main server, where all of the transfer events data is stored, or you can incorporate a backup server in your system. The backup server in Central Governance switches to the secondary server should the primary server fail. Note that the backup server functions exclusively as a failover server, and does not replicate the information on the primary server.

About the internal server

An internal visibility server is embedded in the Central Governance system. You can choose to use this server as your main monitoring server, or as the backup server.

About the external server

The external server is not part of the Central Governance system. However, this server may be hosted on the same machine, or separately from Central Governance or your Transfer CFT.

About buffers

There is one buffer per Transfer CFT. This buffer is used to store events when the visibility server is not available. You can define that Transfer CFT stop when the buffer reaches threshold, as described in the Parameters section below.

Fields

The user interface indicates default values.

Enable

Indicates the use of Transfer CFT transfer event tracking, which includes flow monitoring and log file information.

Disabling this option means that not only is there no Transfer CFT event tracking, but also that both the main and backup server options are disabled.

Servers

Main server

Indicates the primary monitoring server. Select Internal to use Central Governance or External to use an external server for visibility.

If you select the External server option, you are prompted to enter the host and port for the server.

- **Host**: Host name or IP address of the main monitoring server.
- **Port**: Port for the main monitoring server.

Backup server

Indicates the use of a backup server for monitoring events if the main server is down. This server is used only if the main server is unavailable.

Select Internal to use Central Governance or External to use an external server for visibility. None indicates there is no backup server.

If you select the External server option, you are prompted to enter the host and port for the server.

- **Host**: Host name or IP address of the backup visibility server.
- **Port**: Port for the backup visibility server.

Note If you select Internal s the main server, you cannot select it as the backup server.

Events

Transfer steps reported

Indicates the Transfer CFT events to send to the Central Governance monitoring.

- **All**: Transfer CFT sends monitoring events each time a transfer enters a new step, such as ready, on hold, transferred, and so on. Additionally Transfer CFT sends an event at each sync point, as well as at the frequency defined in the parameter **Transfer status frequency**.
- **First and last**: Transfer CFT sends events to the Central Governance monitoring when a new transfer request is made and when this transfer is completed, either successfully or in error. When this option is selected, no events are sent for any of the intermediate steps. When you select this setting, the **Transfer status frequency** parameter is disabled.
- **None**: No transfer events are sent to the Central Governance monitoring, and the **Transfer status frequency** parameter is disabled.

Transfer status frequency

Indicate how often Transfer CFT sends events to the monitoring server.

The parameter defines the time value to use when the **Transfer steps reported** option when it is set to **All**. The value set here defines how often Transfer CFT sends events related to an ongoing transfer, and includes updated information such as the amount of data transferred, the percentage of completion, and so on.

Minimum log level

Minimum severity level of the messages to display.

For example, if you select **Warning**, Error and Fatal messages also are displayed.

Buffer capacity

Maximum number of messages in the monitoring buffer.

When buffer is full

Indicates the action when the buffer is full.

- **Drop new messages**: All messages that exceed the buffer capacity are lost.
- **Shut down**: When the buffer is full, Transfer CFT shuts down and cannot continue to perform transfers. This requires user intervention to correct the issue and restart Transfer CFT. Use of this option might cause an interruption in your production environment, notably if you have an extended period of unavailability.

Logging

This section describes how to modify the Transfer CFT log file properties.

Overview

The Transfer CFT log tracks and lists messages and codes related to transfer operations. This information can be used for troubleshooting and monitoring session activity. You can modify the default log values to, for example, use a more precise timestamp, rotate log files at certain times each day, or change the default number of entries in the log file.

Fields

Use the following fields to set the Transfer CFT log properties. The user interface indicates default values.

Entry size

Size in bytes of each entry in the log file. Typically, the default value is adequate.

Timestamp precision

Indicates the preciseness of the time displayed in the log, in seconds, 10 milliseconds, or 100 milliseconds.

File Rotation

Number of files in rotation

Number of log files used in the rotation process.

This field is not supported on Transfer CFTs running on z/OS and IBM i computers.

Daily rotation time

Time of day the log file rotates. Additionally, you can rotate the file depending on the size.

Rotate based on size

Indicates whether log files are rotated based on size.

Every _ KB

If files are rotated based on size, indicates the size to trigger the rotation.

Rotate on stop

Indicates whether log files are rotated when Transfer CFT stops.

Log format

The default log layout for Transfer CFT includes the following standard entry categories.

Date/time

The server date and time of the log entry. Format: YYYY-MM-DD hh:mm:ss.

Level

Level of the log entry. Levels, from highest to lowest verbosity, are INFO, WARNING, ERROR, FATAL.

Code

Code associated with the log message.

Message

Actual log message.

SecureTransport configuration 13

The following topics describe configuring SecureTransports managed by Central Governance.

When SecureTransport registers with Central Governance, it has a default configuration that enables it to operate without changes. However, your needs might require changing the default settings.

You can edit the SecureTransport server configuration to use it in flows. However, you cannot deploy the configuration from Central Governance. When you change the SecureTransport configuration in the Central Governance user interface, you also must change the configuration in SecureTransport before using the SecureTransport in flows where it acts as a server.

When SecureTransport connects to a PeSIT server, it resolves server host names to IP addresses and uses the first resolved address to connect to a remote site. SecureTransport checks whether IPv6 is enabled on the computer. If enabled, it returns an IPv6 address. If not, it returns an IPv4 address. If the resolution of host name is IPv6 mode, the other product also must support IPv6 mode. For example, when SecureTransport tries to communicate with Transfer CFT on an IPv6 address, IPv6 mode must be enabled on Transfer CFT. See *Network configuration* on page 169.

SecureTransport network zones

The following topics describe network zones and server communication profiles and provide steps to add, view, edit and remove them in the Central Governance user interface.

Overview

A network zone represents a logical group of one or more SecureTransport servers or edges and can be:

- The private network zone that specifies the static configuration running on the SecureTransport server.
- The DMZ or network zone that identifies the logical group of SecureTransport edges and the static configuration running on the edges.

Although there can be multiple network zones, there can be only one private network zone per SecureTransport.

Within a network zone are server communication profiles. A profile has the technical details for a client to connect to the SecureTransport server via a specified protocol. A network zone can have multiple communication profiles. Minimally, SecureTransport must have at least one profile to transfer files as defined in flows configured in Central Governance and deployed to the product.

A communication profile requires a unique name. This is the name you can select when defining a protocol in a flow.

The protocol is fixed and cannot be changed once the server communication profile is defined.

Server communication profiles belong to a specific network zone, according to whether they represent a service running on SecureTransport server in the private network zone or SecureTransport Edge in the DMZ.

See the SecureTransport Administrator Guide for more information about SecureTransport server and edges.

Manage network zones and communication profiles

The following topics describe actions you can perform on network zones and server communication profiles in the Central Governance user interface.

Add network zone or communication profile

1. Open the SecureTransport configuration page in edit mode. See *Change SecureTransport configuration* on page 202 for details.
2. Scroll to the bottom of the page and click **Add network zone**. Or, select a network zone and click **Add communication profile**.
3. Complete at least the required fields. Add at least one server communication profile. See *Network zone and server communication profile fields* on page 202 for descriptions of the fields.
4. Click **Save**.

View, edit network zone or communication profile

1. Open the SecureTransport configuration page in view or edit mode. See *Change SecureTransport configuration* on page 202 for details.
2. If editing, change the configuration as needed. See *Network zone and server communication profile fields* on page 202 for descriptions of the fields.
3. Click **Save**.

Remove network zone or communication profile

Removing a network zone removes all the server communication profiles within it. You cannot remove the private network because it represents the SecureTransport server default network zone.

1. Open the SecureTransport configuration page in edit mode. See *Change SecureTransport configuration* on page 202 for details.

2. Locate the network zone or communication profile to remove and click the **X** on the right side of the page.
3. Click **Save**.

Change SecureTransport configuration

Use this procedure to change and save a SecureTransport configuration.

1. Select the **Products** tab to view available products.
2. Click the product name to open its details page. On this page you can click **Edit** and change details about the product, including groups tags, description and contact information. However, you must go to the next step to change the product configuration.
3. Click **Configuration** to open the Configuration page and click **Edit** to open edit mode.
4. Select a network zone to edit. Or, scroll to the bottom and click to add a communication profile or network zone. See *SecureTransport network zones* on page 200 for more information.
5. Change the configuration as needed.
6. Click **Save**.

Network zone and server communication profile fields

The following are the fields and descriptions for SecureTransport network zones and server communication profiles. You use these fields when adding or editing these objects in the Central Governance user interface. See *Change SecureTransport configuration* on page 202 or *SecureTransport network zones* on page 200 for how to access the fields.

Network zone fields

Name

The unique identifier of the network zone. The name is configurable only for zones in the DMZ. The first network zone is always the private zone and cannot be renamed.

Host

The list of SecureTransport servers or edge hosts represented by the host name or IP address.

FQDN

The fully qualified domain name or IP address used by partners and applications to connect to SecureTransport. In a cluster environment, it represents the address of the load balancer, which is the entry point for the SecureTransport environment.

Server communication profile fields

Name

The name of the server communication profile.

Protocol

The protocol for the server communication profile.

Edges

Comma-separated list of active SecureTransport Edge servers in the server communication profile. This field is displayed only for network zones in the DMZ and not for the private network zone.

The following are the server communication profile fields by protocol.

PeSIT

Port

Port on which the server listens for connection requests.

Network protocol

Indicates the network protocol.

- **TCP** - The Transmission Control Protocol (TCP), one of the core protocols of the Internet protocol suite (IP), is often called TCP/IP. TCP provides reliable, ordered and error-checked delivery of a stream of octets between programs running on computers connected to a local area network, intranet or the public Internet. It resides at the transport layer.
- **pTCP** - The parallel Transmission Control Protocol (pTCP) is an end-to-end transport layer protocol that supports striped connections.

PeSIT login

User name for connecting to the server.

Password and confirm password

Password for connecting to the server.

Enable SSL/TLS

Indicates whether the connection is secured via SSL or TLS.

Client authentication required

If SSL/TLS is enabled, indicates whether to use the client's public key certificate to authenticate the client to the server.

Yes means the server and the client must be authenticated.

No means only the server must be authenticated.

Optional means the server and the client must be authenticated, and the server tolerates an invalid certificate.

Private certificate

If SSL/TLS is enabled, upload a file containing the server's private key certificate or reuse an existing certificate. For a new certificate, get the certificate file from the server administrator. Supported file types are P12 and PFX.

For a new certificate, specify an alias for it. This enables you to use the same certificate in multiple profiles. The certificate alias must be unique at the product configuration level.

The user interface warns if you try to add a duplicate alias.

For details see *Certificates for HTTP, FTP, PeSIT* on page 105.

Enable FIPS transfer mode

When SSL/TLS is enabled, indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

SFTP

Port

Port on which the server listens for connection requests.

Client authentication

Indicates the method for authenticating clients to the server.

- **Password** - The user name and password for connecting to the server is used to authenticate the client.
- **Public key** - The client's public key is used to authenticate to the server.
- **Password or public key** - The public key or password can be used by clients to authenticate to the server.

Server encryption

Upload a file containing the private key or select an existing key. SecureTransport uses this key to encrypt the SSH-FTP channel.

For a new key, specify an alias for it. This enables you to use the same certificate in multiple profiles.

The user interface warns if you try to add a duplicate alias. Aliases are unique by the objects related to them. For example, an alias for a partner certificate must be unique for a specific partner, but the same alias could be used for another partner.

Enable FIPS transfer mode

When SSL/TLS is enabled, indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

FTP

SecureTransport supports explicit security only. This is why there is no security setting in the user interface. With explicit security, the initial connection is unencrypted. To establish the secure link, explicit security requires the FTP client to issue a specific command to the FTP server after establishing a connection. The default FTP server port is used.

Port

Port on which the server listens for connection requests.

Passive port range

SecureTransport has activated both active and passive mode. For the passive mode, specify the range of ports for the server to listen for connections.

Enable SSL/TLS

Indicates whether the connection is secured via SSL or TLS.

Client authentication required

If SSL/TLS is enabled, indicates whether to use the client's public key certificate to authenticate the client to the server.

Yes means the server and the client must be authenticated.

No means only the server must be authenticated.

Optional means the server and the client must be authenticated, and the server tolerates an invalid certificate.

Private certificate

If SSL/TLS is enabled, upload a file containing the server's private key certificate or reuse an existing certificate. For a new certificate, get the certificate file from the server administrator. Supported file types are P12 and PFX.

For a new certificate, specify an alias for it. This enables you to use the same certificate in multiple profiles. The certificate alias must be unique at the product configuration level.

The user interface warns if you try to add a duplicate alias.

For details see *Certificates for HTTP, FTP, PeSIT* on page 105.

Enable FIPS transfer mode

When SSL/TLS is enabled, indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

HTTP

Port

Port on which the server listens for connection requests.

Enable SSL/TLS

Indicates whether the connection is secured via SSL or TLS.

Client authentication required

If SSL/TLS is enabled, indicates whether to use the client's public key certificate to authenticate the client to the server.

Yes means the server and the client must be authenticated.

No means only the server must be authenticated.

Optional means the server and the client must be authenticated, and the server tolerates an invalid certificate.

Private certificate

If SSL/TLS is enabled, upload a file containing the server's private key certificate or reuse an existing certificate. For a new certificate, get the certificate file from the server administrator. Supported file types are P12 and PFX.

For a new certificate, specify an alias for it. This enables you to use the same certificate in multiple profiles. The certificate alias must be unique at the product configuration level.

The user interface warns if you try to add a duplicate alias.

For details see *Certificates for HTTP, FTP, PeSIT* on page 105.

Enable FIPS transfer mode

When SSL/TLS is enabled, indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

Policies

14

A policy represents common configuration settings for multiple Transfer CFTs. You can simultaneously deploy the same configuration changes to all Transfer CFTs assigned to a policy. For example, if multiple Transfer CFTs use parallel TCP (pTCP), you can create a policy with this configuration and deploy it to the Transfer CFTs.

The Policy List page lets you add and manage the configuration policies defined in Central Governance. You can see the list of policies and the number of Transfer CFTs assigned to them. The page also reports the status of the policy, which indicates where the policy is in its lifecycle. In addition, you can add, deploy and remove policies.

Policy lifecycle

Various statuses are displayed during phases of a policy lifecycle. When a policy is first created, the status is displayed on the Policy List and policy detail pages. Once you start to deploy the policy, the statuses also are displayed on the Deployment List page.

Phases

This topic describes each phase in a policy lifecycle and the effect of the phase on the configuration of Transfer CFT.

The date format has two variants. All of the examples use the current day message format.

- *<Status> at <time> today* when today.
- *<Status> on <date>*when not today.

Creation

Add a policy that contains the configuration values you want to deploy. The status of the policy is *Saved at <time> today*.

Assignment

Assign, or link, the policy to individual Transfer CFTs. See *Status changes* on page 208 for information about the statuses that can be displayed during this phase.

- When a policy is assigned, the configuration of each Transfer CFT is saved in Central Governance with the values defined in the policy. The policy status is Saved, not deployed or *Deployed to <n> of <n>*.

- When an existing policy is updated and saved, the configuration of each Transfer CFT assigned to the policy is automatically updated locally with the changes. This means that the change was made only in Central Governance, and not on the Transfer CFT.
- When a policy is unlinked, the Transfer CFT configuration retains the changes that were applied when the policy was assigned.

Deployment

Deployment pushes the saved changes to each Transfer CFT assigned to the policy. During and after deployment, status messages report the success or failure of deploying to the Transfer CFTs. For example:

- *Deploying since <time> today (2 in progress)* indicates the policy is being deployed to two Transfer CFTs.
- *Deployed at <time> today (2 deployed)* indicates the policy was deployed successfully to the two Transfer CFTs assigned to the policy.
- *Deployed at <time> today (1 deployed, 2 in error)* indicates the policy was deployed successfully to one Transfer CFT but failed to deploy to two other Transfer CFTs.

You can get more information about policy deployments statuses by selecting **Administration> Deployments** and clicking **In error** or **Policies**.

Modification

Edit the policy and save it. See *Status changes* on page 208 for information about the statuses that can be displayed during this phase.

Removal

Remove the policy from Central Governance.

- When a policy that was deployed is removed, the configuration changes that were applied to the Transfer CFT remains unchanged.
- When a policy that was only assigned, but not deployed, is removed, the configuration of the Transfer CFT in Central Governance is that of the policy. Since the policy was never deployed, there are no changes to the actual Transfer CFT configuration.

Status changes

The policy status changes when you perform an action on a policy. The following table illustrates how the policy status is affected when a user assigns, unassigns or edits a policy after its creation. All of the examples use the current day message format.

Policy status before	User action	Policy status after
Saved at <time> today	Assign policy	Saved at <time> today, not deployed
	Edit policy	Saved at <time> today
Saved at <time> today, not deployed	Assign policy	Saved at <time> today, not deployed
	Unassign policy	Saved at <time> today, not deployed
	Edit policy	Saved at <time> today, not deployed
Deployed to <n> of <n'> Transfer CFTs at <time> today	Assign policy	Deployed to <n> of <n'+1> Transfer CFTs at <time> today
	Unassign policy	Deployed to <n-1> of <n'-1> Transfer CFTs at <time> today
	Edit policy	Saved at <time> today, not deployed
Deployed at <time> today	Assign policy	Deployed to <n> of <n+1> Transfer CFTs at <time> today
	Unassign policy	Deployed at <time> today
	Edit policy	Saved at <time> today, not deployed

Business lifecycle scenario

The following table illustrates the phases a policy goes through and the status changes that occur in a store-to-corporate business use case. Multiple stores in different regions require the same configuration. The stores communicate with the product catalog application at corporate headquarters. All of the examples use the current day message format.

Business need	Policy status before	User action	Policy status after
All stores must be able to communicate with the corporate application in a common way		Create policy	Saved at <time> today
All stores must be configured in the same way, for example, by using the same protocol	Saved at <time> today	Assign policy	Saved at <time> today, not deployed
To meet a short time window to exchange files, must optimize throughput, for example, by configuring file transfer acceleration	Saved at <time> today, not deployed	Edit policy	Saved at <time> today, not deployed
Save time and money by deploying the configuration to all stores at one time	Saved at <time> today, not deployed	Deploy policy	Deployed at <time> today
Due to expansion into new region, add stores communicating with the corporate application	Deployed at <time> today	Assign policy	Deployed to <n> of <n+n'> Transfer CFTs at <time> today
Save time and money by deploying the configuration to all new stores at one time	Deployed to <n> of <n+n'> Transfer CFTs at <time> today	Deploy policy	Deployed at <time> today
Close some stores that are not performing	Deployed at <time> today	Unassign policy	Deployed at <time> today

Manage policies

Use the following procedures to manage policies.

Add a policy

1. Select **Products > Policies** to open the Policy List page.
2. Click **Add policy** to open the Add Policy page.
3. In the Policy Information section, enter a name and, optionally, tags and a description.
4. In the left menu, select the section of the configuration to provide new values.

You can click the help link for each section to open a configuration topic that maps to the target section.

5. Click the appropriate icon next to the field you want to modify.

   One icon sets the value so that it can be overridden and the other locks the value. When a field is locked, you cannot edit it on the Transfer CFT configuration page. This applies to all assigned Transfer CFTs.

   The field is now editable.

6. If a default value is displayed in the field, accept it or enter a different value.
7. You can change as many sections as you want for the policy.
8. Click **Save**.

Assign a policy

When assigning a policy, you link it to one or more Transfer CFTs. The configuration values in the policy are applied immediately and saved in Central Governance. Changes to the Transfer CFTs only occur when the policy is deployed. The Policy List page provides access to all members of a policy.

You can assign only one policy at a time to Transfer CFT.

1. Select **Products** on the top toolbar to open the Product List page.
2. Select one or more Transfer CFTs and click **Assign policy**.
3. On the Assign Policy dialog, select the policy to assign and click **Apply**.

To unassign a policy, select **None** and click **Apply**.

Unassigning a policy disassociates it from the Transfer CFT, but there is no effect on the configuration itself. All policy values are retained in Central Governance and no changes are made to the Transfer CFT configuration.

Deploy a policy

After a policy is assigned, you can deploy it. The policy is deployed only on the assigned Transfer CFTs on which it is not deployed already.

1. Select **Products > Policies** to open the Policy List page.
2. Select the policy to deploy and click **Deploy**. In the dialog, confirm the deployment. The affected Transfer CFTs are restarted after the policy is deployed.

The status of the policy changes as the deployment progresses. If a deployment fails on one or more systems, an error status is displayed. When this happens, check the in-error section on the Deployment List page at **Administration > Deployments**. Also check the message column of the Core Services log for information about the cause of the failure. A link to view the log is at **Administration > Services**.

View, edit a policy

When you edit a policy:

- A changed value applies to all assigned Transfer CFT configurations.
- If a value is overridden, it overwrites the Transfer CFT configuration value.
- Transfer CFT values not defined or changed in a policy remain unchanged on Transfer CFT.

1. Select **Products > Policies** to open the Policy List page.
2. Click the name of a policy to open its details page.
3. Click **Edit** to open its edit page.
4. Click **Save** when you are done making changes.

Copy a policy

A duplicate policy is created when you copy a policy, but no Transfer CFTs are assigned to the copy until you do so.

1. Select **Products > Policies** to open the Policy List page.
2. Click the name of a policy to open its details page.
3. Click **Copy** to make a copy of the policy.
4. Change the policy as you like and click **Save** when done.

When you copy a policy, Central Governance gives the copy a name in the format:

```
<original policy name> - Copy<n>
```

Where `<n>` is the number of the copy. The first time a policy is copied, `n` = 1, the second time `n` = 2 and so on.

The copy also is given a default description in the format:

```
Policy copied from <original policy name>
```

You can keep or change the default name or description of the copy.

If you make a copy of a copy, the default name of the subsequent copy is in the format:

```
<original policy name> - Copy<n> - Copy<n>
```

Remove a policy

A removed policy is unassigned from Transfer CFT, but no changes are made to the Transfer CFT configuration.

1. Select **Products > Policies** to open the Policy List page.
2. Select the policy or policies to remove and click **Remove**. In the dialog, confirm the removal.

Applications 15

An application is the logical representation of a business software application that is the true sender or true receiver in a file exchange. An application represents a back-end enterprise resource planning system such as SAP or PeopleSoft.

File-exchange processes are specified in flows. You must add applications in Central Governance before you can define flows involving applications.

You can view and manage applications on the Application List page in the Central Governance user interface. You can filter the list and perform other tasks on the page.

Manage applications

The following topics describe how to add, view, edit and remove applications in Central Governance.

Adding an application in the user interface makes it known to Central Governance. Adding applications is a prerequisite to defining flows involving them and being able to exchange files.

An application can be associated with multiple registered products; the products can be on the same or different hosts. One or more products in the application can be assigned for use within a single flow. And you can have another flow that uses the same application as the source or target, but that specifies different products in the application. See *Defining flows* on page 272 for more information.

Multiple products and multiple types of products on the same or different hosts are allowed within the same application because all have similar folder-monitoring behavior. All of these products can consume files from a directory and produce files to a directory.

Add an application

1. Click **Applications** on the top toolbar to open the Application List page.
2. Click **Add application**.
3. Do the following:
    a. Enter a name for the application. Typically, use a unique name, but you can reuse a name provided the associated product is on a different computer. Central Governance checks whether an application name is valid by name-host pairs.
    b. Specify the computer that hosts the application. You can provide a computer name or an IP address. You can find the hosts of registered products by clicking **Products** on the top toolbar and checking the Product List page.

When you enter a host value, Central Governance detects all registered products on the computer and lists their names in the Products field. For example, if instances of Transfer CFT and SecureTransport are on the computer and are registered in Central Governance, the names of all three are added to the Products field.

c. If necessary, edit the Products field. For example, although two registered products might be on the same host, you want only one of the products to be associated with the application. Also, you might want to add the name of a registered product that is on a different host than the one specified in the Host field.

4. In the optional Details area:
   a. Enter a group name if you want the application to be a member of a group. To associate an application to multiple groups, use a comma to separate group names.
   b. Enter any tags to help you identify or categorize the application.
5. In the optional Contact area, enter information about the primary business contact for the application.
6. Click **Save application** to add it and return to the Application List page. The products and product types associated with the new application are listed in the Products and Product Types columns.

View or edit an application

The details page for an application enables you to review more information than is displayed on the Application List page.

Click **Applications** on the top toolbar and then click the name of an application to open its details page.

On the details page, you can:

- Click **Edit** to edit any fields, such as:
  - The host associated with the application
  - Contact information
- Click **Remove** on the right side of the page to remove it. See the next topic.

Remove an application

You can remove an application from Central Governance by:

- Selecting the application on the Application List page and clicking **Remove**.
- Clicking **Remove** when viewing the application details page.

If the application is used in flows, a confirmation message is displayed. If you confirm the removal, the application is removed from the flows and the status of the flows is recalculated. You must redeploy the flows manually.

Application groups

You can organize multiple applications that have a common purpose into application groups. For example, group all stores in a region so all can be members of the same flow.

Groups do not have to be predefined. You can assign an application to one or more groups when you add or edit the application. If those groups do not exist, they are created.

Alternatively, you can make group assignments and add groups from the Application List page. You also can create groups directly from the page

Flow management

Grouping applications eases the task of flow definition and management. For example, if your organization needs to exchange files between corporate headquarters and stores located in a geographic region, you can create a single group that contains the applications running at those stores. When you define the flow for the exchange, the flow target is the entire group. If a new store opens, you can add it to the group, and the store is automatically included in the flow and receives the exchanges that all other stores in the group receive.

If you remove an application from a group that is used as a source or target in a flow, the application or the system no longer sends or receives the exchanges.

Central Governance supports using, as sources and targets, application groups containing applications linked only to Transfer CFTs and not to any other registered products.

Assign applications and add group at same time

1. Click the **Applications** tab to open the Application List page.
2. Select one or more applications to assign to a group and click **Grouping**.
3. Select the group and click **Apply**.
4. If the group does not exist, click **Add group**.
5. Enter a name, select the check box if it is not already selected, and click **Apply**.

Add group and add applications to it

1. Click the **Applications** tab to open the Application List page.
2. Click **Grouping**.
3. Click **Add group**.
4. Enter a name and click **Apply**.
5. You can now select applications from the list to add to this group.

Remove application from group

1. Click the **Applications** tab to open the Application List page.
2. Select one or more applications.
3. Click **Grouping**.
4. Deselect the check box next to the group name.
5. Click **Apply**.

Manage application groups

The Application Group List page enables viewing all application groups and performing actions on groups: add, edit, remove. You can see the number of application members assigned to a group. The page also lists the product types represented within groups, which is helpful because you can associate different types of products with one application.

Access the Application Group List page by:

- Selecting **Applications > Groups** on the top toolbar
- On the Application List page, clicking **Grouping > Manage groups**.

Add a group

1. Click **Add application group** on the Application Group List page.
2. Enter a unique name, and optionally, tags and a description.
3. Click **Save application group**.

Edit a group

1. Click a group name.
2. Click **Edit**.
3. Make your changes.
4. Click **Save application group**.

Remove a group

1. Select the group to remove.
2. Click **Remove**.
3. Click **Remove application group** to confirm.

View group members

In the Members column, click a number to view the group members. The number represents the total group members. The Application List page is displayed and lists the group members. Note the Filter field above the table shows the list of applications is filtered by the group name.

Groups 16

Groups enable you to organize and manage similar items. Groups are collections of objects with a common purpose or characteristics. You can group products, unmanaged products and applications. The main use cases for grouping items are for configuration management and flow management.

See *Application groups* on page 215 for more information.

Grouping products

Grouping products enables you to deploy configurations and perform operations all together. For example, perform operations such as starting or stopping from the command line interface by specifying the group of products.

Things to know

Groups must contain only like items, such as applications or products. A group cannot contain both applications and products.

A group cannot be a sub-group of another group.

An object can be added to a group and later removed.

An object can belong to one or more groups.

Partners 17

Partners represent entities such as companies that send or receive business data in file transfers governed by Central Governance flows. Partners can use third-party products or Axway products not registered in Central Governance to communicate with other parties over supported protocols.

Partners can be sources or targets in Central Governance flows. They also can be in client or server roles, depending on whether the partners initiate transfers. Partners support multiple types of communication protocols.

Partner objects in Central Governance consist of:

- General information that includes the partner name, tags, description and contact information such as street and email addresses.
- Server communication profiles that specify the protocols for secure or unsecure connections to partners. See *Communication profiles* on page 236.

You must add partners in Central Governance before you can use them in flows.

Manage partners

Use the following procedures to manage partners.

View list of partners

Select **Partners** to open the Partner List page. The page lists all partners and details about them.

You can:

- Filter the list of partners by conditions such as name, email address and so on.
- Click the name of a partner to view or edit its details.
- Add or remove a partner.

Add partner

1. Select **Partners > Add partner** to open the Add Partner page.
2. Complete at least the required fields. Optionally, you can add one or more tags. See *Partner fields* on page 220.
3. Click **Save** to add the partner.

View, edit partner

1. Select **Partners** to open the Partner List page.
2. Click the name of a partner to view its details.
3. Click **Edit** on the details page.
4. Enter changes as needed. See *Partner fields* on page 220.
5. Click **Save**.

Remove partner

Select **Partners** to open the Partner List page, select one or more partners and click **Remove**.

This removes the partner from the flow or flows where it is used. The flows are then recalculated to reflect the new status following this change.

Remove a partner server communication profile

You can remove a server communication profile without removing the partner. Select **Server communication profiles** and click Edit. Scroll to the name of the profile to remove and click **X** in the right margin. One of the following can occur when trying to remove it:

- The remove is forbidden: This means the partner server communication profile is used in one or more flows in flow protocol and you cannot proceed.
- The remove is allowed: The partner communication profile can be removed since it is not associated with any flows.

Partner fields

The following are the fields for adding or editing partners in the Central Governance user interface. See *Manage partners* on page 219 for actions you can perform.

General information

Only the partner name field is required in the general information section for a partner. All other fields — tags, description, address, phone, email — are optional.

Server communication profiles

A server communication profile contains the technical details for a client to connect to the partner's server via a specified protocol. A partner can have multiple communication profiles. Minimally, a partner must have at least one profile to transfer files as defined in flows configured in Central Governance and deployed to products.

A communication profile requires a unique name. This is the name you can select when defining a protocol in a flow. Optionally, you can specify tags and a description.

The user interface warns if you try to add a duplicate profile name. Names are unique by the objects related to them. For example, a name for a partner profile must be unique for a specific partner. But the same name could be used for another partner.

The following describes the fields by protocol. To add multiple profiles, click **Add communication profile**.

PeSIT

Network protocol

Indicates the network protocol.

- **TCP** - The Transmission Control Protocol (TCP), one of the core protocols of the Internet protocol suite (IP), is often called TCP/IP. TCP provides reliable, ordered and error-checked delivery of a stream of octets between programs running on computers connected to a local area network, intranet or the public Internet. It resides at the transport layer.
- **pTCP** - The parallel Transmission Control Protocol (pTCP) is an end-to-end transport layer protocol that supports striped connections.
- **UDT** - The UDP-based Data Transfer Protocol (UDT) is a high performance data transfer protocol designed for transferring large volumetric data sets over high-speed wide-area networks.

Host

URL of the PeSIT server. You can specify multiple hosts separated by commas.

Port

Port on which the server listens for connection requests.

PeSIT login

User name for connecting to the server.

Password and confirm password

Password for connecting to the server.

Enable SSL/TLS

Indicates whether the connection is secured via SSL or TLS.

Client authentication required

If SSL/TLS is enabled, indicates whether to use the client's public key certificate to authenticate the client to the server.

Yes means the server and the client must be authenticated.

No means only the server must be authenticated.

Optional means the server and the client must be authenticated, and the server tolerates an invalid certificate.

Server authentication

If SSL/TLS is enabled, upload a file containing the server's public key certificate or reuse an existing certificate. For a new certificate, get the certificate file from the server administrator. Supported file types are DER, CER, CRT and P7B.

For a new certificate, specify an alias for it. This enables you to use the same certificate in multiple profiles. The certificate alias must be unique at the product configuration level.

Enable FIPS transfer mode

When SSL/TLS is enabled, indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

SFTP

Host

The IP address or fully qualified domain name of the server. You can specify multiple hosts separated by commas.

Port

Port on which the server listens for connection requests.

Client authentication

Indicates the method for authenticating clients to the server.

- **Password** - The user name and password for connecting to the server is used to authenticate the client.
- **Public key** - The client's public key is used to authenticate to the server.
- **Password or public key** - The public key or password can be used by clients to authenticate to the server.

Server verification

Upload a file containing the public key or select an existing key.

For a new key, specify an alias for it. This enables you to use the same certificate in multiple profiles. The alias must be unique at the product configuration level.

The user interface warns if you try to add a duplicate alias. Aliases are unique by the objects related to them. For example, an alias for a partner certificate must be unique for a specific partner, but the same alias could be used for another partner.

Enable FIPS transfer mode

When SSL/TLS is enabled, indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

FTP

Host

The IP address or fully qualified domain name of the server. You can specify multiple hosts separated by commas.

Port

Port on which the server listens for connection requests.

Connection mode

Indicates whether the connection mode is active, passive or both. The server initiates the session in active mode, while the client initiates in passive mode. Both means the server or client can initiate the session.

Port range

If the connection mode is passive or both, specify the range of ports for the server to listen for connections.

Enable SSL/TLS

Indicates whether the connection is secured via SSL or TLS.

Client authentication required

If SSL/TLS is enabled, indicates whether to use the client's public key certificate to authenticate the client to the server.

Yes means the server and the client must be authenticated.

No means only the server must be authenticated.

Optional means the server and the client must be authenticated, and the server tolerates an invalid certificate.

Security mode

Indicates whether the security mode is explicit or implicit.

FTP supports two methods to accomplish security through a sequence of commands passed between two computers. The sequence is initiated with explicit (active) or implicit (passive) security.

- **Explicit security**. The initial connection is unencrypted. To establish the secure link, explicit security requires the FTP client to issue a specific command to the FTP server after establishing a connection. The default FTP server port is used.
- **Implicit security**. Implicit security begins with a secure connection as soon as the FTP client connects to an FTP server. The FTP server defines a specific port for the client to be used for secure connections.

Server authentication

If SSL/TLS is enabled, upload a file containing the server's public key certificate or reuse an existing certificate. For a new certificate, get the certificate file from the server administrator. Supported file types are DER, CER, CRT and P7B.

For a new certificate, specify an alias for it. This enables you to use the same certificate in multiple profiles.

The user interface warns if you try to add a duplicate alias. Aliases are unique by the objects related to them. For example, an alias for a partner certificate must be unique for a specific partner. But the same alias could be used for another partner.

Enable FIPS transfer mode

When SSL/TLS is enabled, indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

HTTP

Server address format

Specify the server with a URL or a host-name pair.

URLs

If the URL format is selected, enter one or more URLs separated by commas. The expected pattern for a URL is: `http://{host}:{port}`

Host

If the host-port format is selected, enter one or more computer names separated by commas.

Port

If the host-port format is selected, enter the port that the specified server or servers listen for connections. You can enter only one port number.

HTTP methods

Select one or more methods for transferring data:

PUT requests the enclosed entity be stored under the supplied URI. If it refers to an existing resource, the URI is changed. If the URI does not point to an existing resource, the server can create the resource with that URI.

POST requests the server accept the entity enclosed in the request as a new subordinate of the web resource identified by the URI. The data POSTed might be, for example, an annotation for existing resources.

GET requests a representation of the specified resource. Requests using GET should only retrieve data and should have no other effect.

Enable SSL/TLS

Indicates whether the connection is secured via SSL or TLS.

Client authentication required

If SSL/TLS is enabled, indicates whether to use the client's public key certificate to authenticate the client to the server.

Yes means the server and the client must be authenticated.

No means only the server must be authenticated.

Optional means the server and the client must be authenticated, and the server tolerates an invalid certificate.

Server authentication

If SSL/TLS is enabled, upload a file containing the server's public key certificate or reuse an existing certificate. For a new certificate, get the certificate file from the server administrator. Supported file types are DER, CER, CRT and P7B.

For a new certificate, specify an alias for it. This enables you to use the same certificate in multiple profiles.

The user interface warns if you try to add a duplicate alias. Aliases are unique by the objects related to them. For example, an alias for a partner certificate must be unique for a specific partner. But the same alias could be used for another partner.

Enable FIPS transfer mode

When SSL/TLS is enabled, indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

Unmanaged products 18

Unmanaged products are systems that are not registered in Central Governance, but that are integrated in flows for transferring files. Unmanaged products can be:

- Transfer CFTs 3.1.2 or later that are not registered with Central Governance
- SecureTransports 5.3.1 or later that are not registered with Central Governance
- Earlier versions of Transfer CFT or SecureTransport that cannot register with Central Governance
- Axway products other than Transfer CFT or SecureTransport
- Third-party products

Unlike registered products, Central Governance cannot detect, start, stop or change the configurations of unmanaged products. However, the Central Governance user interface provides a way to define unmanaged products and include them in flows.

In Central Governance, unmanaged products can be defined in flows as a source, target or relay. Central Governance deploys flows on registered products. When unmanaged products are used in flows, however, the flow definitions must be applied externally to those products.

Certain Central Governance predefined user roles allow users access to unmanaged products.

- The predefined Middleware Manager role lets users view unmanaged products.
- The predefined IT Manager role lets users view, add, change and delete unmanaged products.

Use Unmanaged Products page

Use the Unmanaged Products page to administrate products that are known to Central Governance, but are not managed by it.

Select **Products > Unmanaged Products** to open the page.

On the page you can:

- View a list of all unmanaged products. Use the **Filter** button to filter the list according to selected criteria.
- Click **Add unmanaged product** to add one. See *Add unmanaged product* on page 227.
- Click the name of a product to view its details page. See *View unmanaged product* on page 227.
- Select one or more in the list and click **Remove** to delete.

Add, view, edit unmanaged products

Use the following procedures to add and edit unmanaged products.

Add unmanaged product

1. Select **Products > Unmanaged Products** to open the Unmanaged Products page.
2. Click **Add unmanaged product** to open the Add Unmanaged Product page.
3. Complete at least the required fields. See *Unmanaged product fields* on page 227 for details.
4. Click **Save unmanaged product** to add it.

View unmanaged product

1. Select **Products > Unmanaged Products** to open the Unmanaged Products page.
2. Click the name of a product to open its details page. See *Unmanaged product fields* on page 227 for field descriptions.

Edit unmanaged product

1. Select **Products > Unmanaged Products** to open the Unmanaged Products page.
2. Click the name of a product to open its details page.
3. Click **Edit** to open its edit page.
4. Change values and click **Save changes** when done. See *Unmanaged product fields* on page 227 for details.

Changing some values, such as host or protocol, triggers Central Governance to check whether an unmanaged product is used in a flow. If so, a message advises that changing the product also changes the flow and prompts whether you want to continue.

Unmanaged product fields

The following are the fields for unmanaged products. You only need complete required fields. Required fields are identified on the user interface page.

Name

Unique name of the unmanaged product.

The name must be unique among all unmanaged products in Central Governance. For instance, you can register a product with the same name as an unmanaged product if you plan to migrate flows to the new product instead of using the old unmanaged product.

Host

A single host or multiple hosts separated by commas. Host name must have alphanumeric characters, but the following special characters are allowed:

. period
: semicolon
_ underscore
- hyphen

Type

You can select an Axway product as the type. Or, select **Custom** for any product not listed and enter the name in the provided field.

Version

Product version.

Operating system

Operating system of the computer where the product is installed.

Protocol

There are three supported network protocols. At least one must be defined. The types are TCP, pTCP and UDT. All three are associated with TCP, but pTCP and UDT are used specifically for file transfer acceleration.

PeSIT login

The identifier for connecting to the PeSIT server. The PeSIT login must be unique among all unmanaged products in Central Governance. However, it can be the same as the PeSIT login for a registered product.

PeSIT password and confirm password

The password for connecting to the PeSIT server. The password might be required or optional, depending on the requirements of the unmanaged product.

PeSIT/

Optionally, specify mutual authentication for file transfers. Specify a certificate if mutual authentication is selected.

Port in

The port the server listens for connections.

Certificate

Click **Browse** to select a public-key certificate file in the format DER or PEM or a public certificate chain file in the format P7B (PKCS#7). A certificate is required when mutual authentication is enabled.

When a certificate is selected, click **Display** to show certificate details.

Details

Groups

One or more groups separated by commas. See *Groups* on page 218.

Tags

One or more tags separated by commas.

Description

Description of the unmanaged product.

Contact

Optionally, the name, job title, email address and phone number of a contact person for the unmanaged product.

General concepts about flows 19

You can view and manage the flows defined in Central Governance on the Flow List page. You can filter the list of flows and perform other tasks.

A flow in Central Governance is the complete definition of the file transfers between the source and target.

See the following topics for information about use of Axway products in flows.

- *Transfer CFT flow concepts* on page 256
- *SecureTransport flow concepts* on page 250

Composition, deployment, execution

A flow is an exchange of business data between business applications or partners. You create flows in Central Governance and deploy them. Flows are triggered at the system level.

Flow composition

A flow is comprised of the following elements:

- A name.
- Optionally, tags and a description.
- Optionally, contact information about the business owner of the flow.
- Information about the source that owns the data being transferred. You can select as the source applications, application groups, partners, or unmanaged products. When the source is a business application, it must be associated with a registered product. In the Transfer CFT case, the flow definition includes transfer and file properties, processing rules and integration information.
- Information about the target that receives the data being transferred. You can select as the target applications, application groups, partners, or unmanaged products. When the target is a business application, it must be associated with a product. In the Transfer CFT case, the flow definition includes transfer and file properties, processing rules and integration information.
- Information about relays, or hubs, if they are part of the flow. A relay can be a registered product or an unmanaged product.
- The protocols used for the exchange between each segment: source-relay, relay-relay, relay-target.

Flow deployment and execution

Typically, a flow runs automatically, but can be run manually by executing a command on the products in the flow. For example, running the SEND command on Transfer CFT.

In the case of Transfer CFT, the flow definition:

- Might override some Transfer CFT configuration values, such as default processing scripts, during flow execution.
- Uses some network configurations that must be set in the Transfer CFT configuration. For example, all Transfer CFT have the following settings in the network system configuration: Interface=Any, PeSIT/No encryption.

Deploying a flow does not change the configurations of the products involved in the flow.

Direction in flows

Direction indicates the initiator of a file transfer in a flow, which can have multiple way points where direction changes. Direction is set in the protocol, and can be set differently in multiple places in flows with multiple protocols.

In the simplest flow, there is only the sender and receiver and the protocol for the transfer. The initiator of the transfer is configured in the protocol as:

- **Sender pushes file**. The sender is the initiator of the transfer request. The sender is the client and the receiver acts as the server.

  or

- **Receiver pulls file**. The receiver is the initiator of the transfer request. The receiver is the client and the sender acts as the server. The receiver's request triggers the sender to send the file.

A more complex flow might specify one or more relays between the sender and receiver. For example:

Source > Relay 1 > Relay 2 > Target

Direction in this flow is set in three places, as indicated in the following examples:

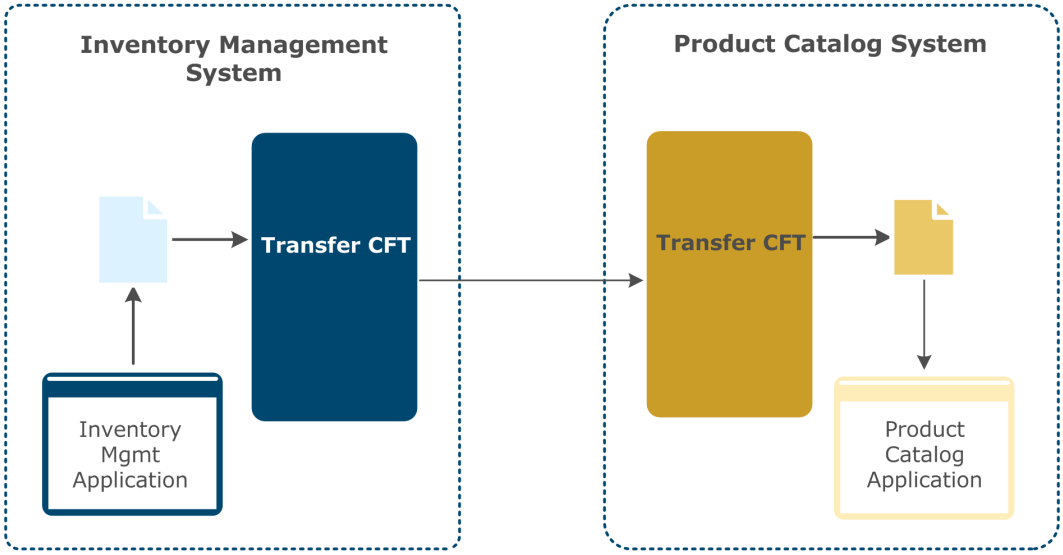
- Protocol between source and relay 1. Direction = Sender pushes file.
- Protocol between relay 1 and relay 2. Direction = Sender pushes file.
- Protocol between relay 2 and target. Direction = Receiver pulls file.

Direction = Sender pushes file

Direction = Sender pushes file describes a flow where a sender sends one or more files to a receiver.

Business scenario

The inventory management application receives reports from all warehouses about stock levels. It generates a file daily containing consolidated information on product availability. The inventory management application sends the file to the product catalog application.



Flow definition

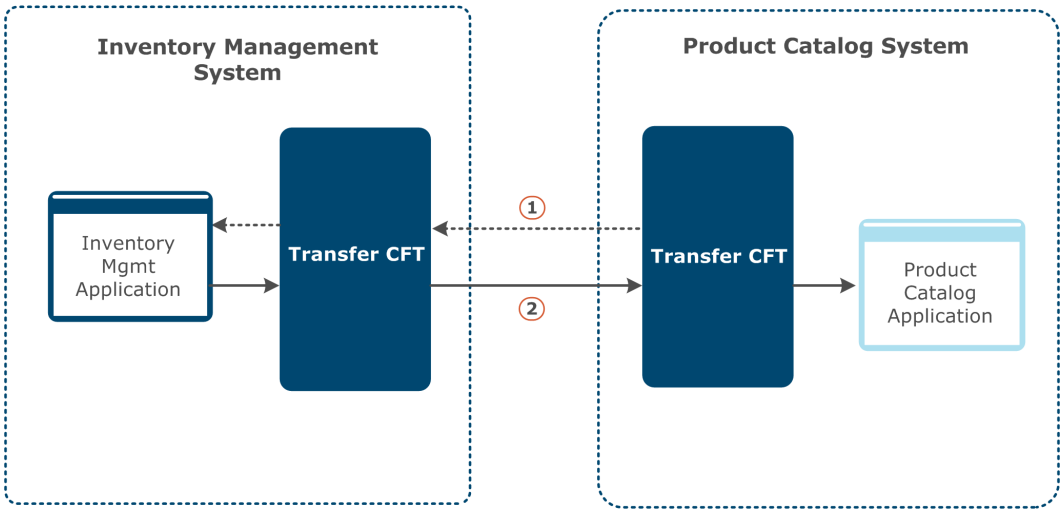
1. Add source and target applications. The source is the inventory management application. The target is one or more warehouse applications.
2. Optionally, edit the flow definition properties for the source and target.
3. Click **Protocol** between the source and target. Use **Sender pushes file** as the direction and set **PeSIT** as the protocol.
4. Save and deploy the flow.

Direction = Receiver pulls file

Direction = Receiver pulls file describes a flow where the receiver requests one or more files from a sender.

Business scenario

The inventory management application receives reports from all warehouses about stock levels. It generates a file daily containing consolidated information on product availability. The product catalog application requests the file, which the inventory management application sends in response.



Flow definition

1. Add source and target applications. The source is one or more warehouse applications. The target is the inventory management application.
2. Optionally, edit the flow definition properties for the source and target.
3. Click **Protocol** between the source and target. Set **Receiver pulls file** as the direction and **PeSIT** as the protocol.
4. Save and deploy the flow.

Flow lifecycle

Statuses are displayed on the Flows and flow detail pages during the phases of a flow's lifecycle.

The user interface uses two date formats for status messages:

- When today: *<Status> at <time> today*
- When a past day: *<Status> on <date>*

Phases

The following describes each phase in a flow's lifecycle. All of the following examples use the today message format.

Add flow

Add a flow. If the flow is not complete, the status is *Saved at <time> today*. If the flow is defined fully , the status is *Saved at <time> today, not deployed*. The flow definition is saved in Central Governance.

Deploy flow

Deploying a flow pushes the definition saved in Central Governance to products involved in the flow. The following statuses are possible during deployment.

Deploying since <time> today

Displayed while the deployment is in progress. Next to the status are details on the deployment status on all products:

- Number of products where the flow is being deployed
- Number of products where the flow is deployed successfully
- Number of products where the flow failed to deploy

You can click a link to access details of the deployment.

Deployed at <time> today

Displayed after the flow is deployed to all systems. Next to the status are details on the deployment status on all products:

- Number of products where the flow is deployed successfully
- Number of products where the flow failed to deploy

You can click a link to access details of the deployment.

Edit flow

Edit a flow that was deployed previously and save it. The status is *Saved not deployed*. Note that if you only change the details or contact of the general information section, the status does not change because this information is not pushed to products. For example, if you edit the flow description, the status does not change. Depending on the type of change, the flow might be deployed to one or all of the products involved.

Also, if the flow was deployed previously and then edited, the last deployment status is displayed in the status of the flow page. For example, *Saved at <time>, not deployed (Last deployment: 2 deployed)*. You can click the link for details of the previous flow deployment.

During modification, a flow can become incomplete for deployment. For example, all sources are removed or a protocol is not yet defined or all targets are removed. In these cases, when saving the flow the status becomes *Saved at <time> today*.

Remove flow

Removing a flow removes it from Central Governance. If the flow is deployed on a product, removing it also removes the flow definition from the product.

Likewise, removing a product from a flow also removes it from the flow definition when the flow is deployed.

The following statuses are possible while removing a flow.

Removing since <time> today

Displayed while the removal is in progress.

Removed at <time> today

Displayed after the remove flow process stopped but failed to remove on one or more products. Once the problem is corrected, you can attempt to remove flow again, but the flow can no longer be edited. Next to the status are details on the remove status on all products:

- Number of products where the flow is removed successfully
- Number of products where the flow failed to remove

You can click a link to access details of the deployment. No dedicated status is available once the flow is fully removed.

Business scenario

The following table illustrates the flow phases and the status changes that occur in a store-to-corporate business use case. Multiple stores in different regions must exchange files with the product catalog application at corporate headquarters. All of the examples use the today message format.

Business need	Flow status before	User action	Flow status after
All stores must be able to exchange files with the corporate application in a common way		Create flow	Saved at <time> today, not deployed
Save time and money by deploying the flow to all stores at one time	Saved at <time> today, not deployed	Deploy flow	Deployed at <time> today
Due to expansion into new region, add stores exchanging with the corporate application	Deployed at <time> today	Edit flow to add new stores	Saved not deployed

Business need	Flow status before	User action	Flow status after
Save time and money by deploying the flow to all new stores at one time	Saved not deployed	Deploy flow	Deployed at <time> today
Close some stores that are not performing	Deployed at <time> today	Remove stores from groups	Deployed at <time> today
All stores must be able to exchange files with the corporate application in a common way		Create flow; all store groups not defined	Saved at <time> today
Ensure all stores assigned to regional groups for ease of management	Saved at <time> today	Edit flow; flow definition complete	Saved at <time> today, not deployed
Save time and money by deploying the flow to all stores at one time	Saved at <time> today, not deployed	Deploy flow	Deployed at <time> today
Due to expansion into new region, add a new application group for the new stores exchanging with the corporate application	Deployed at <time> today	Edit flow to add new application group to the source	Saved not deployed
Save time and money by deploying the flow to all new stores at one time	Saved not deployed	Deploy flow	Deployed at <time> today
Close some stores that are not performing	Deployed at <time> today	Remove stores from application groups; flow is not modified	Deployed at <time> today

Communication profiles

A communication profile contains the technical details for making connections between clients and servers to transfer data. The two types of profiles are based on the roles of senders and receivers in file transfers.

- A **server communication profile** contains details for a client to transfer data via a protocol to the sender or receiver that acts as a server.

    - The sender acts as a server when it publishes files for the receiver.
    - The receiver acts as a server when it receives files pushed by the sender.
- A **client communication profile** contains details for the sender or receiver to connect via a protocol to the server.
    - The sender acts as client when it pushes files to the receiver.
    - The receiver acts as a client when it pulls files from the sender.

PeSIT, which supports acknowledgments, also must have communication profiles for acknowledgments to enable sending receipts or ACKs when requested.

Communication profiles are used in flow segments between source and target, source and relay, relay and relay, and relay and target. The user interface prompts when communication profiles are required.

Registered products, unmanaged products and partners own server communication profiles. The following describes how to view the communication profiles of these objects.

Products

Click **Products** on the top toolbar, click the name of a product and click **Configuration**. For SecureTransport the profiles are listed under Server Communication Profiles. For Transfer CFT the profiles are listed under Protocols.

Unmanaged products

Select **Products > Unmanaged Products** and click the name of an unmanaged product. The profiles are listed under Protocol.

Partners

Select **Partners** on the top toolbar and click the name of a partner. The profiles are listed under Server Communication Profiles.

Client communication profiles only are designated in flow definitions. When a flow is removed, any client communication profiles within it also are removed.

Applications do not own communication profiles. They inherit the profiles of their associated products.

Relays in a flow

A relay is a product used to route a file from a source to a target or another relay. A relay can push the files it receives to the next application or product in the flow, which is the target or another relay. It also can pull files from the source or another relay in the flow.

You can define multiple relays in a flow. For example, a bank's branches in the United States must send files to its branches in France. The files are transferred through a central relay in the United States to a central relay in France, which sends the files to the branches.

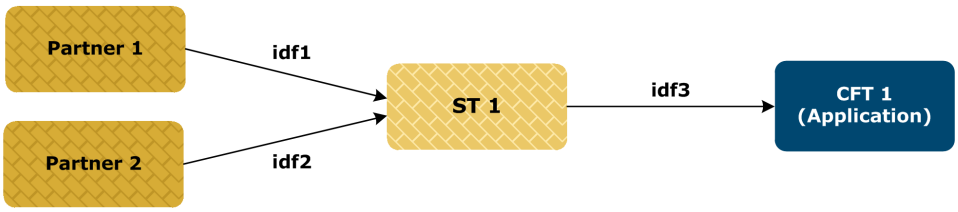
The following are reasons for using relays.

- Reduce network complexity and cost. If multiple groups of stores are sending files to a corporate system, all files can pass through the relay to corporate using a single network connection. Also, if there are multiple regional networks, you can route traffic through a regional hub system that acts as a relay and sends files on to another regional hub, which then distributes files to one or more targets.
- Simplify tracking of flows and simplify identifying and resolving problems. Using Central Governance monitoring, you can track a flow from source to target through one or more relays.

For more information see *Transfer CFT as relay* on page 256 and *SecureTransport as relay in flows* on page 251.

Flow identifiers

Flow identifiers are defined at each step in a flow when files are transferred via PeSIT, as the following illustrates.

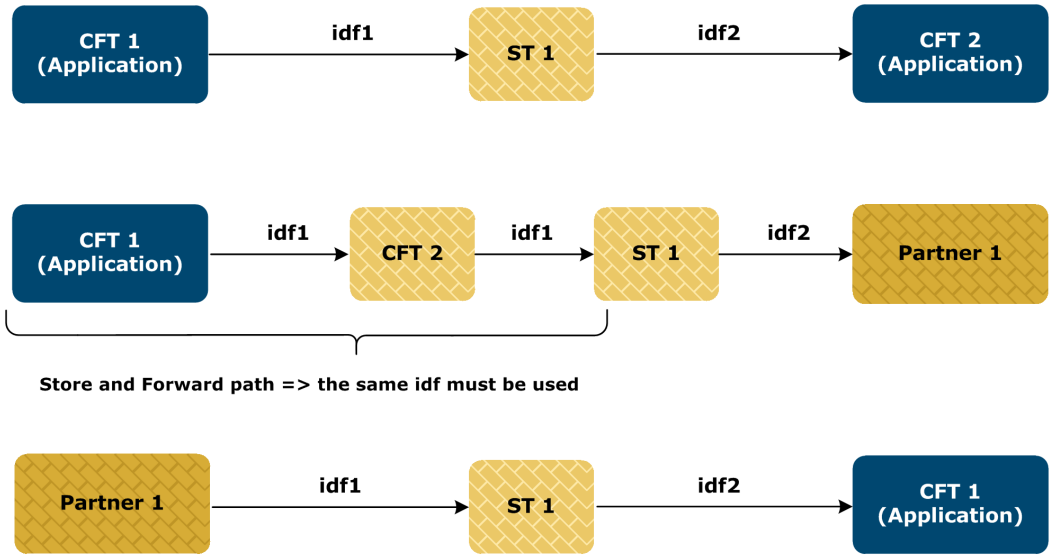


In the graphic, a partner is an external partner, ST represents an instance of SecureTransport and CFT represents an instance of Transfer CFT.

When deploying to Transfer CFT, flow identifiers are used to set the name of the send template and receive template. This identifier is used when you trigger a transfer on PeSIT: **idf parameter**. A flow identifier is called an IDF in Transfer CFT.

The flow identifier is defined at each step of the flow only for PeSIT for each sender-receiver pair. A flow identifier is unique across all flows, but the same flow identifier can be used multiple times in the same flow. In store-and-forward cases, the identifier must be the same all along the store-and-forward path.

The following illustrates how identifiers can be set in different flows.



Flow patterns

You can have many types of flow patterns for internal file transfers between senders and receivers. Most of the patterns described in the following topics are for flows where direction is set as sender pushes file. However, you can implement these as receiver pulls file.

Internal flows describe file transfers between applications.

Relays are Axway products or unmanaged products.

The following outlines the types of flow patterns.

1. Internal flows
    a. Point to point (no relay)
        - One to one
        - One to many
        - Many to one
    b. Via relays
        - One to one
        - One to many
        - Many to one
2. External flows

In addition, external flows can have one-to-one, one-to-many or many-to-many patterns.

a. Inbound

b. Outbound

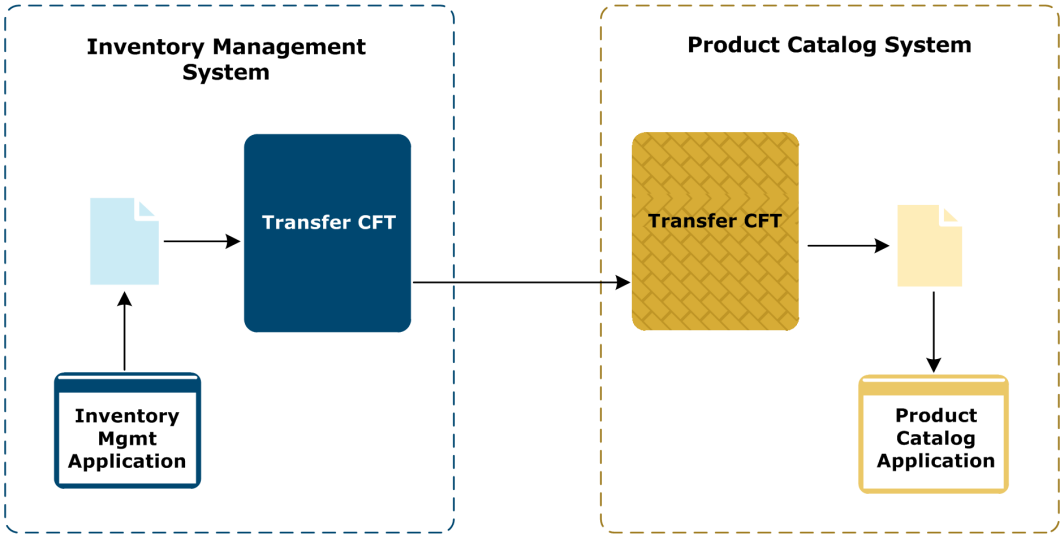
c. Partner to partner

Internal flow: One to one

The one-to-one pattern describes a flow where a single source application sends one or more files to a single target application.

Business scenario 1

The inventory management application receives reports from all warehouses about stock levels. It generates a file daily containing consolidated information on product availability. It sends the file to the product catalog application.

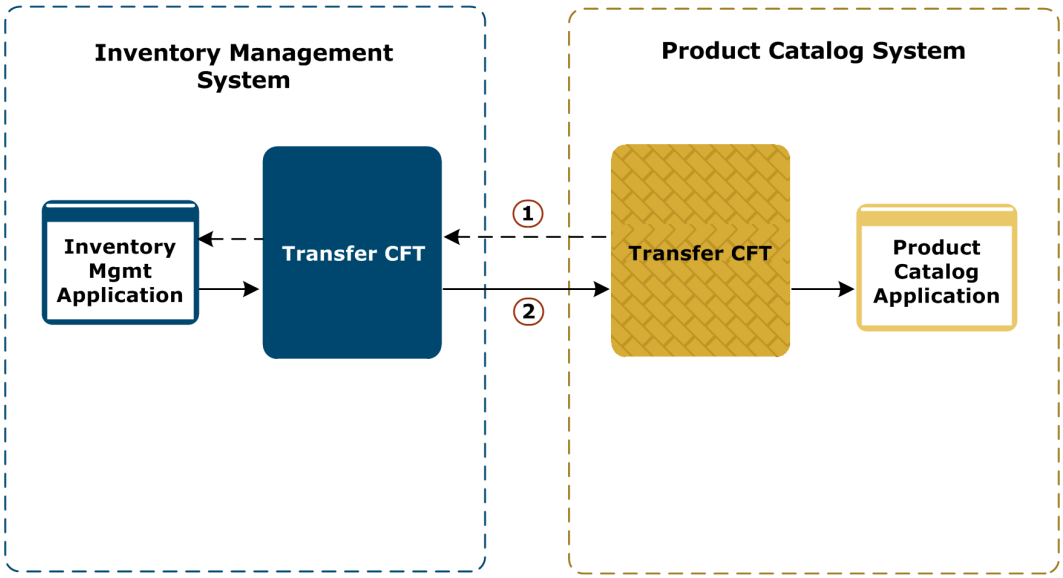


Flow definition

1. Add the application inventory management system as the source and the application product catalog system as the target.
2. Optionally, edit the flow definition properties for the source and target.
3. Edit the protocol between the source and target and set **PeSIT** as the exchange protocol.
4. Save and deploy the flow.

Business scenario 2

The inventory management application receives reports daily from all warehouses about stock levels. It generates a file daily containing consolidated information on product availability. The product catalog application requests the file, which the inventory management application sends in response.



Flow definition

1. Add the application product catalog system as the source and the application inventory management system as the target.
2. Optionally, edit the flow definition properties for the source and target.
3. Edit the protocol between source and target and set **Receiver pulls file** as the direction and **PeSIT** as the exchange protocol.
4. Save and deploy the flow.

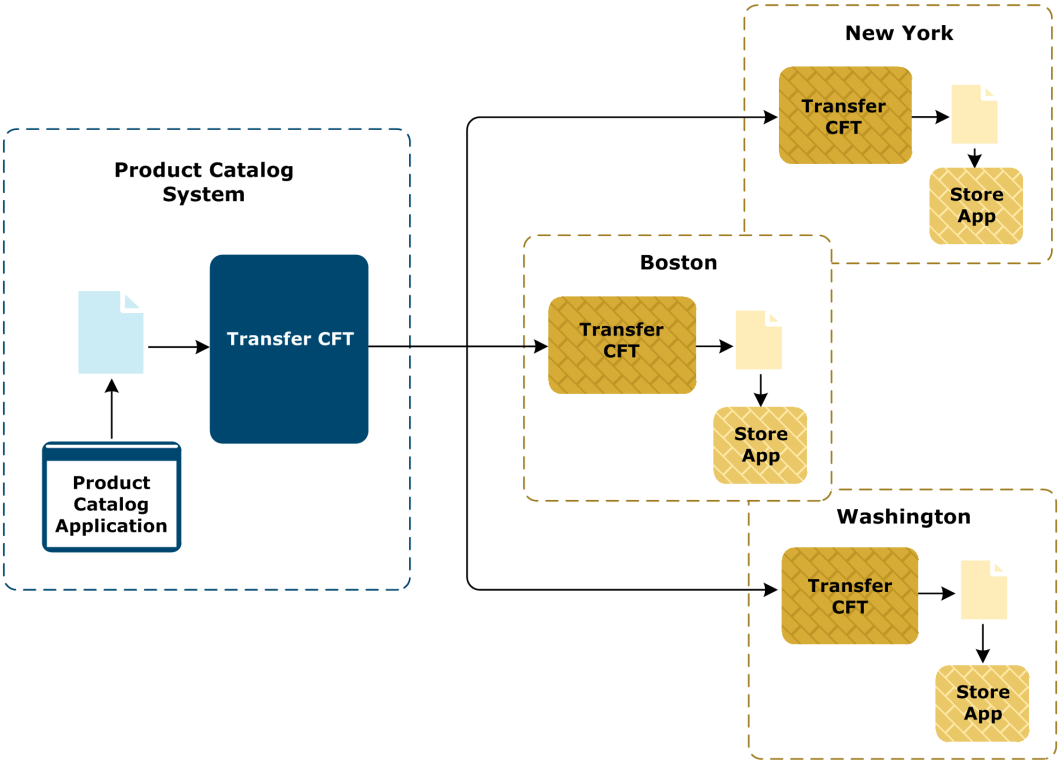
Internal flow: One to many

The one-to-many pattern describes a flow where a single source application sends a file to multiple target applications.

Business scenario

The product catalog application generates a file daily describing products and their prices. It sends the file to all stores nightly so stores have the updated information before opening in the morning.

The product catalog application uses a broadcast list to send the file to all stores at once. See *Transfer CFT broadcast and collect* on page 268.



Flow definition

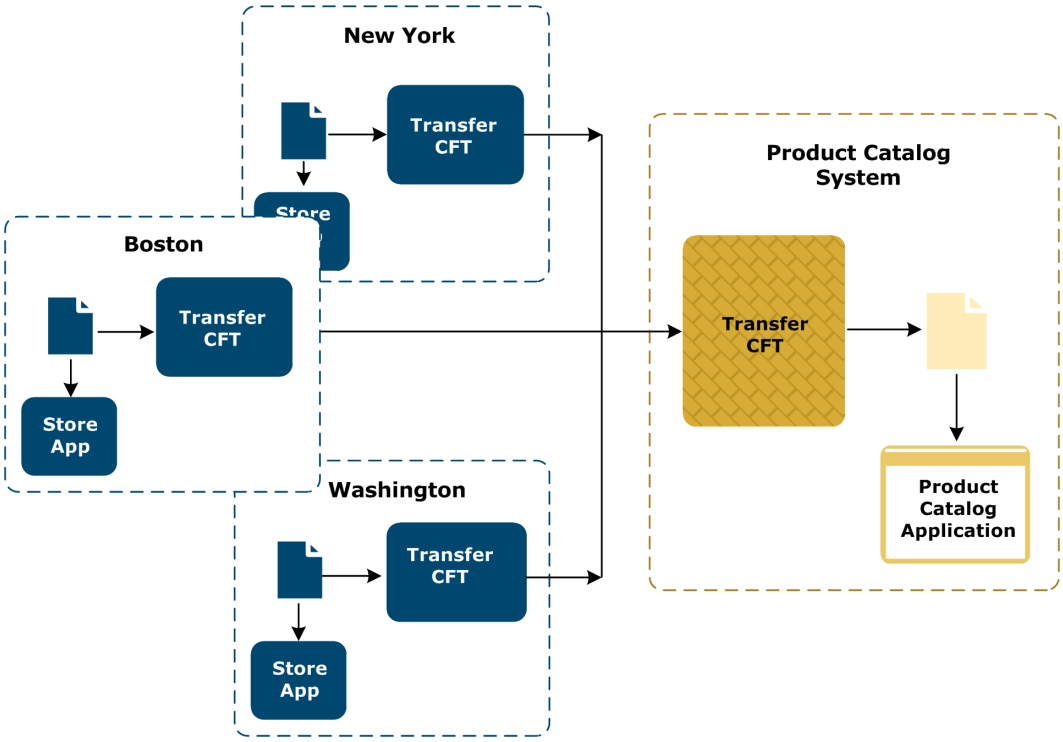
1. Add the product catalog system application as the source.
2. Add applications corresponding to all stores as the targets. Optionally, you can add the store applications as a group. See *Application groups* on page 215.
3. Edit the protocol between source and target and set **PeSIT** as the exchange protocol.
4. To use a broadcast list, edit source transfer properties and enable the broadcast list. Specify the name of the list and the action to take if a target is unknown at the time of the transfer.
5. Optionally, edit source file properties.
6. Optionally, edit source processing scripts to specify how the scripts are applied to the broadcast list.
7. Optionally, edit target transfer properties and processing scripts.
8. Save and deploy the flow.

Internal flow: Many to one

The many-to-one pattern describes a flow where multiple source applications send a file to a single target application.

Business scenario

The store applications generate and send a daily sales report to the product catalog application at the end of each business day. The product catalog application might use the data to generate and send files to the inventory management application. That application in turn might notify logistics and warehouse applications whether store inventory levels are low and need restocking.



Flow definition

1. Add applications corresponding to all stores as the sources. Optionally, you can add the store applications as a group. See *Application groups* on page 215.
2. Add the product catalog system application as the target.
3. Edit the protocol between source and target and set **PeSIT** as the exchange protocol.
4. Optionally, edit the flow definition properties for the source.
5. Optionally, edit target file properties and processing scripts.
6. Save and deploy the flow.

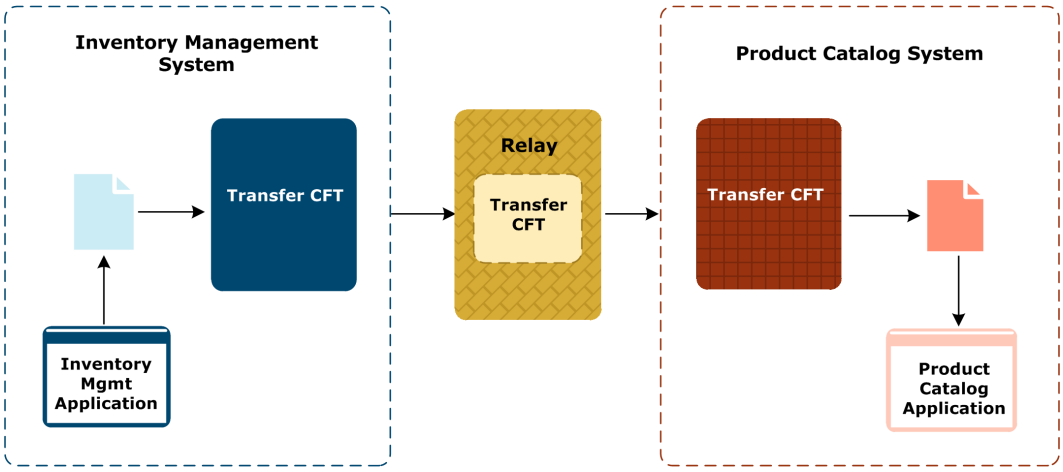
Internal flow: One to one via relay

The one-to-one via relay pattern describes a flow where a single source application sends one or more files to a single target through a relay.

Only the use of a relay makes this pattern different than the one-to-one pattern.

Business scenario

The inventory management application receives reports from all warehouses about stock levels. It generates a file daily containing consolidated information on product availability. It sends the file to the product catalog application through a relay. The relay is Transfer CFT.



Flow definition

1. Add the application inventory management system as the source and the application product catalog system as the target.
2. Optionally, edit the flow definition properties for the source and target.
3. Add a relay.
4. Edit the protocol between the source and relay and set **PeSIT** as the exchange protocol.
5. Edit the protocol between relay and target and set **PeSIT** as the exchange protocol.
6. Save and deploy the flow.

As Transfer CFT is the relay in this example, you cannot change the protocol directions between source-relay and relay-target. When used as a relay, Transfer CFT uses store and forward for transfers. See *Direction in flows* on page 231.

Internal flow: One to many via relay

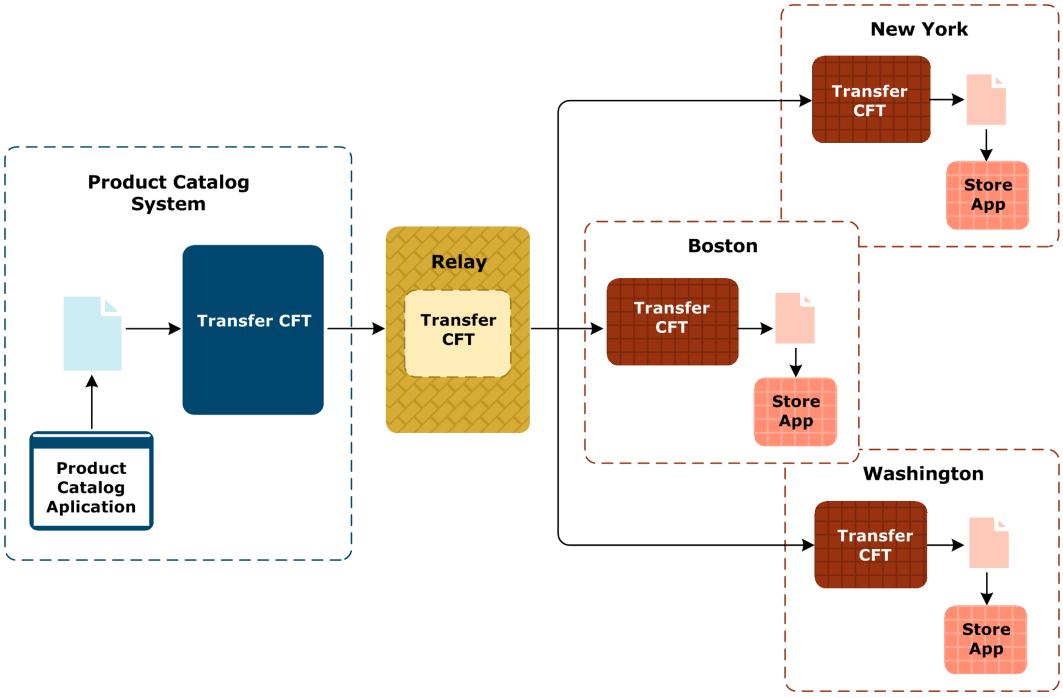
The one-to-many via relay pattern describes a flow where a single source application sends a file to multiple targets through a relay.

Only the use of a relay makes this pattern different than the one-to-many pattern.

Business scenario

The product catalog application generates a file daily describing products and their prices. It sends the file to each store nightly so the stores have the updated information before opening in the morning. The relay is Transfer CFT.

The product catalog application uses a broadcast list to send the file to all stores at once. See *Transfer CFT broadcast and collect* on page 268.



Flow definition

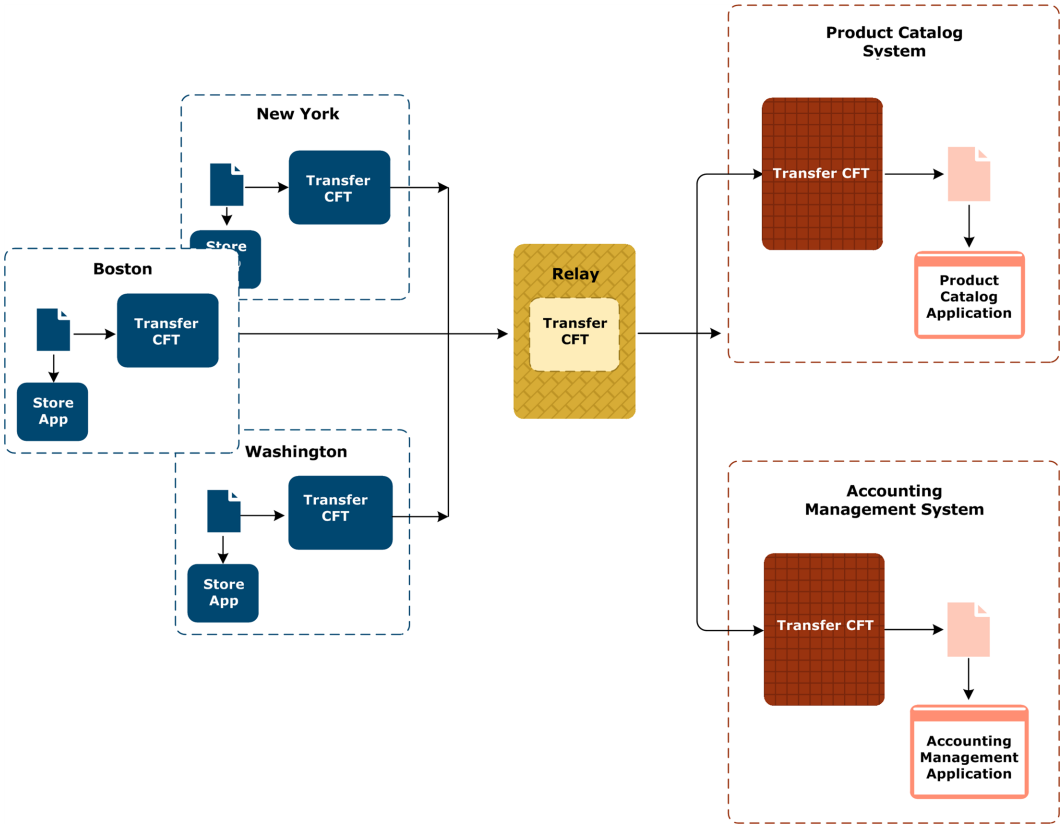
1. Add the product catalog system application as the source.
2. Add applications corresponding to all stores as the targets. Optionally, you can add the store applications as a group. See *Application groups* on page 215.
3. Add a relay.
4. Edit the protocol between the source and relay and set **PeSIT** as the exchange protocol.
5. Edit the protocol between relay and target and set **PeSIT** as the exchange protocol.
6. To use a broadcast list, edit source transfer properties and enable the broadcast list. Specify the name of the list and the action to take if a target is unknown at the time of the transfer.
7. Optionally, edit source file properties.
8. Optionally, edit source processing scripts to specify how the scripts are applied to the broadcast list.
9. Optionally, edit target transfer properties and processing scripts.
10. Save and deploy the flow.

Internal flow: Many to many via relay

The many-to-many via relay pattern describes a flow where multiple source applications send a file to multiple target applications through a relay.

Business scenario

The store applications generate a daily sales report at the end of each business day. They send the files nightly to the product catalog and accounting management applications. The relay is Transfer CFT.



Flow definition

1. Add applications corresponding to all stores as the sources. Optionally, you can add the store applications as a group. See *Application groups* on page 215.
2. Add the product catalog and accounting management applications as the targets. Optionally, you can add both applications as a group. See *Application groups* on page 215.
3. Optionally, edit the flow definition properties for the sources and targets.
4. Add the relay.
5. Edit the protocol between the source and relay and set **PeSIT** as the exchange protocol.

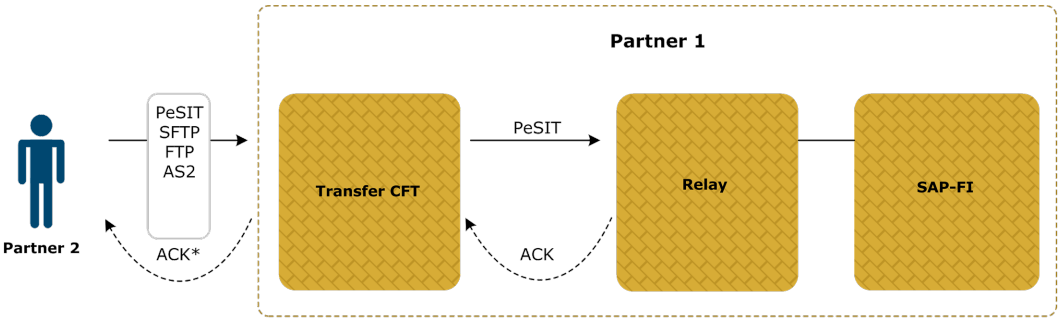
6. Edit the protocol between relay and target and set **PeSIT** as the exchange protocol.
7. Save and deploy the flow.

External flow: Inbound

The inbound pattern describes a flow where a partner uploads a file and the file is routed to an application.

Business scenario

Partner 1's SAP FI application must receive payments from partner 2.



*ACK only for PeSIT and AS2

Flow definition

1. Add partner 2 as the source and the SAP FI application as the target.
2. Optionally, edit the flow definition properties for the source and target.
3. Add a relay. This can be SecureTransport.
4. Edit the protocol between source and relay, setting any protocol as the exchange protocol.
5. Edit the protocol between the relay and target, setting **PeSIT** as the exchange protocol.
6. Save and deploy the flow.

In this example you can change the directions between source-relay and relay-target. One or both of the following are possible:

- The relay can request files from partner 2.
- The SAP FI application can request files from the relay.

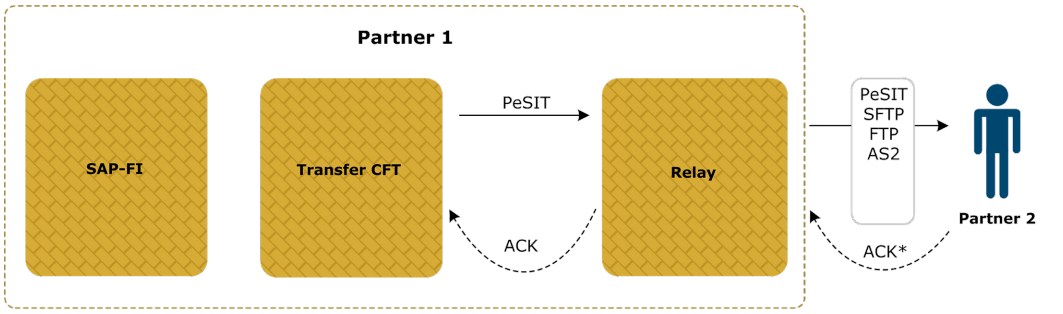
The flow can have multiple partners as sources. For example, two partners could send data to partner 1.

External flow: Outbound

The inbound pattern describes a flow where an application uploads a file that is routed to a partner

Business scenario

Partner 1 must send an invoice to partner 2. Partner 1's SAP-FI application uploads the invoice to SecureTransport and the file is routed to partner 2.



*ACK only for PeSIT and AS2

Flow definition

1. Add the SAP FI application as the source and partner 2 as the target.
2. Optionally, edit the flow definition properties for the source and target.
3. Add a relay. This can be SecureTransport.
4. Edit the protocol between source and relay and set **PeSIT** as the exchange protocol.
5. Edit the protocol between relay and target and set any protocol as the exchange protocol.
6. Save and deploy the flow.

In this example you can change the directions between source-relay and relay-target. One or both of the following are possible:

- The relay can request files from the SAP FI application.
- Partner 2 can request files from the relay.

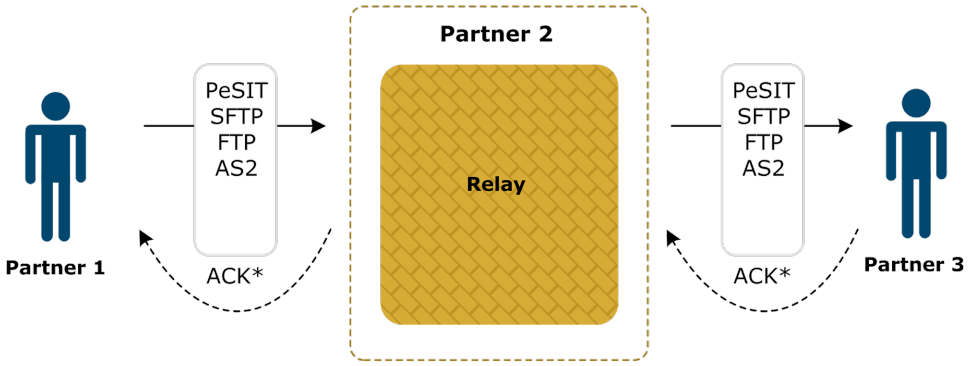
The flow can have multiple partners as targets. For example, two partners could receive data from partner 1.

External flow: Partner to partner

The partner-to-partner pattern describes a flow where a partner uploads a file that is routed to another partner.

Business scenario

Partner 2 receives orders from partner 1 that must be sent to partner 3. Partner 1 uploads the orders to partner 2's SecureTransport and the file is routed to partner 3.



*ACK only for PeSIT and AS2

Flow definition

1. Add partner 1 as the source and partner 3 as the target.
2. Optionally, edit the flow definition properties for the source and target.
3. Add a relay. This can be SecureTransport.
4. Edit the protocol between source and relay and set any protocol as the exchange protocol.
5. Edit the protocol between relay and target and set any protocol as the exchange protocol.
6. Save and deploy the flow.

In this example you can change the directions between source-relay and relay-target. One or both of the following are possible:

- The relay can request files from partner 1.
- Partner 3 can request files from the relay.

The flow can have multiple partners as sources and targets.

SecureTransport flow concepts 20

The following topics describe using SecureTransport in flows.

SecureTransport as source in flows

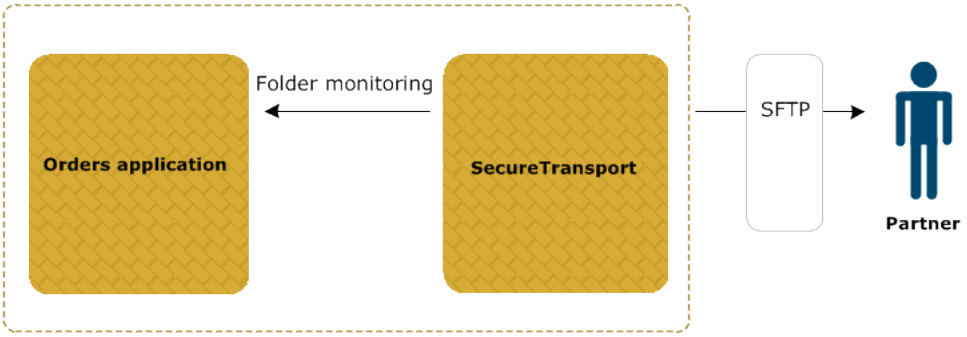
When used as the source in a flow, SecureTransport picks up files made available by an internal application and sends them, directly or via relays, to other applications or external partners. The internal application moves files into a directory local to SecureTransport. SecureTransport monitors the application directory according to an implicit or explicit scheduler, selects files matching a filter and sends the files.

You define receive, file processing and send properties in flows when SecureTransport is the sender.

- Receive properties contain the folder monitoring properties for how SecureTransport scans the application directory.
- File processing properties are configured the same as when SecureTransport is a relay.
- Send properties depend on the exchange protocol and direction between SecureTransport and the next participant to the flow.

Business scenario

This scenario describes a one-to-one flow pattern where SecureTransport is the source. SecureTransport has folder monitoring enabled to retrieve files from an internal orders application. After receiving the file, SecureTransport pushes it to the external partner, which is the flow target.



Flow definition

1. Add the orders application and associate it with SecureTransport.
2. Add a flow, using SecureTransport as the source and the partner as the target.
3. Configure SFTP as the protocol between the source and target.
4. Edit the flow properties for the source.
5. Save and deploy the flow

SecureTransport as relay in flows

SecureTransport can be used as relays in external and internal flows, but use in external flows is more common.

Overview

In external flows:

- Partners can send files to applications via SecureTransport relays.
- Applications can send files via SecureTransport relays to partners.
- Atypically, partners can send files via SecureTransport relays to other partners.

In internal flows, applications can send files via SecureTransport relays to other applications.

As a relay, SecureTransport can receive files directly from the sender or initiate the request by pulling files from the sender. SecureTransport can push files to a receiver or it can publish received files that a receiver pulls.

Flow properties you define for SecureTransport as a relay are receive, file processing and send properties.

Receive properties depend on the protocol and direction definition configured before the SecureTransport relay. Receive properties also depend on the protocol and direction definition configured after the SecureTransport relay. If you change the protocol or direction in the flow, different properties are displayed in the user interface. For example:

- When SecureTransport pulls files from a sender, you can enable a scheduler to trigger the pull. (The scheduler is not available when the sender pushes files to SecureTransport.)
- When SecureTransport pushes files to a receiver via SFTP, you must define the remote directory of the receiver. (Over PeSIT the remote directory is not specified in SecureTransport.)

Flows defined in Central Governance and deployed on SecureTransport use advanced routing for routing files from senders to receivers.

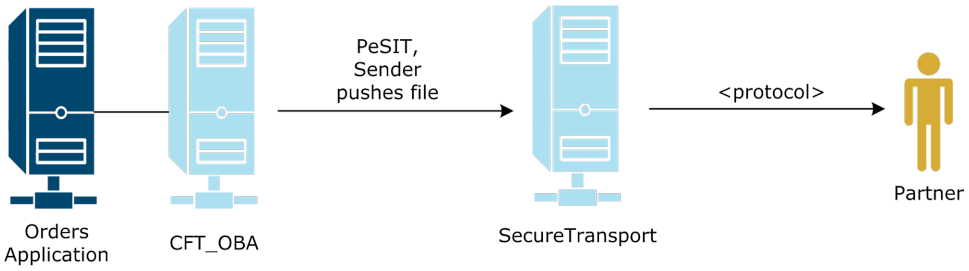
When used as a relay in PeSIT flows, SecureTransport does not use store-and-forward explicitly for transfers. The sender sends files to SecureTransport, which routes the files to receivers. If store-and-forward was explicit, the sender would send files to the receiver directly.

Relay examples

The following scenarios describe one-to-one flow patterns where SecureTransport is used as a relay.

Business scenario 1

An internal application sends orders to an external partner via a relay. The source pushes files to SecureTransport, which sends to the partner.

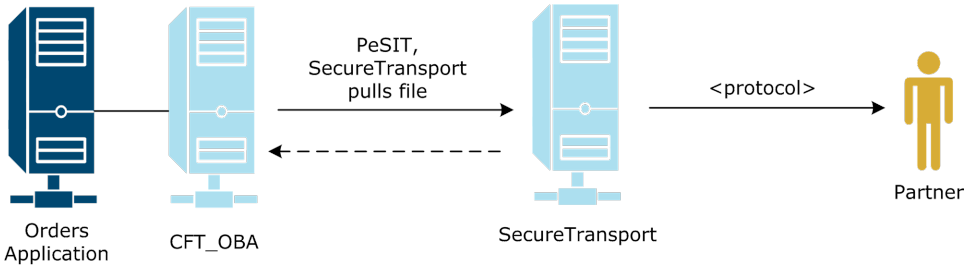


Flow definition

1. Add the orders application as the source and the partner as the target.
2. Optionally, edit the flow definition properties for the source.
3. Add SecureTransport as the relay.
4. Edit the protocol between the source and relay and set PeSIT as the exchange protocol.
5. Edit the protocol between relay and target.
6. Edit the flow definition properties for the relay.
7. Save and deploy the flow.

Business scenario 2

An internal application sends orders to an external partner via a relay. SecureTransport pulls files from the source and then sends to the partner.

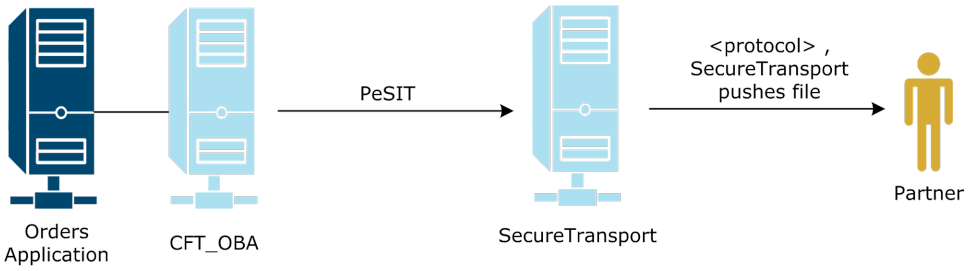


Flow definition

1. Add the orders application as the source and the partner as the target.
2. Optionally, edit the flow definition properties for the source.
3. Add SecureTransport as the relay.
4. Edit the protocol between the source and relay and set PeSIT as the exchange protocol and the direction as receiver pulls file.
5. Edit the protocol between relay and target.
6. Edit the flow definition properties for the relay.
7. Save and deploy the flow.

Business scenario 3

An internal application sends orders to an external partner via a relay. The source pushes files to SecureTransport, which pushes to the partner.

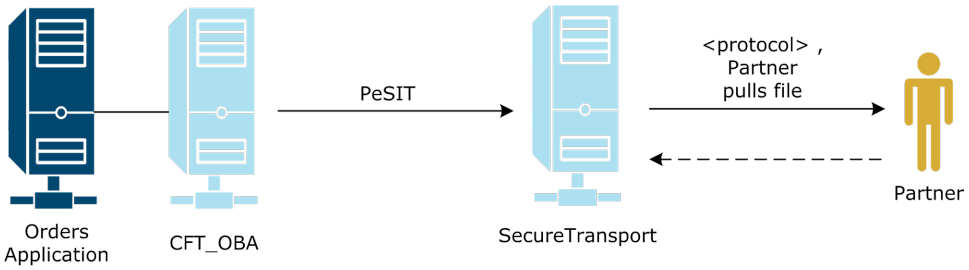


Flow definition

1. Add the orders application as the source and the partner as the target.
2. Optionally, edit the flow definition properties for the source.
3. Add SecureTransport as the relay.
4. Edit the protocol between the source and relay and set PeSIT as the protocol.
5. Edit the protocol between relay and target and set the exchange protocol.
6. Edit the flow definition properties for the relay.
7. Save and deploy the flow.

Business scenario 4

An internal application sends orders to an external partner via a relay. The source pushes files to SecureTransport, and the partner pulls the files from SecureTransport.

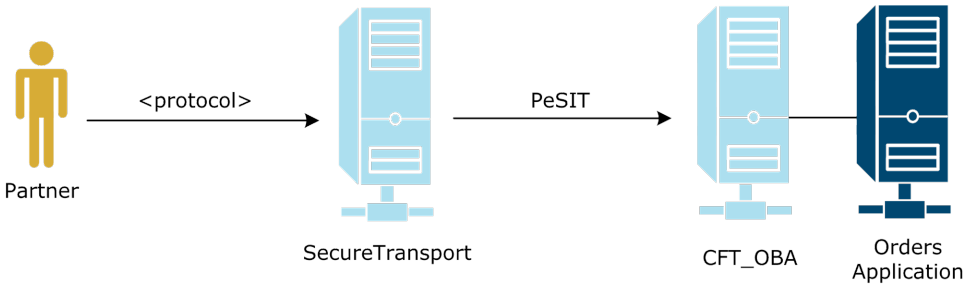


Flow definition

1. Add the orders application as the source and the partner as the target.
2. Optionally, edit the flow definition properties for the source.
3. Add SecureTransport as the relay.
4. Edit the protocol between the source and relay and set PeSIT as the protocol.
5. Edit the protocol between relay and target and set the exchange protocol and the direction as receiver pulls file.
6. Edit the flow definition properties for the relay.
7. Save and deploy the flow.

Business scenario 5

A partner sends invoices to an internal application via a relay. The partner pushes files to SecureTransport, which sends the files to the application.



Flow definition

1. Add the partner as the source and the orders application as the target.
2. Optionally, edit the flow definition properties for the target.
3. Add SecureTransport as the relay.
4. Edit the protocol between the source and relay and set the exchange protocol.
5. Edit the protocol between relay and target.
6. Edit the flow definition properties for the relay.
7. Save and deploy the flow.

As SecureTransport is the relay in this example, you can change the protocol directions between source-relay and relay-target.

SecureTransport as target in flows

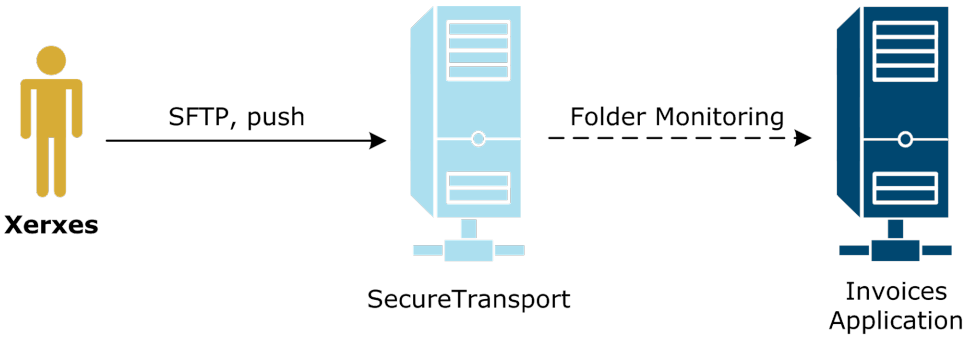
When used within an application as the target in a flow, SecureTransport pushes the files received, directly from partners or other relays, to a directory where an internal application has access. SecureTransport can create the directory or rename received files.

You define receive, file processing and send properties in flows where SecureTransport is the receiver.

- Receive properties depend on the exchange protocol and direction between SecureTransport and the preceding participant in the flow.
- File processing properties are configured the same as when SecureTransport is a relay.
- Send properties have information about the directory where files are pushed and settings on possible post-sending actions.

Business scenario

The following scenario describes a one-to-one flow pattern where SecureTransport is the target. SecureTransport receives from a partner an invoice that must be moved to a folder the invoice application can access.



Flow definition

1. Add the invoice application and associate it with SecureTransport.
2. Add a flow, using the partner as the source and the application invoice application as the target.
3. Configure SFTP as the protocol between the source and target.
4. Edit the flow properties for the target.
5. Save and deploy the flow.

Transfer CFT flow concepts 21

The following topics describe using Transfer CFTs in flows. Central Governance uses the Copilot WebService to deploy flows to Transfer CFTs.

Transfer CFT as relay

When Transfer CFT is the relay in a flow and receives a transfer, it creates a temporary file that is deleted after the file is sent to the target or next relay.

A Transfer CFT relay only can receive files and push files to the next relay or target.

A relay transfers the file to the next relay or the target exactly as received from the source or previous relay. During the transfer, the relay does not perform post-processing, error processing or acknowledgment processing. Processing scripts can be executed on source and target systems. When the target receives the file, it can send an acknowledgment to the source.

Transfer CFT flow transfer modes

The following table lists the Transfer CFT flow transfer modes by direction. Transfer CFT can push files to next receiver in the flow or pull files from the sender.

Transfer mode	Direction
Send a file – closed mode on page 257	Sender pushes
Send a file – open mode on page 258	Sender pushes
Receive a file – explicit provisioning on page 260	Receiver pulls
Receive a file – implicit provisioning on page 261	Receiver pulls
Send a group of files – grouped on page 259	Sender pushes
Send a group of files – file-by-file on page 260	Sender pushes
Receive a group of files – grouped, explicit provisioning on page 262	Receiver pulls
Receive a group of files – file-by-file, implicit provisioning on page 262	Receiver pulls

Transferring groups of files

When defining a flow with Transfer CFT in Central Governance, you can specify a group of files to send in source file properties by selecting the Multiple option in the Source field.

Specify a set of files or a directory in the File name sent field:

- Use wildcard characters in the file name. The asterisk (*) is used for multiple characters and the question mark (?) is used for a single character.
    - /mydir/*.txt
    - /mydir/a?.*
- Specify a directory name.
    - mydir

Or, specify a file containing the list of files to be transferred in the File list field:

- myfile

Group transfer modes

There are two processing modes when sending groups of files:

- File-by-file: Files are transmitted individually. This is the default mode. You can change the mode in the transfer processing section of the Transfer CFT configuration.
- Grouped: Files are transmitted as a group when possible. On the source, a single file (archive) is created from the group of files. On the target, there is a directory where the archive (working file) is received. Files are automatically extracted from the working file into this directory.

The mode used depends on the operating system of the Transfer CFTs involved in the flow and the transfer processing mode configuration of each. When you use the grouped mode, the files are transferred as a group only when the source and target systems have the same operating system. If the operating systems are different, the transfer is sent file-by-file.

Transfer mode: Sender pushes files

The following topics describe source-initiated transfer modes in flows involving Transfer CFT.

Prerequisite

The source and target of the flow are business applications associated with Transfer CFTs. The direction is **Sender pushes file** in the definition of the protocol after the source.

Send a file - closed mode

The source defines the location of the file to send. The target defines the file to receive.

Source file properties

- Select the **Single** option.
- Optionally, in the **Filename** field , specify the full path name of the file to send. If you don't specify a value, it can be set at transfer execution time. If you do set a value, you can override it at transfer execution time.

Target file properties

Specify a value in the **Filename** field. The value must contain a path that exists on the target and the file name of the received file. The default value is `pub/&IDF.&IDTU.&FROOT.RCV`. This is a required field because the value cannot be set at transfer execution time.

Transfer execution command

On the Transfer CFT source, the command syntax to initiate the transfer is:

```
CFTUTIL send PART=<CFT_target>, IDF=<flow_ID>[, fname=<source_filename>]
```

Note The `fname` parameter is required if a value was not supplied in the **Path** field in the source file properties definition.

Send a file - open mode

The source defines the location of the file to send and the file name on the target.

Source file properties

- Select the **Single** option.
- In the **Filename** field, optionally specify the full path name of the file to send. If you don't specify a value, it can be set at transfer execution time.
- In the **File name sent** field, specify an existing file path on the target and the desired file name on the target.

Target file properties

Specify a value in the **Filename** field. The value must contain a path that exists on the target and the file name of the received file. Best practice is to set this value to **&NFNAME**.

Transfer execution command

On the Transfer CFT source, the command syntax to initiate the transfer is:

```
CFTUTIL send PART=<CFT_target>, IDF=<flow_ID>[, fname=<source_filename>,
nfname=<target_filename>]
```

In this command, `fname` represents the file to be sent and `nfname` matches the filename under which the file is sent to the target. For example:

```
CFTUTIL send PART=CFT10, IDF=WR1, fname=/home/DS1234.txt,
nfname=DS1234QR.txt
```

Note The `fname` parameter is required if a value was not supplied in the **Path** field and the `nfname` parameter is required if a value was not supplied in the **File name sent** field in the source file properties definition.

Send a group of files - grouped

The source defines the group of files to send. Source and target Transfer CFTs must have the same operating system. The source Transfer CFT must be configured to use this mode, which requires setting **Transmit files individually** to **When necessary** in the Transfer processing section of the configuration.

Source file properties

- Select the **Multiple** option.
- In the **Path** field, specify the group of files to send using wildcard characters or a directory name. You can instead specify a file list using the file list field. If you don't specify a value, it can be set at transfer execution time.
- In the **Archive name** field, specify a value. For example, &IDF.&IDTU.RCV.

Target file properties

- In the **Filename** field, specify the path for the received files. The default value is pub/&IDF.&IDTU.&FROOT.RCV. The path is created if it does not exist.
- In the **Temporary file** field, specify the name of the temporary archive. The default value is pub/&WFNAME.&IDTU.RCV. If you do not change the default value, pub/&IDF.&IDTU.&FROOT.RCV is created as the directory.

Transfer execution command

On the Transfer CFT source, the command syntax to initiate the transfer is:

```
CFTUTIL send IDF=<flow_ID>, PART=<CFT_target>[, fname=<#|@filename>,
wfname=<archive_name>]
```

Note The `fname` and `wfname` parameters are required if they were not set in the target file properties definition. The @ character must precede the file name on Linux and UNIX platforms, and the # character on Windows platforms.

Send a group of files - file-by-file

The source defines the group of files to send. Source and target Transfer CFTs have different operating systems, or **Transmit files individually** is set to **Always** in the Transfer processing section of the source system configuration.

Source file properties

- Select the **Multiple** option.
- In the **Path** field, specify the group of files to send using wildcard characters or a directory name. If you don't specify a value, it can be set at transfer execution time.

Target file properties

In the **Filename** field, specify the path for the received files. The default value is pub/&IDF.&IDTU.&FROOT.RCV. The path is created if it does not exist.

Transfer execution command

On the Transfer CFT source, the command syntax to initiate the transfer is:

```
CFTUTIL send IDF=<flow_ID>, PART=<CFT_target>[, fname=<#|@filename>]
```

Note The `fname` parameter is required if it was not set in the target file properties definition.

Transfer mode: Target pulls files

The following topics describe target-initiated transfer modes in flows involving Transfer CFT.

Prerequisite

The source and target of the flow are business applications associated with Transfer CFTs. The direction is **Receiver pulls file** in the definition of the protocol before the target.

Receive a file - explicit provisioning

This mode is for making a file available on the source for a target to pick up.

Source transfer properties

Optionally, set the **Transfer state** field to **Hold**. If you don't specify a value, it can be set at transfer execution time. When the target is the initiator, there is no transfer state option in the source.

Source file properties

- Select the **Single** option.
- In the **Filename** field, specify the full path and file name.

Target file properties

Optionally, specify a value in the **Filename** field. The value must contain a path that exists on the target and the file name of the received file. The default value is pub/&IDF.&IDTU.&FROOT.RCV. If you don't specify a value, it can be set at transfer execution time.

Transfer execution commands

On the Transfer CFT source, the command syntax to initiate the transfer is:

```
CFTUTIL send PART=<CFT_target>, IDF=<flow_ID>[, state=HOLD]
```

On the Transfer CFT target, the command syntax to retrieve the transfer is:

```
CFTUTIL recv IDF=<flow_ID>, part=<CFT_target>[, fname=<received_filename>]
```

Note The `fname` parameter is required if a value was not specified in the **Filename** field in the target file properties definition.

Receive a file - implicit provisioning

This mode is for making a file available for any requesting target to pick up from the source.

Source file properties

- Select the **Single** option.
- In the **Filename** field, specify the full path and file name. This is a required field, since the value cannot be set at transfer execution time.

Target file properties

Optionally, specify a value in the **Filename** field. The value must contain a path that exists on the target and the file name of the received file. The default value is pub/&IDF.&IDTU.&FROOT.RCV. If you don't specify a value here, it can be set at transfer execution time.

Transfer execution command

On the Transfer CFT target, the command syntax to retrieve the transfer is:

```
CFTUTIL recv IDF=<flow_ID>, part=<CFT_target>[, fname=<received_filename>]
```

Note The `fname` parameter is required if a value was not specified in the **Filename** field in the target file properties definition.

Receive a group of files - grouped, explicit provisioning

This mode is for making a group of files available for a target to pick up from the source. The source and target systems must have the same operating system. The source Transfer CFT must be configured to receive in this mode, which requires **Transmit files individually** set to **When necessary** in the Transfer processing section of the configuration.

Source file properties

- Select the **Multiple** option.
- In the **Path** field, specify the group of files to be sent using wildcard characters or a directory name. You can instead specify a file list.
- In the **Archive name** field, specify a value. The default is &IDF.&IDTU.RCV.

Target file properties

- Optionally, specify a value in the **Filename** field. The value must contain a path that exists on the target and the file name of the received file. The default value is pub/&IDF.&IDTU.&FROOT.RCV. If you don't specify a value, it can be set at transfer execution time.
- Optionally, in the **Temporary file** field, specify the name of the temporary archive. The default value is pub/&WFNAME.&IDTU.RCV. If you do not change the default value, pub/&IDF.&IDTU.&FROOT.RCV is created as the directory. If you don't specify a value, it can be set at transfer execution time.

Transfer execution command

On the Transfer CFT target, the command syntax to retrieve the transfer is:

```
CFTUTIL recv IDF=<flow_ID>, part=<CFT_target>[, fname=<dir_for_received_
files> ,wfname=<temporary_archive>]
```

Note The `fname` parameter and `wfname` parameters are required if values were not specified in the **Filename** and **Temporary file** fields in the target file properties definition.

Receive a group of files - file-by-file, implicit provisioning

This mode is for making a group of files available for any requesting target to pick up from the source. The source and target systems have different operating systems or **Transmit files individually** is set to **Always** in the Transfer processing section of the source system configuration

Source file properties

- Select the **Multiple** option.
- In the **Path** field, specify the group of files to send using wildcard characters or a directory name. You can instead specify a file list.

Target file properties

Optionally, specify a value in the **Filename** field. The value must contain a path that exists on the target and the file name of the received file. The default value is pub/&IDF.&IDTU.&FROOT.RCV. If you don't specify a value, it can be set at transfer execution time.

Transfer execution command

On the Transfer CFT target, the command syntax to retrieve the transfer is:

```
CFTUTIL recv IDF=<flow_ID>, part=<CFT_target>[, fname=<dir_for_received_
files>, FILE=ALL]
```

Note The `fname` parameter is required if a value was not specified in the **Filename** field in the target file properties definition.

Flow conversion, validation

Central Governance can convert implicitly Transfer CFT legacy flows to Central Governance flows. Central Governance enables you to reuse only IDs of the following objects in legacy flows when Central Governance flows are deployed:

- Send template
- Receive template
- Distribution list
- Partner

In relays, a partner is checked across all partners and distribution lists as well. A relay also can have a distribution list.

Other fields of legacy flows are not migrated automatically to Central Governance flows.

Central Governance flow IDs are validated across legacy flow IDs in saved and deployed statuses. The following tables show what is checked.

Transfer CFT source

Central Governance flow object	Legacy flow object
Flow identifier	Send template ID
Flow broadcast name	Distribution list ID Partner ID
Transfer CFT instance IDs of Transfer CFT target or relay	Partner ID Distribution list ID

Transfer CFT target

Central Governance element	Transfer CFT object
Flow identifier	Receive template ID
Flow collect list	Distribution list ID Partner ID
Transfer CFT instance IDs of Transfer CFT target or relay	Partner ID Distribution list ID

Transfer CFT relay

Central Governance element	Transfer CFT object
Transfer CFT instance IDs	Partner for the Transfer CFTs target and relay partners are available by name.

When conflicts are found

When conflicts are found between Central Governance flows and legacy flows, Central Governance displays messages asking users to confirm or cancel deployment actions.

- If the user confirms, reused objects are no longer accessible through the legacy flows user interface in Central Governance. Meanwhile, Central Governance deploys the flow on Transfer CFT, which updates the objects on Transfer CFT.

- If the user does not confirm, the action to deploy the Central Governance flow on Transfer CFT is canceled.

See *Transfer CFT legacy flows* on page 358 for more information about how Central Governance manages legacy flows in Transfer CFT.

Transfer CFT store-and-forward in flows

This topic describes how Transfer CFT store-and-forward paths are calculated in Central Governance flows.

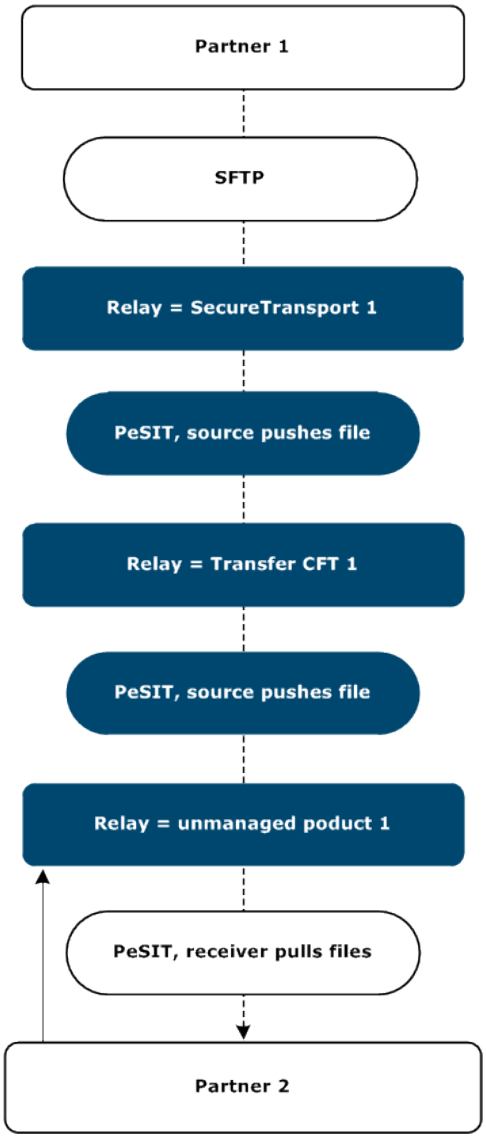
Transfer CFT, unmanaged products as relay

When Transfer CFT is used as a relay in a flow, it operates in store-and-forward mode. In a store-and-forward path there must be at least three consecutive segments of the flow, illustrated in the shaded parts of the graphic, where PeSIT is the protocol and the file direction is source pushes file. There can be several store-and-forward blocks in the same flow.

An unmanaged product can be part of the store-and-forward path, as shown in the graphic. An unmanaged product used as a relay is considered to be in store-and-forward mode and does not interrupt the path. On the other hand, an instance of SecureTransport that is a registered product and that is used as a relay in a flow cannot be in the path.

The unshaded parts of the graphic are not part of the store-and-forward path because:

- The protocol is SFTP between Partner 1 and SecureTransport 1.
- The protocol is PeSIT between the unmanaged product 1 relay and Partner 2, but the direction is receiver pulls file.



Flows with relays

The following describes a store-and-forward case where the flow is Transfer CFT 1 > Transfer CFT 2 > ... > Transfer CFT *n*.

First Transfer CFT

On the first Transfer CFT of a store-and-forward block:

- A partner definition (CFTPART) is generated for the last Transfer CFT in the block (target). Some parameters in CFTPART also are modified as:
  - IPART = identifier of the first next relay just after the first Transfer CFT
  - OMINTIME = 0
  - OMAXTIME = 0
- A partner definition is generated for the first next relay

Relay

On each intermediary relay:

- A partner definition is generated for the last Transfer CFT in the block (target) and some parameters in CFTPART also are also modified if the next Transfer CFT is an intermediary and not the target:
  - IPART = identifier of the first next Transfer CFT
  - OMINTIME = 0
  - OMAXTIME = 0
- A partner definition is generated for the previous Transfer CFT
- A partner definition is generated for the next Transfer CFT if the next is not the target
- If an acknowledgment is defined on the protocol just before this relay and if the Transfer CFT just before this relay is not the Transfer CFT source, a partner definition is generated for the first Transfer CFT in the block:
  - IPART = identifier of the previous Transfer CFT
  - OMINTIME = 0
  - OMAXTIME = 0

Transfer CFT target

On the Transfer CFT target (the last Transfer CFT in the store-and-forward path), if there is an acknowledgment defined on the protocol just before this Transfer CFT, a partner definition is generated for the Transfer CFT source:

- IPART = identifier of the previous CFT
- OMINTIME = 0
- OMAXTIME = 0

A partner definition is generated for the last previous relay.

Transfer CFT partner template

When a flow using Transfer CFTs is deployed, Central Governance deploys on each Transfer CFT the definition of the partners defined in the flow. There are two Transfer CFT partner objects involved: CFTPART and CFTTCP.

- For each CFTPART, Central Governance sets values for the fields ID, NRPART, NRPASSW, NSPART, NSPASSW, PROT, SAP, SSL (if mutual authentication is used)
- For each CFTTCP, Central Governance sets values for the fields ID, CLASS, HOST.

All the other fields of CFTTCP and CFTPART have the default values.

You can overwrite the following fields by editing the `com.axway.cmp.cgcft-default.cfg` file at `<install directory\runtime\com.axway.nodes.ume_<UUID>`. Changing this file does not require restarting Central Governance. Changes apply only to flows defined after the file is edited.

Transfer CFT field	Corresponding property in Central Governance configuration file	Central Governance value	Default Central Governance value if property is not set
CFTPART - ID	Name of the Transfer CFT partner.	%HOSTNAME% (product host as it appears in the product list)	STRING max_ length=32
CFTTCP – CNXIN, CNXOUT, CNXINOUT	cft.partner.cnxin cft.partner.cnxout cft.partner.cnxinout	0..1000	CNXIN = '64', CNXOUT = '64', CNXINOUT = '64'
CFTTCP – RETRYW, RETRYM, RETRYN	cft.partner.retryw cft.partner.retryn cft.partner.retrym	0..32767	RETRYW = '7', RETRYM = '12', RETRYN = '6',

The default Central Governance value is used if any property or value is missing.

File content example:

```
cft.partner.id=%HOSTNAME%

cft.partner.cnxin=32

cft.partner.cnxout=32

cft.partner.cnxinout=64

cft.partner.retryw=5

cft.partner.retryn=3
```

```
cft.partner.retrym=6
```

Transfer CFT broadcast and collect

Broadcast and collect in Transfer CFT are functions for sending files to multiple targets and receiving files from multiple sources with a single request.

Broadcast

Broadcasting a file is similar to using a distribution list. To use a broadcast list, the source must be an business application associated with Transfer CFT, and it must push files to the receiver.

For example, a file such as an updated product list, can be generated by a product catalog application and transmitted to multiple stores. In this scenario, the product catalog application is the source in the flow and the store applications are the targets.

When you enable broadcasting in source transfer properties, you also specify the name of the broadcast list. This name represents a logical list of all the targets defined in the flow. You can also define what occurs if a target is unknown. In the processing scripts definition, you can specify how the scripts are applied to the broadcast list with the unknown target option.

When the transfer of a single file is executed, one SEND command sends the file to all defined targets. The transfer list contains a record for the main SEND command and records for each target. When multiple files are transferred, the number of records in the transfer list depends on additional factors, such as whether the transfer is done in grouped or file-by-file mode.

When each transfer is completed, the state of the record in the transfer list changes to either the transferred (T) state or the executed (X) state, if processing scripts were executed after the transfer.

When all the transfers to all targets are successfully completed, the record associated with the main SEND command changes to the transferred (T) or the executed (X) state.

Collect

Collecting a file is the inverse of a broadcast list. To use a collect list, the target must be an business application associated with Transfer CFT, and it must pull files from the sender.

For example, each store generates a file containing daily sales information. These files are used to update an inventory management system. In this scenario, the inventory management system is the target in the flow and the stores are the sources.

When you enable collection in target transfer properties, you also specify the name of the collect list. This name represents a logical list of all the sources defined in the flow. You can also define what occurs if a source is unknown. In the processing scripts definition, you can specify how the scripts are applied to the collect list.

When the transfer of a single file is executed, one RECV command requests the file from all defined sources. The transfer list contains a record for the main RECV command and records for each source. When multiple files are transferred, the number of records in the transfer list depends on additional factors, such as whether the transfer is done in grouped or file-by-file mode.

When each transfer is completed, the state of the record in the transfer list changes to either the transferred (T) state or the executed (X) state, if processing scripts were executed after the transfer.

When all the transfers from all sources are successfully completed, the record associated with the main RECV command changes to the transferred (T) or the executed (X) state.

Transfer CFT bandwidth allocation

Central Governance sets the global incoming and outgoing transfer rates when Transfer CFT registers. You can manage these rates using policies or individual Transfer CFT definitions.

You can use the bandwidth allocation fields to manage data rates and the network bandwidth used for incoming and outgoing data in your flows.

Bandwidth allocation is based on the notion of a service class. A service class groups together all of the Transfer CFT services that facilitate data transfers. In Central Governance the transfer services are grouped according to what you define as their business priority: high, medium, or low. For example, when defining a flow you can specify the service class you want to use when executing the flow. For high priority flows you can use the high priority service class. For details see *Composition, deployment, execution* on page 230.

How allocation works

In the Transfer CFT configuration page you can distribute the available network bandwidth between these services. When setting values Central Governance scales them to 100 and orders them according to class priority in case the values do not satisfy these conditions. For example, if you specify the following bandwidth allocation:

High 10 percent, Medium 5 percent, Low 11 percent

When you save the settings, the system reallocates the settings as:

High 43 percent, Medium 38 percent, Low 19 percent

By automatically reordering the values, the system ensures there is more bandwidth for high-priority services than for low-priority services.

When Transfer CFT registers, Central Governance detects whether bandwidth allocation is enabled on the Transfer CFTside. It configures the bandwidth allocation between the three service classes (high, medium, low) using the default values.

Central Governance always overwrites the Transfer CFT configuration for bandwidth allocation unless:

- A specific Central Governance property is disabled. If disabled, the Bandwidth Allocation section for Transfer CFT configuration is not available in the Central Governance user interface. The property is:

  `enableBandwidthThrottling=true`

  True (enabled) is the default value. The property is in the `ume` runtime node at `/conf/com.axway.cmp.cftum-default.cfg`.

  or

- The `enableBandwidthThrottling` property is true and the Transfer CFT configuration fulfills the conditions cos.1 > cos.2 > cos.3 and the total value is 100.

Business scenario

The following is a business scenario for bandwidth allocation.

Two flows are defined from the product catalog application:

- **Flow 1: Product prices** - The product catalog application generates a file daily containing products and their prices. The file is sent to each store every night, so updated information is available before each store opens in the morning.
- **Flow 2: Products for sale** - The product catalog application generates a file daily containing products. The file is sent to the accounting management application to update the list of products for sale.

The product prices file is sent first. The definition of the first flow can ensure it takes a higher bandwidth allocation by setting the value of bandwidth allocation to high.

To enable significantly faster transfer rates for large-file transfers, traveling long distances over high bandwidth networks, you also can enable pTCP network protocol on Transfer CFTs and use it in the flow protocol definition.

Flow 1: Product prices

Prerequisite : Enable bandwidth allocation on the Transfer CFT used by the product catalog application.

1. Add source.
2. Add all targets.
3. Edit source transfer properties. Bandwidth allocation = High.
4. Optionally, edit other source properties.
5. Optionally, edit target properties.
6. Edit protocol definition between source and targets.
7. Save and deploy the flow.

Track a copied file

Transfer CFT supports copying a file to a different location without actually sending the file, yet still have the standard Transfer CFT tracking. This enables you to create a copied file, with visibility in Transfer CFT or Central Governance, without using network resources.

However, you must perform the configuration for this in Transfer CFT. For details see the topic "Use a shared disk as the data transfer medium" in the Transfer CFT User Guide.

Defining flows 22

Review the following prerequisites and outline for defining a flow.

Prerequisites

- Products that you intend to participate in the flow have been registered successfully in Central Governance. See *Product registration* on page 142.
- The following objects have been added in the user interface, as necessary to support the flow:
  - Applications, grouped or not. See *Applications* on page 213.
  - Partners. See *Partners* on page 219.
  - Unmanaged products. See *Unmanaged products* on page 226.

Flow definition outline

The following outlines tasks for defining a flow. Required and optional steps are identified.

1. Enter general information about the data flow, such as a name. This step is required.

   See *Add a flow* on page 274.

2. Add the source and target. Select whether the source or target is the initiator of the flow. The source is the owner of the data being transferred. The target is the receiver of the exchange. This step is required.

   See *Add source and target* on page 274.

3. If the flow goes through one or more intermediate systems, specify these relay points. This step is optional.

   See *Add a relay* on page 276.

4. Define the protocols between source and target. You also can define protocols between relays and source and target. This step is required.

   See *Specify the protocol* on page 277.

5. If the source is applications or a group of applications associated with Transfer CFTs, you can define the following information about the source: transfer properties, file properties, processing scripts. This step is optional.

   See *Transfer CFT source fields in flows* on page 327.

6. If the target is applications or a group of applications associated with Transfer CFTs, you can define the following information about the target: transfer properties, file properties, processing scripts. This step is optional.

See *Transfer CFT target fields in flows* on page 345.

7. Save the flow or save and deploy the flow. This step is required.

   See *Save and deploy a flow* on page 293.

Manage flows

This topic describes actions you can perform on flows in the user interface and provides links where appropriate to other topics for more details on some actions.

There are many steps in adding, configuring and editing flows. See *Defining flows* on page 272 for an outline of tasks.

Flow List page

The Flow List page is the starting point for performing actions on flows. Click **Flows** on the top toolbar to open the page. On this page you can:

Add

Click **Add flow** to add and configure a new flow. See *Add a flow* on page 274.

Deploy

Click **Deploy** to send a flow to the registered products defined in the flow. See *Save and deploy a flow* on page 293.

Remove

Select one or more flows and click **Remove** to remove them in Central Governance and the flow definition from all registered products where deployed.

Filter

Click **Filter** to filter the list of flows by name, status and other conditions.

Flow details page

Click the name of a flow on the Flow List page to open a details page for the flow. The page has the flow's name and information, if available, about tags, a description and a contact person.

Deploy

Click **Deploy** to send a flow to the registered products defined in the flow. See *Save and deploy a flow* on page 293.

Edit

Click **Edit** to change the flow configuration. If you remove a registered product from a flow, the flow definition also is removed from the product when the flow is redeployed. See *Defining flows* on page 272 for an outline of tasks.

Copy

Click **Copy** to add a flow with attributes of the original flow. The copy is identical, except the flow name is appended with *Copy<n>*, where *n* is the number of the copy, and a note in the Description field with the name of the original flow.

You can accept or change the default name and description.

Using a copy as the starting point, you can keep or change the original configuration as you want. Best practice is adding copies when you want multiple flows that differ only in details.

Add a flow

The following steps only specify the initial task in adding a flow, which is completing the general information section for a new flow. See *Defining flows* on page 272 for an outline of all tasks.

1. Select **Flows > Add flow**.
2. Type a friendly name for the flow. For example, the daily sales data from all the stores in the western region might be named West Daily Sales.
3. In the Details area, you can add tags to categorize the flow and a description to differentiate the flow when viewing a long list.
4. In the Contact area, you can provide information about the business owner of the flow.
5. Click **Save** to save the flow or continue defining the flow and save it later.

Add source and target

A flow requires at least one source and one target. However, you can have multiple sources and targets in a flow.

Prerequisites

- Applications are added and linked to registered products. See *Manage applications* on page 213.
- Application groups are added and contain at least one application. See *Manage application groups* on page 216.
- Partners used in flows are added. See *Manage partners* on page 219.

- Unmanaged products used in flows are added. See *Unmanaged products* on page 226.
- The required general information has been completed. See *Add a flow* on page 274.

Steps

1. Open the flow in edit mode, if not already opened.

   This might be a partially defined flow you saved previously or an unsaved flow still in the process of being defined.

   To open a flow for editing, click **Flows** on the top toolbar, click the name of a flow and then click **Edit**.
2. Click **Source** in the panel on the left side of the page.
3. Select a source type from the drop-down field. Objects you can select are:
   - Applications, which are associated with registered products. You can select:
     - A single application or multiple applications that are each linked to one instance of Transfer CFT.
     - A single application linked to a product other than Transfer CFT. For example, you can select one application linked to SecureTransport, but not two applications linked to SecureTransports.
   - Application groups. You can select only application groups containing applications linked to Transfer CFTs.
   - Partners.
   - Unmanaged products.

   For application groups, Central Governance only lets you select groups that contain the same types of products. It does not let you select groups that contain different types of products. For example, if:
   - Group 1 contains an application with one Transfer CFT and an application not associated with any product
   - Group 2 contains an application with one Transfer CFT and an application with one SecureTransport
   - Group 3 contains two applications with one Transfer CFT each

   Central Governance only lets you select Group 1 and Group 3.
4. After selecting a source type:
   - If the source type is applications, select a product type in the drop-down field. The product type corresponds to the type of products for each application selected in the flow. Click **Add source** to display a list of available applications, with products matching the selected product type. Select one or more applications and click **Select as source**. Click **Select Products** to choose the products to involve in the flow for each selected application. By default, all products matching the selected product type are selected. When no product is selected for the application, a warning displays in the notification column. The flow can be saved but cannot be deployed until this warning

has been resolved. To remove a source application, select an application and click **Remove**.

- If the source type is application groups or unmanaged products, click **Edit source**. Select one or more sources from the list and click **Select as source**.
- If the source type is partners, click **Add source** to display a list of available partners. Select one or more partners and click **Select as source**. To remove a source partner, select a partner and click **Remove**.

5. Click **Target** in the panel on the left side of the page.
6. Select one or more targets in the same manner as selecting sources. Click **Select as target**.

You can now edit the fields for the source and target. You can save the flow now or later.

If applications or application groups are changed after creating the flow, the change affects the flow status where the group already is used. You must redeploy the flow.

Next steps

If you select	See
Transfer CFT as source	*Transfer CFT source fields in flows* on page 327
SecureTransport as source	*Source fields in flows* on page 296
Transfer CFT as target	*Transfer CFT target fields in flows* on page 345

Add a relay

A relay in a flow is the product or unmanaged product that receives a file from the source and transmits it to the target or to another relay.

When you define multiple relays in a flow, the files are transferred from source to target through the relays in the order specified in the definition. For example, if a flow from a bank branch in California to a branch in Paris must pass through a relay in New York and one in Paris, the New York relay must be the first relay defined in the flow.

1. Open the flow in edit mode, if not already opened.

   This might be a partially defined flow you saved previously or an unsaved flow still in the process of being defined.

   To open a flow for editing, click **Flows** on the top toolbar, click the name of a flow and then click **Edit**.
2. Click **Relay** in the panel on the left side of the page.
3. Under **No relay selected**, click **Edit Relay** to show a list of products. Or, select **Unmanaged products** in the Select Relay From drop-down list to display unmanaged products.

You can use the Filter to refine the list.

4. Select a product or unmanaged product and click **Select as relay**. If the relay is SecureTransport, see *SecureTransport fields in flows* on page 296.
5. To add another relay, click **Relay** on the left side of the page again and repeat the previous steps.

If you select an unmanaged product as the relay and it is changed after creating the flow, the change affects the flow status where the unmanaged product is already in use. You must redeploy the flow manually.

To remove a relay, select the relay. Under **Relay Product**, place the cursor over the relay to remove. Click **X** to remove it.

Specify the protocol

You must define a protocol between the source and target in a flow. If there are relays between the source and target, you also must define a protocol between each of the following pairs or flow segments:

- Source and target
- Source and relay
- Relay and relay
- Relay and target

The configuration of any protocol must include the direction of the flow (sender pushes file or receiver pulls file) and technical details of the protocol.

Prerequisites

- The flow is open in add or edit mode in the user interface.
- General information is defined.
- Source, target and, if applicable, relays are specified.

Fields

Default values are provided in the user interface. Change the defaults to meet your needs.

Exchange protocol

The protocol used for the exchange.

- SecureTransport and partners support HTTP, FTP, PeSIT and SFTP.
- Transfer CFT, unmanaged products and applications associated with Transfer CFTs support PeSIT only.

Direction

Direction indicates the initiator of a file transfer in a flow.

- **Sender pushes file**. The sender is the initiator of the transfer request. The sender is the client and the receiver acts as the server.
- **Receiver pulls file**. The receiver is the initiator of the transfer request. The receiver is the client and the sender acts as the server. The receiver's request triggers the sender to send the file.

You can set direction for each protocol in the flow. If you want to change direction, the user interface enforces and warns when a protocol or direction is invalid.

The following describes the fields by protocol. Configuring protocols requires having communication profiles. See *Communication profiles* on page 236. Transfer CFT supports only PeSIT.

FTP

Connection mode

Indicates whether active mode or passive mode is used for transfers.

The server can support active mode, passive mode, or both.

SSL/TLS

Indicates whether the connection is secured via SSL or TLS.

- **Client optional** - Only the server must authenticate; the client might assure its identity. The server asks for the client's certificate during the SSL handshake, but allows the connection to proceed if the client does not present a certificate or when authentication with the presented certificate fails.
- **Mutual authentication** - Both parties authenticate themselves to assure identities.
- **None** - No security.
- **Server only** - Only the server must authenticate. The client does not present a certificate for authentication.

Enable FIPS transfer mode

When SSL/TLS is enabled, indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

Security mode

Indicates whether the security mode is explicit or implicit.

FTP supports two methods to accomplish security through a sequence of commands passed between two computers. The sequence is initiated with explicit (active) or implicit (passive) security.

- **Explicit security**. The initial connection is unencrypted. To establish the secure link, explicit security requires the FTP client to issue a specific command to the FTP server after establishing a connection. The default FTP server port is used.
- **Implicit security**. Implicit security begins with a secure connection as soon as the FTP client connects to an FTP server. The FTP server defines a specific port for the client to be used for secure connections.

Client communication profile

Indicates the client communication profile to use. You can use an existing profile or add a profile to use in the flow. If available, Central Governance suggests existing profiles you can use based on the selected protocol, connection mode, SSL/TLS, FIPS, security mode options and relevancy to the sender.

To add an FTP profile for a partner or SecureTransport, see *FTP client communication profile* on page 289.

Server communication profile

Indicates the server communication profile to use. You can use an existing profile in the flow. If available, Central Governance suggests existing profiles you can use based on the selected protocol, connection mode, SSL/TLS, FIPS, security mode options and relevancy to the sender.

More information about server communication profiles:

SecureTransport: *Network zone and server communication profile fields* on page 202

Transfer mode

Indicates the format of the transferred files. You can specify ASCII or binary or select autodetect to determine file format dynamically.

HTTP

Enable HTTP methods check

Indicates whether HTTP methods check is enabled.

HTTP methods

If HTTP methods check is enabled, indicates the method for moving the file.

- **PUT** - Requests the enclosed entity be stored under the supplied URI. If it refers to an existing resource, the URI is changed. If the URI does not point to an existing resource, the server can create the resource with that URI.

- **POST** - Requests the server accept the entity enclosed in the request as a new subordinate of the web resource identified by the URI. The data POSTed might be, for example, an annotation for existing resources.
- **GET** - Requests a representation of the specified resource. Requests using GET should only retrieve data and should have no other effect.

SSL/TLS

Indicates whether the connection is secured via SSL or TLS.

- **Client optional** - Only the server must authenticate; the client might assure its identity. The server asks for the client's certificate during the SSL handshake, but allows the connection to proceed if the client does not present a certificate or when authentication with the presented certificate fails.
- **Mutual authentication** - Both parties authenticate themselves to assure identities.
- **None** - No security.
- **Server only** - Only the server must authenticate. The client does not present a certificate for authentication.

Enable FIPS transfer mode

When SSL/TLS is enabled, indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

Client communication profile

Indicates the client communication profile to use. You can use an existing profile or add a profile to use in the flow. If available, Central Governance suggests existing profiles you can use based on the selected protocol, HTTP methods, SSL/TLS, FIPS, security mode options and relevancy to the sender.

To add an HTTP profile for a partner or SecureTransport, see *HTTP client communication profile* on page 283.

Server communication profile

Indicates the server communication profile to use. You can use an existing profile in the flow. If available, Central Governance suggests existing profiles you can use based on the selected protocol, HTTP methods, SSL/TLS, FIPS, security mode options and relevancy to the sender.

More information about server communication profiles:

SecureTransport: *Network zone and server communication profile fields* on page 202

Transfer mode

Indicates the format of the transferred files. You can specify ASCII or binary or select autodetect to determine file format dynamically.

PeSIT

Flow identifier

The unique identifier of the flow or flow segment. Each protocol within the flow can have the same ID or a different unique ID. See *Flow identifiers* on page 238.

Network protocol

Indicates the network protocol.

- **TCP** - The Transmission Control Protocol (TCP), one of the core protocols of the Internet protocol suite (IP), is often called TCP/IP. TCP provides reliable, ordered and error-checked delivery of a stream of octets between programs running on computers connected to a local area network, intranet or the public Internet. It resides at the transport layer.
- **pTCP** - The parallel Transmission Control Protocol (pTCP) is an end-to-end transport layer protocol that supports striped connections.
- **UDT** - The UDP-based Data Transfer Protocol (UDT) is a high performance data transfer protocol designed for transferring large volumetric data sets over high-speed wide-area networks.

See *About transfer acceleration* on page 174 for more information.

SSL/TLS

Indicates whether the connection is secured via SSL or TLS.

- **Client optional** - Only the server must authenticate; the client might assure its identity. The server asks for the client's certificate during the SSL handshake, but allows the connection to proceed if the client does not present a certificate or when authentication with the presented certificate fails.
- **Mutual authentication** - Both parties authenticate themselves to assure identities.
- **None** - No security.
- **Server only** - Only the server must authenticate. The client does not present a certificate for authentication.

Transfer CFT as server supports only options Mutual authentication and None.

Client communication profile

Indicates the client communication profile to use. You can use an existing profile or add a profile to use in the flow. If available, Central Governance suggests existing profiles you can use based on the selected protocol, network protocol, SSL/TLS, FIPS, security mode options and relevancy to the sender.

To add an PeSIT profile for a partner or SecureTransport, see *PeSIT client communication profile* on page 285.

Server communication profile

Indicates the server communication profile to use. You can use an existing profile in the flow. If available, Central Governance suggests existing profiles you can use based on the selected protocol, network protocol, SSL/TLS, FIPS, security mode options and relevancy to the sender.

More information about server communication profiles:

SecureTransport: *Network zone and server communication profile fields* on page 202

Acknowledgment

Indicates whether acknowledgments for transfers are enabled.

PeSIT properties

Indicates whether the file is compressed during transfer.

When multiple flows are defined between the same Transfer CFTs and security is different, commands for executing flows must include the PROT parameter. This ensures the security level defined in the flow is correct.

SFTP

Client authentication

Indicates the method for authenticating the server.

- **Public key** - The server's public key is used to authenticate the server.
- **Password** - The user name and password for connecting to the server is used to authenticate the server.
- **Password or public key** - The public key or password are used to authenticate the server.

Enable FIPS transfer mode

Indicates whether Federal Information Processing Standards (FIPS) is enabled for transfers. When enabled, the sender and the receiver must use FIPS-compliant ciphers and ciphers suites. Transfers fail if the sender and receiver do not provide them.

Client communication profile

Indicates the client communication profile to use. You can use an existing profile or add a profile to use in the flow. If available, Central Governance suggests existing profiles you can use based on the selected protocol, client authentication, SSL/TLS, FIPS, security mode options and relevancy to the sender.

To add an SFTP profile for a partner or SecureTransport, see *SFTP client communication profile* on page 287.

Server communication profile

Indicates the server communication profile to use. You can use an existing profile in the flow. If available, Central Governance suggests existing profiles you can use based on the selected protocol, client authentication, SSL/TLS, FIPS, security mode options and relevancy to the sender.

More information about server communication profiles:

SecureTransport: *Network zone and server communication profile fields* on page 202

Note When password or public key is used for server authentication, available server communication profiles also are the result of the union of server communication profiles configured with client authentication set at public key or password.

Transfer mode

Indicates the format of the transferred files. You can specify ASCII or binary or select autodetect to determine file format dynamically.

HTTP client communication profile

Defining an HTTP client communication profile depends on whether the client is an external partner or SecureTransport.

If the FIPS transfer mode is enabled, the client communication profile has the property enabled.

Prerequisites

- A flow is open in add or edit mode.
- You have selected HTTP as the protocol in a flow segment.
- You need to add a client communication profile.

See *Specify the protocol* on page 277 for more information.

Fields

Name

The unique name of the communication profile.

Network Zone

The network zone defines the product proxies to use for the transfer.

Select a specific network zone to use the proxy configuration defined for that zone.

The available network zones are defined for a registered product on its configuration page under the Products area of the user interface.

Login

The user name for the client to connect to the server

Password and confirm password

Password for connecting to the server.

Private or Public Certificate

When SSL/TLS is enabled, you can select an existing certificate or upload a new one. This is required when mutual authentication is selected, but optional when client optional is selected. For client optional you can select **No certificate authentication** as the client certificate option when you specifically do not want to use a certificate.

For a new certificate:

- If the certificate is for a product, use a file containing the product's private certificate. Supported file type is P12 (PKCS#12). You have to specify the certificate's password.
- If the certificate is not for a product, upload a file containing the server's public key certificate. Supported file types are DER, PEM and P7B (PKCS#7).

Specify an alias for a new certificate. This enables you to use the same certificate in multiple profiles.

The user interface warns if you try to add a duplicate alias. Aliases are unique by the objects related to them. For example, an alias for a partner certificate must be unique for a specific partner, but the same alias could be used for another partner.

Server certificate verified

Specifies whether the client verifies the server's certificate when communication between the two is being established. You can have the client verify the server certificate when client optional or server only is selected for SSL/TLS. The server certificate always is verified when mutual authentication is selected for SSL/TLS.

SSl/TLS setting	Server certificate verified
None	Not set and not displayed

SSl/TLS setting	Server certificate verified
Client optional	Displayed and by default **yes** regardless whether a certificate is presented
Server only	Displayed and by default **yes**
Mutual authentication	Not displayed and forced to **yes**

PeSIT client communication profile

Defining a PeSIT client communication profile depends on whether:

- The client is an external partner or SecureTransport.
- An SSL option is selected.

If the FIPS transfer mode is enabled, the client communication profile has the property enabled.

Prerequisites

- A flow is open in add or edit mode.
- You have selected PeSIT as the protocol in a flow segment.
- You need to add a client communication profile.

See *Specify the protocol* on page 277 for more information.

Fields

Default values are provided in the user interface. Change the defaults to meet your needs.

Name

Name

The unique name of the communication profile.

Network Zone

The network zone defines the product proxies to use for the transfer.

Select a specific network zone to use the proxy configuration defined for that zone.

The available network zones are defined for a registered product on its configuration page under the Products area of the user interface.

Login

The user name for the client to connect to the server

When SecureTransport is the initiator of the connection, the user name must be all upper-case letters and no more than 24 characters.

Password and confirm password

Password for connecting to the server.

When SecureTransport is the initiator of the connection, a password is optional and can be no more than eight characters.

Private or Public Certificate

When SSL/TLS is enabled, you can select an existing certificate or upload a new one. This is required when mutual authentication is selected, but optional when client optional is selected. For client optional you can select **No certificate authentication** as the client certificate option when you specifically do not want to use a certificate.

For a new certificate:

- If the certificate is for a product, use a file containing the product's private certificate. Supported file type is P12 (PKCS#12). You have to specify the certificate's password.
- If the certificate is not for a product, upload a file containing the server's public key certificate. Supported file types are DER, PEM and P7B (PKCS#7).

Specify an alias for a new certificate. This enables you to use the same certificate in multiple profiles.

The user interface warns if you try to add a duplicate alias. Aliases are unique by the objects related to them. For example, an alias for a partner certificate must be unique for a specific partner, but the same alias could be used for another partner.

Server certificate verified

Specifies whether the client verifies the server's certificate when communication between the two is being established. You can have the client verify the server certificate when client optional or server only is selected for SSL/TLS. The server certificate always is verified when mutual authentication is selected for SSL/TLS.

SSl/TLS setting	Server certificate verified
None	Not set and not displayed
Client optional	Displayed and by default **yes** regardless whether a certificate is presented
Server only	Displayed and by default **yes**

SSl/TLS setting	Server certificate verified
Mutual authentication	Not displayed and forced to **yes**

SFTP client communication profile

Defining an SFTP client communication profile depends on whether:

- The client is an external partner or SecureTransport.
- Client authentication is selected in the protocol definition.

If the FIPS transfer mode is enabled, the client communication profile has the property enabled.

Prerequisites

- A flow is open in add or edit mode.
- You have selected SFTP as the protocol in a flow segment.
- You need to add a client communication profile.

See *Specify the protocol* on page 277 for more information.

Fields

Default values are provided in the user interface. Change the defaults to meet your needs.

Name

Client authentication = password

Name

The unique name of the communication profile.

Network Zone

The network zone defines the product proxies to use for the transfer.

Select a specific network zone to use the proxy configuration defined for that zone.

The available network zones are defined for a registered product on its configuration page under the Products area of the user interface.

Login

The user name for the client to connect to the server

Password and confirm password

Password for connecting to the server.

Client authentication = public key

Name

The unique name of the communication profile.

Network Zone

The network zone defines the SecureTransport proxies to use for the transfer. This is available only when defining client communication profiles.

Select a specific network zone to use the proxy configuration defined for that zone.

The available network zones are defined for a registered SecureTransport on its configuration page under the Products area of the user interface.

Login

The user name for the client to connect to the server

Public or Private Key

The key the client uses to connect to the server. You can select an existing key or upload a new one.

Upload a public SSH key if the client is a partner. You must provide an alias for the key.

Select or upload a PKCS#8 or PEM password-protected private key if the client is SecureTransport. You must provide a password and an alias for the key.

Verify fingerprint

Indicates whether SecureTransport verifies the fingerprint of the SSH key against the value in the Fingerprint field. Connections are refused if values do not match. This field is available only for a product.

Fingerprint

If fingerprint verification is enabled, this field specifies the for value for verifying the fingerprint.

Client authentication = password or public key

Name

The unique name of the communication profile.

Client authentication

Indicates how the client connects to the server.

The final client authentication type depends on the setting in the server communication profile.

- If the server communication profile has password configured for the client authentication, password is the client authentication in the client communication profile. No action is required on the client communication profile.
- If the server communication profile has public key configured for the client authentication, public key is the client authentication in the client communication profile. No action is required on the client communication profile.
- If the server communication profile has password or public key configured for the client authentication, you can select password or public key as the client authentication in the client communication profile.

FTP client communication profile

Defining an FTP client communication profile depends on whether the client is an external partner or SecureTransport.

If the FIPS transfer mode is enabled, the client communication profile has the property enabled.

Prerequisites

- A flow is open in add or edit mode.
- You have selected FTP as the protocol in a flow segment.
- You need to add a client communication profile.

See *Specify the protocol* on page 277 for more information.

Fields

Name

The unique name of the communication profile.

Network Zone

The network zone defines the product proxies to use for the transfer.

Select a specific network zone to use the proxy configuration defined for that zone.

The available network zones are defined for a registered product on its configuration page under the Products area of the user interface.

Login

The user name for the client to connect to the server

Password and confirm password

Password for connecting to the server.

Private or Public Certificate

When SSL/TLS is enabled, you can select an existing certificate or upload a new one. This is required when mutual authentication is selected, but optional when client optional is selected. For client optional you can select **No certificate authentication** as the client certificate option when you specifically do not want to use a certificate.

For a new certificate:

- If the certificate is for a product, use a file containing the product's private certificate. Supported file type is P12 (PKCS#12). You have to specify the certificate's password.
- If the certificate is not for a product, upload a file containing the server's public key certificate. Supported file types are DER, PEM and P7B (PKCS#7).

Specify an alias for a new certificate. This enables you to use the same certificate in multiple profiles.

The user interface warns if you try to add a duplicate alias. Aliases are unique by the objects related to them. For example, an alias for a partner certificate must be unique for a specific partner, but the same alias could be used for another partner.

Server certificate verified

Specifies whether the client verifies the server's certificate when communication between the two is being established. You can have the client verify the server certificate when client optional or server only is selected for SSL/TLS. The server certificate always is verified when mutual authentication is selected for SSL/TLS.

SSl/TLS setting	Server certificate verified
None	Not set and not displayed
Client optional	Displayed and by default **yes** regardless whether a certificate is presented
Server only	Displayed and by default **yes**
Mutual authentication	Not displayed and forced to **yes**

Symbolic variables

A symbolic variable is a representation of a value that is not known at the time the flow is defined, but only at the time the transfer is executed.

The syntax of the variable is &VAR, where VAR is the variable name. For example, the variable &FNAME represents the name of the file at the source.

The table lists the variables grouped by category, the value represented by the variable, and the maximum length of the substituted value in characters. For ease of scanning, the variables are listed without the leading ampersand (&).

Category	Variable	Value	Max Length
Application	IDA	Application identifier	64
	PARM	Relates to the **Additional information** field on the Source transfer properties page of the flow definition	513
	RAPPL	Target application name	48
	SAPPL	Source application name	48
Date/time a file was made available for transfer	FDATE	Date	8
	FTIME	Time	8
	FYEAR	Year	4
	FMONTH	Month	2
	FDAY	Day	2
Date/time associated with a transfer	BDATE	Transfer start date	8
	BTIME	Transfer start time	8
	BYEAR	Transfer start year	2
	BMONTH	Transfer start month	2
	BDAY	Transfer start day	2
Date/time of the system	SYSDATE	System date	8
	SYSTIME	System time	8

Category	Variable	Value	Max Length
File	FNAME	Name of the physical file at the source including the path to the file. For example, `/usr/TATA/tmp/TOTO.TXT`	512
	FPATH	Path to the physical file at the source. For example, `/usr/TATA/tmp/`	512
	FROOT	Name of the physical file at the source. For example, `TOTO.TXT`	512
	IDF	Flow identifier	32
	NFNAME	Name of the file as it is being transmitted over the network	512
	NFVER	Version number of the file	255
	NIDF	Identifier of the flow as it is being transmitted over the network	512
Source/target Transfer CFT	GROUP	Group to which the source or target belongs	32
	PART	Identifier of the Transfer CFT on which you are using the variable	32
	IPART	Identifier of the Transfer CFT that is the relay in the flow	32
	NPART	Network name of the source or target, depending on the direction of the transfer	32
	RPART	Name of the target	32
	SPART	Name of the source	32

Category	Variable	Value	Max Length
Transfer	DIRECT	Indicates the transfer direction, that is, whether the file is sent or received. For example, if combined with other variables, it may detect if the file was sent twice.	4
	IDT	Identifies a transfer for a given source or target and the transfer direction. It does not ensure the uniqueness in the transfer list. To ensure uniqueness, the IDTU variable is used.	8
	IDTU	Transfer list identifier for the transfer. When there are several Transfer CFT systems on the same computer, this variable guarantees the uniqueness among the systems sharing the same transfer list.	8
	MODE	Indicates whether the transfer was initiated by the source (S) or by the target (R)	1
User	RUSER	Name of user receiving the file or files	32
	SUSER	Name of user sending the file or files	32

Save and deploy a flow

After defining a flow, you can save it or save and deploy it. This enables you to prepare a deployment before pushing the configuration to the source and target systems. After deployment, when the flow is executed, the configuration changes defined in the flow are used.

Click **Save** to store the flow definition locally in Central Governance. You can do this if you still have changes to make or if you are not ready to deploy the flow.

Click **Deploy** to save the definition and push it to the products in the flow. For example:

1. You define a flow with Host A as the source and Host B as the target.
2. You deploy the flow to Host A and Host B.
3. You change a field in the source properties section of the flow and deploy the flow. Only Host A is updated, since no changes were made to the target properties.
4. You add Host C as a target and deploy the flow. In addition to Host C, the flow is deployed to Host A, which requires the information about the new target.

For Transfer CFT, Central Governance uses the Copilot WebService to deploy flows to Transfer CFT.

For SecureTransport, Central Governance uses SecureTransport RESTful APIs to deploy flows to SecureTransport.

Back up flows from UI

You can export files of flows to a specified directory from the Central Governance user interface. The intent behind this backup feature is promoting flows from one instance of Central Governance to another. For example, promote from a development or staging environment to a production environment. After backing up flows, you could have an external process retrieve the files and import to another instance. However, Central Governance only can back up flows; any external process for retrieving and importing is outside its control.

UI flow backup is an alternative to using CLI to export flows to files. The UI feature only can write flow files to a static directory accessible to the server, while CLI can export to any server or local directory when used remotely. See *flowExport* on page 85.

None of the Central Governance default user roles gives permissions for backing up flows from the UI. Users who want to must have a role with the Central Governance Backup Flow predefined privilege. Or, create a user-defined privilege based on the Central Governance Flow resource with the View and Backup actions enabled and add it to a role. The Flow resource is FGAC-enabled (see *Fine-grained access control* on page 122).

See *Prerequisites for promoting flows* on page 398 for more information about promoting flows.

Use flow backup

1. Click **Flows** on the top toolbar in Central Governance to open the Flow List page. The control is available only for users whose role gives permission to back up flows from the UI.
2. Select one or more flows and click **Backup**. Central Governance responds with a message indicating whether backup has succeeded or failed. Success messages list the names of the backed-up flows.

Central Governance writes the file to the configured backup directory. The default directory is `<install directory>/backup`. The file is in the following format:

```
export_flow_<timestamp>.json
```

If you create multiple backup files, only the timestamp differentiates one file from another. If you need to identify the flows in a file, open the file to review its contents.

Change flow backup directory

Use this procedure if you want to change the default backup directory.

1. Create a directory to store the flow backup files. This can be any directory Central Governance can access. The external process for promoting flows must have access, too. Also, the user who starts Central Governance must have rights to write files to the directory.

2. Navigate to the file named `com.axway.cmp.flow-default.cfg` under:

   `<Central Governance install directory>/runtime/com.axway.nodes.ume_<UUID>/conf/`

3. Edit the property to point it to your directory:

   `backup.dir=<path to the backup directory>`

   For example, on Linux the directory could be

   `/home/<user>/<mybackupdir>`

4. Save the file. You do not have to restart Central Governance.

SecureTransport fields in flows 23

The following topics describe by protocol the send, file processing and receive fields for SecureTransport in flows defined in Central Governance.

Receive and send properties are different depending on direction of a flow segment: sender pushes file or receiver pulls file. File processing properties are the same regardless of direction.

When the direction is sender pushes files:

- Receive properties control the behavior of a transfer when files are pushed to SecureTransport. You can define receive properties when SecureTransport is used as a relay
- Send properties control the behavior of transfers when SecureTransport pushes files to the receiver.

When the direction is receiver pulls files:

- Receive properties control the behavior of a transfer when SecureTransport pulls files from the sender. You can define receive properties when SecureTransport is used as a relay.
- Send properties control the behavior of transfers when SecureTransport publishes files to a receiver that pulls them.

File processing properties control whether transferred files are compressed or decompressed, PGP encrypted or decrypted, or have specified line endings.

Source fields in flows

The following describes the fields when SecureTransport is used as the source in a flow. See *SecureTransport as source in flows* on page 250 for an example of SecureTransport as the source in a flow.

Receive properties

The following are the receive properties when SecureTransport is the source in a flow.

Folder monitoring

The following fields specify how SecureTransport scans the application directory.

Directory to scan

The absolute path for the directory where the application puts files for SecureTransport.

File filter

The method for filtering the files in the directory to scan.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport Administrator Guide for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are filtered.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are filtered.

Scan sub-directories

Specify whether files in subdirectories of the directory to scan are filtered.

Directory depth

When scan sub-directories is enabled, specifies the depth of subdirectories to scan.

Sub-directory filter

Specifies the names of subdirectories to scan. You can use regular expressions or file globbing.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport Administrator Guide for details. For example, if you specify (ORDERS|INVOICES), only ORDERS and INVOICES directories are scanned.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***PART***, all directories containing PART are scanned.

Scheduler

You can specify a schedule for retrieving files from the application directory. If not enabled, folder monitoring is triggered according to a global set of properties that is set externally and not in Central Governance. When enabled, you can set a one-time or recurring schedule for monitoring files. You can specify a frequency ranging from daily to annually and set specific times of day. You also can define a validity period by setting start and end dates.

File properties

Directory

The path of the directory where files retrieved via folder monitoring are moved. These files are processed for sending as defined in send properties.

The directory value is relative to the home folder of the generic account for monitoring application folders defined in SecureTransport.

For SecureTransport on Linux, the directory name cannot be equal to:

.. or .

The name also cannot contain:

/../ or /./ or // or : * ? " < > |

It cannot start with:

../ or ./ or ~

And it cannot end with:

/.. or /.

For SecureTransport on Windows, the directory name cannot contain drive letters or the following characters:

/ * ? " < > |

Receive file as

Name of the files scanned when files are retrieved form an application folder and moved on the SecureTransport side. The value is relative to the generic SecureTransport account home directory. It can contain any valid expression.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

Post-reception actions

On failure

Specifies the action to take when transfers fail. A failure occurs when the transfer is incomplete and all retry attempts have failed. You can select:

No Action causes the files to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files.

File processing

See *File processing properties in flows* on page 317.

Send properties

The send properties depend on the flow direction and the protocol between SecureTransport and the next participant in the flow. See *Send properties in flows* on page 301.

Target fields in flows

The following describes the fields when SecureTransport is used as the target in a flow. See *SecureTransport as target in flows* on page 255 for an example of SecureTransport as the target in a flow.

Receive properties

Receive properties depend on the exchange protocol and direction between SecureTransport and the preceding participant to the flow. See *Receive properties in flows* on page 309.

File processing

See *File processing properties in flows* on page 317.

Send properties

Upload directory

The upload directory represents the absolute path where files are pushed. Validation is done only if Use expression language is No. Otherwise, you can use expressions like the following where the current date is appended to the target file's name.

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

For SecureTransport on Linux, the value cannot equal:

.. or .

And it cannot contain:

/../ or /./ or // or :*?"<>|

Nor two or more of the following characters in a sequence:

( ) _ - + = { } ~ ! @ # $ % ^ & ; "

It also cannot start with:

../ or ./ or ~

Or end with:

/.. or /.

For SecureTransport on Windows the folder path cannot contain a drive letter.

Use expression language

When enabled the upload folder can contain expressions.

Create directory if not existent

An upload folder is created if it doesn't exist. The folder is owned by the user running the SecureTransport TM server process.

Upload file as

Use to rename the file to push. You can use an expression to specify a file name.

Post-sending sections

On failure

Specifies the action to take when transfers fail. A failure occurs when the transfer is incomplete and all retry attempts have failed. You can select:

No Action causes the files to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

On success

Specifies the action to take when transfers succeed. The **No action** and **Move/Rename** actions are the same as for failure.

Send properties in flows

The following are the send properties fields for all protocols in flows using SecureTransport. Fields are described when the direction is sender pushes file and receiver pulls file. Default values are provided in the user interface. Change values to meet your needs.

Prerequisites

- The flow is open in add or edit mode in the user interface.
- SecureTransport is in the flow as a source, target or relay.

Multiple receivers

If a flow has multiple receivers, the user interface for send properties behaves differently depending on the protocol.

Protocol after SecureTransport is SFTP, FTP or HTTP

If the protocol after SecureTransport in the flow is SFTP, FTP or HTTP, the UI displays a table that enables you to edit the send properties for each receiver by clicking **Edit** next to the receiver's name. The Status column in the table indicates whether the properties for each receiver are configured properly.

Protocol after SecureTransport is PeSIT

If the protocol after SecureTransport in the flow is PeSIT, the send properties, with one exception, are the same for all receivers, according to the flow direction set after SecureTransport. The file filter is the only property configured for each receiver.

If the receiver is a group of applications, a file filter is configured for each application in the group.

SFTP, FTP, HTTP: Send properties, sender pushes file

The following are the send properties for SFTP, FTP and HTTP when SecureTransport is in the flow and the direction is sender pushes file.

File properties

Remote directory

Represents the directory on the receiver where SecureTransport pushes files.

The directory value cannot contain:

\ * " < > |

File name sent

You can use the expression language to specify the criteria you want to match. The expression uses the criteria provided to create a new file name from the original file name.

If the field is not used, the name of the received file is unchanged when the file is sent.

See regular expressions topics in the SecureTransport Administrator Guide for details.

Example 1. New file name based on the current file name (since the transformation might have changed it):

```
${basename(currentfulltarget)}.sent
```

Example 2. New file name based on the original filename with a timestamp:

```
${basename(transfer.target)}..${timestamp}.${extension
(transfer.target)}
```

File filter

Represents, when set, the filter on files SecureTransport pulls from the remote directory. You can select:

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport Administrator Guide for details. For example, if you specify ***\.(xml|txt)**, all XML and TXT files are downloaded.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are downloaded.

If you do not use the field, all files in the remote directory are downloaded.

Post-sending actions

On failure

Specifies the action to take when transfers fail. A failure occurs when the transfer is incomplete and all retry attempts have failed. You can select:

No Action causes the files to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

On success

Specifies the action to take when transfers success. The actions are the same as for on failure.

SFTP, FTP, HTTP: Send properties, receiver pulls file

The following are the send properties for SFTP, FTP and HTTP when SecureTransport is in the flow and the direction is receiver pulls file.

Transfer properties

File exists

Detects duplicate transfers on the remote directory. Actions you can specify for duplicates are:

Cancel refuses the transfer.

Overwrite replaces an existing file by overwriting it.

Rename allows the transfer but renames the existing file.

Rename transferred file leaves the existing file name as-is, but renames the transferred file.

Append renames the transferred file by appending it. The name is derived from the name specified in the **Publish file as** field. For example:

If file name is `myFile.txt`, the file is renamed `myFile (new copy 1).txt`.

However, if this file already exists, the file is renamed `myFile (new copy 2).txt`.

If the file doesn't have an extension, the name transformation is the same but without the extension. For example, `myFile (new copy 1)`.

File properties

Directory

The directory where SecureTransport makes available files for the receiver to pull. The directory value cannot contain the following characters:

\ * " < > |

Publish file as

Represents the name of the published file. If this field is not used, the name used is the name of the file SecureTransport received from the sender.

You can use the regular expression language to specify creating a file name. See regular expressions topics in the SecureTransport Administrator Guide for details.

Example 1. New file name based on the current filename (since the transformation might have changed it):

```
${basename(currentfulltarget)}.sent
```

Example 2. New file name based on the original filename with a timestamp:

```
${basename(transfer.target)}..${timestamp}.${extension
(transfer.target)}
```

File filter

Represents, when set, the filter on files SecureTransport publishes to the receiver. You can select:

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport Administrator Guide for details. For example, if you specify ***\.(xml|txt)**, all XML and TXT files are downloaded.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are downloaded.

If you do not use the field, all files in the remote directory are downloaded.

Post-sending actions

On failure

Specifies the action to take when transfers fail. A failure occurs when the transfer is incomplete and all retry attempts have failed. You can select:

No Action causes the files to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

On success

Specifies the action to take when transfers success. The actions are the same as for on failure.

Post-download actions

On success

Specifies the action to take after files are downloaded.

No Action specifies nothing is done.

no actions take place on the downloaded files.

Delete removes the file from the directory where SecureTransport received the it.

PeSIT: Send properties, sender pushes file

The following are the send properties when SecureTransport is in the flow and the direction is sender pushes file over PeSIT.

Transfer properties

User message

Overrides the predefined user message (PI99) in the PeSIT transfer site. To preserve the predefined user message, leave the field blank or enter the expression language expression `${pesit.pi.serviceParam}`. This field corresponds to the PARM in Transfer CFT.

File properties

File name sent

Overrides the file label (PI37) predefined in the transfer profile. If you select **Custom**, you must enter a string. This field corresponds to the NFNAME in Transfer CFT.

File filter

Select the method for filtering the files.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport Administrator Guide for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are processed.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are processed.

File encoding file type

Specifies the file type. Select **Binary** to send a mix of file types and ensure there is no custom encoding or transcoding required for the text files. Select **EBCDIC (native)** for files that use EBCDIC LF (0x25) as the end-of-line character.

Record format record type

Indicates whether the records in the file are fixed or variable length.

Maximum record length

Specifies in bytes the maximum length of the records.

Post-sending actions

On failure

Specifies the action to take when transfers fail. A failure occurs when the transfer is incomplete and all retry attempts have failed. You can select:

No Action causes the files to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

On success

Specifies the action to take when transfers success. The actions are the same as for on failure.

PeSIT: Send properties, receiver pulls file

The following are the send properties when SecureTransport is in the flow and the direction is receiver pulls file over PeSIT.

Transfer properties

File exists

Detects duplicate transfers on the remote directory. Actions you can specify for duplicates are:

Cancel refuses the transfer.

Overwrite replaces an existing file by overwriting it.

Rename allows the transfer but renames the existing file. For example, if the file is named `myFile.txt`, it is renamed `myFile(old copy 1).txt`. But if that file already exists, the new name is `myFile (old copy 2).txt`.

Rename transferred file leaves the existing file name as-is, but renames the transferred file. The name is derived from the name specified in the **Publish file as** field. If file name is `myFile.txt`, it is renamed `myFile (new copy 1).txt`. But if this file already exists, the file is renamed `myFile (new copy 2).txt`. If the file doesn't have an extension, the name transformation is the same but without the extension. For example, `myFile (new copy 1)`.

Append renames the transferred file by appending it.

File properties

File name sent

Overrides the file label (PI37) predefined in the transfer profile. If you select **Custom**, you must enter a string. This field corresponds to the NFNAME in Transfer CFT.

Files to send

Indicates whether one or multiple files are transferred.

File filter

Select the method for filtering the files.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport Administrator Guide for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are processed.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are processed.

Publish file as

Represents the name of the published file. If this field is not used, the name used is the name of the file SecureTransport received from the sender.

You can use the regular expression language to specify creating a file name. See regular expressions topics in the SecureTransport Administrator Guide for details.

Example 1. New file name based on the current filename (since the transformation might have changed it):

```
${basename(currentfulltarget)}.sent
```

Example 2. New file name based on the original filename with a timestamp:

```
${basename(transfer.target)}..${timestamp}.${extension
(transfer.target)}
```

File encoding file type

Specifies the file type. Select **Binary** to send a mix of file types and ensure there is no custom encoding or transcoding required for the text files. Select **EBCDIC (native)** for files that use EBCDIC LF (0x25) as the end-of-line character.

Record format record type

Indicates whether the records in the file are fixed or variable length.

Maximum record length

Specifies in bytes the maximum length of the records.

Post-sending actions

On failure

Specifies the action to take when transfers fail. A failure occurs when the transfer is incomplete and all retry attempts have failed. You can select:

No Action causes the files to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

On success

Specifies the action to take when transfers success. The actions are the same as for on failure.

Post-download actions

On success

Specifies the action to take after files are downloaded.

No Action specifies nothing is done.

no actions take place on the downloaded files.

Delete removes the file from the directory where SecureTransport received the it.

Receive properties in flows

The following are the receive properties fields for all protocols in flows using SecureTransport. Fields are described when the direction is sender pushes file and receiver pulls file. Default values are provided in the user interface. Change values to meet your needs.

Prerequisites

- The flow is open in add or edit mode in the user interface.
- SecureTransport is in the flow as a source, target or relay.

Multiple senders

If a flow has multiple senders, the user interface for receive properties behaves differently depending on the protocol.

Protocol preceding SecureTransport is SFTP, FTP or HTTP

If the protocol preceding SecureTransport in the flow is SFTP, FTP or HTTP, the UI displays a table that enables you to edit the receive properties for each sender by clicking **Edit** next to the sender's name. The Status column in the table indicates whether the properties for each sender are configured properly.

Protocol preceding SecureTransport is PeSIT

If the protocol preceding SecureTransport in the flow is PeSIT, the receive properties are the same for all senders, according to the flow direction set before SecureTransport.

SFTP, FTP, HTTP: Receive properties, sender pushes file

The following are the receive properties for SFTP, FTP and HTTP when SecureTransport is in the flow and the direction is sender pushes file.

File properties

Directory

The directory represents the path where the sender can push files to SecureTransport. Files received in this directory are handled as defined in *Receive properties in flows* on page 309 and processed for sending as defined in *Receive properties in flows* on page 309.

The directory value is relative to the home folder of the sender account defined in SecureTransport.

For SecureTransport on Linux, the directory name cannot be equal to:

.. or .

The name also cannot contain:

/../ or /./ or // or : * ? " < > |

It cannot start with:

../ or ./ or ~

And it cannot end with:

/.. or /.

For SecureTransport on Windows, the directory name cannot contain drive letters or the following characters:

/ * ? " < > |

Post-reception actions

On failure

Specifies the action to take when transfers fail. A failure occurs when the transfer is incomplete and all retry attempts have failed. You can select:

No Action causes the files to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

SFTP, FTP, HTTP: Receive properties, receiver pulls file

The following are the receive properties for SFTP, FTP and HTTP when SecureTransport is in the flow and the direction is receiver pulls file.

File properties

Remote directory

Represents the folder on the sender where SecureTransport pulls files. The directory value cannot contain:

\ * " < > |

File filter

Represents, when set, the filter on files SecureTransport pulls from the remote directory. You can select:

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport Administrator Guide for details. For example, if you specify ***\.(xml|txt)**, all XML and TXT files are downloaded.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are downloaded.

If you do not use the field, all files in the remote directory are downloaded.

Directory

The directory represents the path where the sender can push files to SecureTransport. Files received in this directory are handled as defined in *Receive properties in flows* on page 309 and processed for sending as defined in *Receive properties in flows* on page 309.

For SecureTransport on Linux, the directory name cannot be equal to:

.. or .

Also, the name cannot contain:

/../ or /./ or // or : * ? " < > |

It also cannot start with:

../ or ./ or ~

And it cannot end with:

/.. or /.

For SecureTransport on Windows, the directory name cannot contain drive letters or the following characters:

/ * ? " < > |

Scheduler

You can specify a schedule for retrieving files from the sender. If not enabled, files are pulled according to a definition that is set externally and not in Central Governance.

When enabled, you can set a one-time or recurring schedule for pulling files. You can specify a frequency ranging from daily to annually and set specific times of day. You also can define a validity period by setting start and end dates for a schedule.

Post-reception actions

On failure

Specifies the action to take when transfers fail. A failure occurs when the transfer is incomplete and all retry attempts have failed. You can select:

No Action causes the files to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

On success

Specifies the action to take when transfers succeed. You can select:

No Action causes the files to stay in the original location. If the receiver trigger the transfer again, the original file is pulled again.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files. This option is available only for SFTP and FTP.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

PeSIT: Receive properties, sender pushes file

The following are the receive properties for PeSIT when SecureTransport is in the flow and the direction is sender pushes file.

Receive properties in flows control the behavior of a transfer when files are pushed over PeSIT to SecureTransport. You can define receive properties when SecureTransport is used as relay.

File properties

Receive file as

Name pf the files received when files are transferred to SecureTransport. The value is relative to the sender's account home directory. It can contain any valid expression, including PeSIT expressions. The suggested default value is:

```
${basename(pesit.fileLabel)}-${timestamp}-
${pesit.transferID}${extension(pesit.fileLabel)}
```

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

File encoding file type

Specifies the file type. Select **Binary** to send a mix of file types and ensure there is no custom encoding or transcoding required for the text files. Select **EBCDIC (native)** for files that use EBCDIC LF (0x25) as the end-of-line character.

Record format record type

Indicates whether the records in the file are fixed or variable length.

Maximum record length

Specifies in bytes the maximum length of the records.

Post-reception actions

On failure

Specifies the action to take when transfers fail. A failure occurs when the transfer is incomplete and all retry attempts have failed. You can select:

No Action causes the files to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

PeSIT: Receive properties, receiver pulls file

The following are the receive properties for PeSIT when SecureTransport is in the flow and the direction is receiver pulls file.

Receive properties in flows control the behavior of a transfer when SecureTransport pulls files from the sender over PeSIT. You can define receive properties when SecureTransport is used as relay.

File properties

Receive file as

Name pf the files received when files are transferred to SecureTransport. The value is relative to the sender's account home directory. It can contain any valid expression, including PeSIT expressions. The suggested default value is:

```
${basename(pesit.fileLabel)}-${timestamp}-
${pesit.transferID}${extension(pesit.fileLabel)}
```

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

Files to receive

Indicates whether SecureTransport receives one file or multiple files.

One file means SecureTransport gets the first available file.

Multiple files mean SecureTransport gets all available files.

File encoding file type

Specifies the file type. Select **Binary** to send a mix of file types and ensure there is no custom encoding or transcoding required for the text files. Select **EBCDIC (native)** for files that use EBCDIC LF (0x25) as the end-of-line character.

Record format record type

Indicates whether the records in the file are fixed or variable length.

Maximum record length

Specifies in bytes the maximum length of the records.

Scheduler

You can specify a schedule for retrieving files from the sender. If not enabled, files are pulled according to a definition that is set externally and not in Central Governance.

When enabled, you can set a one-time or recurring schedule for pulling files. You can specify a frequency ranging from daily to annually and set specific times of day. You also can define a validity period by setting start and end dates for a schedule.

Post-reception actions

On failure

Specifies the action to take when transfers fail. A failure occurs when the transfer is incomplete and all retry attempts have failed. You can select:

No Action causes the files to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten.

Delete removes the files from the original location.

Move/Rename File requires you to specify a directory to move the files and an expression for renaming the files.

If a file name expression begins with /, the transformed file is put in the subfolder indicated by this file name expression relative to the subscription folder. Otherwise, the transformed file is put in the subfolder indicated by this file name expression relative to the source file folder.

Example 1. Append current date to the target file name:

```
${stenv['target']}_${date('yyyyddMMHHmmss')}
```

Example 2. Append a random ID to the file name:

```
${stenv['target']}_${random()}
```

File processing properties in flows

The following are the file processing properties when SecureTransport is in the flow. The fields are the same regardless of direction or protocol.

If a flow has multiple senders and receivers, the user interface displays a table that enables you to configure the file processing properties for each sender-receiver pair by clicking **Edit** next to the pair's names. The Status column in the table indicates whether the properties for each pair are configured properly.

Condition

Represents, when configured, a set of comparison expressions, functions and logical operations for defining the files to route to the receiver. When no value is set, the trigger condition is always used. When a value is defined, the trigger condition is based on the entered expression. Use expression language to define the route trigger condition.

See regular expressions topics in the SecureTransport Administrator Guide for details.

Example 1. Files uploaded only through a specific protocol:

```
${extension(transfer.target) eq '.txt'}
```

Example 2. Files uploaded from specific partner over PeSIT:

```
${pesit.pi.senderID.toLowerCase() eq 'partner'}
```

Processing type

Select a processing type and see the fields and descriptions for the type. You can enter a description for the selected processing type.

PGP encryption

Enable to encrypt or sign files, or both, with PGP keys. The matching PGP keys for encrypting or signing are detected at runtime. The keys must exist on SecureTransport or transfers fail. Only keys in non-delimited ASCII (ASC) file format are supported.

File filter

Select the method for filtering the files to process.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are compressed.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport documentation for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are compressed.

Operations

Select the encryption and signature settings.

Encrypt and sign means all files are encrypted and signed.

Encrypt only means all files are encrypted but not signed.

Sign only means all files are signed but not encrypted.

Encryption key

Specifies the SecureTransport public PGP key for encrypting files to partners, applications, unmanaged products or other relays.

You can select the alias of an existing public PGP key or upload a new key. For a new key, upload the key in an ASC file and specify an alias. The user interface warns if you try to add a duplicate alias. All aliases are unique on SecureTransport.

Signing key

Specifies the SecureTransport private PGP key for signing files to partners, applications, unmanaged products or other relays.

Once you provide a password, you can select the alias of an existing private PGP key or upload a new key. For a new key, upload the key in an ASC file and specify an alias. The user interface warns if you try to add a duplicate alias. All aliases are unique on SecureTransport.

Compression

Specifies the type of compression.

Use preferred uses recipient's PGP key to determine the compression method. If the data compression method you choose is not one of the recipient’s preferred methods, the recipient cannot access the data.

Compression level

Specifies the level of compression. Fast to Best represent the compression ratio. As compression file size decreases, the time to compress increases.

Encode using ASCII armor

Specifies whether ASCII armor encoding is used. ASCII armor refers to using binary-to-text encoding for plain text data.

Rename file as

Specifies a regular expression to name output files.

Example 1. New file name based on the current file name:

```
${basename(currentfulltarget)}.transformed
```

Example 2. New file name based on the original file name with a timestamp:

```
${basename(currentfulltarget)}.${timestamp}.${extension
(currentfulltarget)}
```

Example 3. New file name based on its name after the transformation (for example, name

of the file as extracted from an archive) by appending a timestamp:

```
${transformedfilename}.${timestamp}
```

PGP decryption

Enable to decrypt files that were encrypted with PGP keys or verify signatures of files signed with PGP keys. You also can specify whether transfers fail if files are not encrypted and signed. The matching PGP keys for decryption and signature verification are detected at runtime. The keys must exist on SecureTransport or transfers fail. Only keys in non-delimited ASCII (ASC) file format are supported.

File filter

Select the method for filtering the files to process.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport documentation for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are compressed.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are compressed.

Decryption

Specifies when to decrypt files.

Always means all files are expected to be PGP-encrypted and will be decrypted. Transfers fail when SecureTransport receives or pulls unencrypted files.

Only if encrypted means only PGP-encrypted files are decrypted. Unencrypted files are processed normally.

Decryption key

Specifies the SecureTransport private PGP key for decrypting files from partners, applications, unmanaged products or other relays or picked up from an application via folder monitoring.

Once you provide a password, you can select the alias of an existing private PGP key or upload a new key. For a new key, upload the key in an ASC file and specify an alias. The user interface warns if you try to add a duplicate alias. All aliases are unique on SecureTransport.

Signature verification

Specifies when to verify signed files.

Always means all files are expected to be PGP-signed and will be verified. Transfers fail when SecureTransport receives or pulls files not signed with a trusted PGP key.

Only if signed means only PGP-signed files are verified. Unsigned files are processed normally.

Verification key

Specifies the public key for validating PGP signatures of files SecureTransport receives or pulls from a participant.

You can select the alias of an existing public PGP key or upload a new key. For a new key, upload the key in an ASC file and specify an alias. The user interface warns if you try to add a duplicate alias. Aliases are unique at the participant level.

PGP public keys used for signature verification belong to any of the following types of participants that send files or put files in SecureTransport: partners, applications, unmanaged products and products acting as relays.

A new PGP public key is linked to its owner when uploaded. You can use the same key in other flows where the key owner is involved.

Rename file as

Specifies a regular expression to name output files.

Example 1. New file name based on the current file name:

```
${basename(currentfulltarget)}.transformed
```

Example 2. New file name based on the original file name with a timestamp:

```
${basename(currentfulltarget)}.${timestamp}.${extension
(currentfulltarget)}
```

Example 3. New file name based on its name after the transformation (for example, name

of the file as extracted from an archive) by appending a timestamp:

```
${transformedfilename}.${timestamp}
```

Decompression

Enable to decompress ZIP, JAR, GZIP or TAR files. Then select a filter method and specify the name of the decompressed files.

File filter

When decompression is enabled, select the method for filtering the files to decompress.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport documentation for details. For example, if you specify ***\.(zip|tar)**, all ZIP and TAR files are decompressed.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.zip**, all ZIP files are decompressed.

Use archive password

Specifies whether a password is required to decompress the archive. Enter the password in the provided fields.

Rename file as

You can use a regular expression to specify the name of the decompressed file.

Compression

Enable to compress files as ZIP, JAR, GZIP or TAR files. Then select a filter method and specify the name of the compressed files.

File filter

When compression is enabled, select the method for filtering the files to compress.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport documentation for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are compressed.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are compressed.

Algorithm

Specifies the algorithm to use for compression.

Level

Specifies the speed of the compression. For ZIP, JAR and GZIP the following compression levels are available: Store (level 0), Fast (level 1), Normal (level 3), Good (level 5), Best (level 6). TAR supports only Store.

Use archive password

Specifies whether a password is required to decompress the archive. Enter the password in the provided field.

Archives

Specifies whether to compress files singly or together in one archive.

Archive name

You can use a regular expression to name the archive file. If files are archived in a single file, specify the archive name or expression. If files are archived singly and no value is specified in the Archive name field, each file is archived in a different file containing the file name and the algorithm extension.

Line ending

Enable to change line endings in files. Then select a filter method and specify the end of record characters in files.

File filter

Select the method for filtering the files.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport documentation for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are processed.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are processed.

Source and Target

Specify line ending characters and encoding for source and target files.

The available line-ending formats are CRLF, LF and custom.

If custom, specify the hex encoded value of the line ending character. This is any character \n, \r, and the combination of both. The custom line ending char in Unicode notation:

Windows: \u000d\u000a

*nix, MacOS: \u000a

Mainframe: \u0025

Select the file encoding format from the available formats in the drop-down list.

Rename file as

You can use a regular expression to name output files.

Example 1. New file name based on the current file name:

```
${basename(currentfulltarget)}.transformed
```

Example 2. New file name based on the original file name with a timestamp:

```
${basename(currentfulltarget)}.${timestamp}.${extension
(currentfulltarget)}
```

Example 3. New file name based on its name after the transformation (for example, name of the file as extracted from an archive) by appending a timestamp:

```
${transformedfilename}.${timestamp}
```

Character replacement

Enable to replace one or more sequences of characters in files. Then select a filter method and specify the encoding and transcoding of the files.

File filter

Select the method for filtering the files.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport documentation for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are processed.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are processed.

Find lines with

Specify a single or multiple sequences of characters to find in files, separated by commas.

Characters in ASCII and Unicode \uXXXX format are supported.

Delete found lines

Specify whether the lines with the found characters must be deleted in the file.

Yes means found lines are removed.

No means found lines are replaced with specific characters.

Replace with

Replace with one string or comma-separated strings in same respective order as find, in ASCII or Unicode \uXXXX format.

Example 1. Find is a,b,c. If replace is 123 and found text is axbxxc, after the file processing the text becomes 123x123xx123.

Example 2. Find is aaa,bbb,ccc. If replace is 123,456,789 and found text is aaaxbbbxxccc, after the file processing the text becomes 123x456xx789.

Source and Target encoding

Select the file encoding format from the available formats in the drop-down list for the source and target.

Rename file as

You can use a regular expression to name output files.

Example 1. New file name based on the current file name:

```
${basename(currentfulltarget)}.transformed
```

Example 2. New file name based on the original file name with a timestamp:

```
${basename(currentfulltarget)}.${timestamp}.${extension
(currentfulltarget)}
```

Example 3. New file name based on its name after the transformation (for example, name of the file as extracted from an archive) by appending a timestamp:

```
${transformedfilename}.${timestamp}
```

Line truncating

Enable truncating lines to a given line width. Then select a filter method and specify the encoding and transcoding of the files.

File filter

Select the method for filtering the files.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport documentation for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are processed.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are processed.

Truncate length

Specify the maximum length to which lines are truncated.

Source and Target encoding

Select the file encoding format from the available formats in the drop-down list for the source and target.

Rename file as

You can use a regular expression to name output files.

Example 1. New file name based on the current file name:

```
${basename(currentfulltarget)}.transformed
```

Example 2. New file name based on the original file name with a timestamp:

```
${basename(currentfulltarget)}.${timestamp}.${extension
(currentfulltarget)}
```

Example 3. New file name based on its name after the transformation (for example, name of the file as extracted from an archive) by appending a timestamp:

```
${transformedfilename}.${timestamp}
```

Line padding

Enable padding lines to a given line width. Then select a filter method and specify the encoding and transcoding of the files.

File filter

Select the method for filtering the files.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport documentation for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are processed.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are processed.

Line length

Specify the maximum length that lines must be padded to.

Padding character

Specify the Unicode character format \uXXXX used for padding lines.

Source and Target encoding

Select the file encoding format from the available formats in the drop-down list for the source and target.

Rename file as

You can use a regular expression to name output files.

Example 1. New file name based on the current file name:

```
${basename(currentfulltarget)}.transformed
```

Example 2. New file name based on the original file name with a timestamp:

```
${basename(currentfulltarget)}.${timestamp}.${extension
(currentfulltarget)}
```

Example 3. New file name based on its name after the transformation (for example, name of the file as extracted from an archive) by appending a timestamp:

```
${transformedfilename}.${timestamp}
```

Line folding

Enable wrapping lines to a given line width. Then select a filter method and specify the encoding and transcoding of the files.

File filter

Select the method for filtering the files.

Regular expression uses a regular expression for the filter. See regular expressions topics in the SecureTransport documentation for details. For example, if you specify ***\.(txt|xml)**, all TXT and XML files are processed.

File globbing uses wildcard characters to specify a pattern. For instance, **?** matches any single character and ***** matches any number of characters. For example, if you specify ***.xml**, all XML files are processed.

Fold width

Specify the maximum length to which lines are wrapped.

Source and Target encoding

Select the file encoding format from the available formats in the drop-down list for the source and target.

Rename file as

You can use a regular expression to name output files.

Example 1. New file name based on the current file name:

```
${basename(currentfulltarget)}.transformed
```

Example 2. New file name based on the original file name with a timestamp:

```
${basename(currentfulltarget)}.${timestamp}.${extension
(currentfulltarget)}
```

Example 3. New file name based on its name after the transformation (for example, name of the file as extracted from an archive) by appending a timestamp:

```
${transformedfilename}.${timestamp}
```

Transfer CFT fields in flows 24

The following topics describe the Transfer CFT fields in flows defined in Central Governance.

Source and target fields in flows control the behavior of transfers. You can define source or target properties when the source or target is applications or a group of applications associated with Transfer CFTs. Source and target properties have default values.

Transfer CFT source fields in flows

Source fields in flows control the behavior of a transfer. You can define source properties when the source is applications or a group of applications associated with Transfer CFTs. Source properties have default values.

The following topics are related to source properties for Transfer CFT:

- *Source transfer properties* on page 327
- *Source file properties* on page 332
- *Source processing scripts* on page 339

The following are the steps to get started.

1. Open the flow in edit mode, if not already opened.

   This might be a partially defined flow you saved previously or an unsaved flow still in the process of being defined.

   To open a flow for editing, click **Flows** on the top toolbar, click the name of a flow and then click **Edit**.
2. Add a source, if you haven't already done so.
3. Click the name of the source to display the menu of properties.
4. Click the type of property you want to change.

Source transfer properties

Default values are provided for source transfer properties in the user interface for Transfer CFT. Change values to meet your needs.

Prerequisites

- General flow information is defined
- Source or sources are selected

- The source properties page is displayed. See *Transfer CFT source fields in flows* on page 327 for the procedure to access this page.

Fields

Transfer priority

Transfer priorities are equivalent to integer values ranging from 0 (low) to 255 (high). If you select **Custom**, you must enter an integer between 0 and 255. Note that the integer value for medium priority is 128.

When Transfer CFT reaches the maximum number of transfers allowed, it queues transfers. When an ongoing transfer is finished and a slot is available for a new transfer, the system selects the one with the highest priority.

Transfer priority is available when the initiator is the source.

Bandwidth allocation

The amount of bandwidth allocated to this flow. The value you select determines the data transfer rate for this flow.

Transfer state

Indicates the state of the transfer request.

- Ready - Transfer is available and can start immediately.
- On hold - Transfer is deferred until a remote receive request is accepted, or until a local start command changes this transfer to the ready state.
- Kept - Transfer is deferred until a local start command changes this transfer to the ready state.

User ID

The identifier of the transfer owner.

Sending user ID

The identifier of the user sending the transfer.

Receiving user ID

The identifier of the user receiving the transfer.

Detect duplicate transfers

This field is used in detecting duplicate transfers and may contain a list of symbolic variables separated by a period ".". Duplicate transfers can occur, for example, if there is an error in a processing script. Possible variables include:

- &PART
- &IDF

- &DIRECT
- &MODE
- &SAPPL , &RAPPL
- &IDA
- &SUSER , &RUSER
- &FNAME, &NFNAME
- &SYSDATE, &SYSTIME
- &PARM

Example: &PART.&IDF.&IDA.&SAPPL

See *Symbolic variables* on page 291 for descriptions of the variables.

Compress file

Indicates whether files are compressed before transferred.

Do not enable compression if the type of files being transferred would not benefit from it and possibly result in longer processing times. For example, do not activate compression if transferring JPG or ZIP files.

On File not found

Specify what happens when files to transfer are not found.

- Abort transfer - Transfer fails and status is changed to Canceled.
- Ignore transfer - Transfer is successful and status is changed to Completed.

This parameter is available starting with Transfer CFT 3.1.3 Service Pack 4.

On file modification

Specify what happens if files are modified during the transfer.

- Continue transfer - Modified file is transferred.
- Stop transfer - Transfer status is changed to Kept.

Action after transfer

Specifies what happens to the file when the transfer has completed.

- Delete - Deletes the file.
- Delete file content - Removes the contents of the file but leaves the end-of-file mark at the beginning of the file.
- None - No action is performed on the file.

Delete file on purge

Indicates the transfer states of files that will be deleted when you remove the associated transfers from the transfer list or when you purge the transfer list. You can select any combination of statuses. If you do not select anything, files are not deleted even when the

associated transfers are removed from the transfer list.

- Ready (D) - Transfer is available and can start immediately.
- Transferring (C) - Transfer is being executed.
- On hold (H) - Transfer was interrupted due to an error, such as a network failure, or by a user.
- Kept (K) - Transfer was suspended by Transfer CFT or by a user.
- Transferred (T) - Transfer was completed successfully.
- Executed (X) - Transfer was ended by an application or user.

Purge completed transfer

Indicates whether a completed transfer is purged from the transfer list or kept.

- Yes - Transfer CFT purges the transfer from the transfer list when the status is Completed.
- No - Transfer CFT keeps completed transfers in the transfer list.

Additional information

Use this field for any information you want to provide. For example, you can use this field to propagate some business identifiers or values from the source to the target at the protocol level (without having to open the files). Note that in this use case, the attribute must be populated dynamically at runtime for each file transfer.

Maximum transfer duration

The maximum time in minutes to complete a file transfer before the transfer is canceled. If a transfer times out, you can restart the transfer manually. A transfer will not restart automatically. A value of **0** indicates there is no time limit.

This field is only available for the flow initiator. That is, it is only available for the source when the source is the initiator and it is only available for the target when the target is the initiator.

Activation period

If enabled, you can define the interval when transfers can occur by setting a start date and time and an end date and time.

This field is only available for the flow initiator. That is, it is only available for the source when the source is the initiator and it is only available for the target when the target is the initiator.

Visibility

On file modification

Specify the level of transfer process step details that are sent as events to the Visibility service.

- Default – Events are sent as defined in the configuration of the source Transfer CFT.
- All – Events related to every step of the transfer process are sent.
- First and last – Events related to only the initial and final steps of each transfer process are sent.
- None – No events are sent.

Broadcasting

A broadcast list represents the list of targets in the flow. Using a broadcast list enables to you send a file to all targets in the list with a single command. See *Transfer CFT broadcast and collect* on page 268 for more information.

Broadcast list

Enables or disables the use of a broadcast list. If you enable broadcast list (choose either File or Partner list) and have fewer than two targets defined, a warning message is displayed.

- File – Using a file in which the list of partners is saved. The file is automatically generated based on the selected targets.
- Partner list – Explicitly using a list type parameter. If there are more than 200 targets defined, a warning message is displayed specifying that the File option should be used instead. Transfer CFT partner list is limited to 200 targets.
- None – Disable the broadcast list.

When you enable broadcast list, the following fields are displayed depending on whether you select File or Partner list.

Name

Identifier for the list of targets defined in the flow. The name must be unique among all flows for the source or sources.

This field displays when you select either File or Partner list.

Filename

Name of the broadcast list file. The new file is available on Transfer CFT after the flow is deployed successfully. The new file is uploaded to Transfer CFT with the default path `$(cft.runtime_dir)/conf/ws_upload`. The file is overwritten if it already exists on Transfer CFT.

This field displays only when you select File.

Unknown target

Specifies the action to take when a target in the broadcast list is not found.

- Continue – Display an informational message and continue processing

- Ignore – Continue processing without an informational message
- Cancel – The transfer stops at the first error, but all transfers that were started before the error continue. For example, if you have 10 targets in the list and the fourth one is unknown, targets 1, 2 and 3 receive the file, but targets 4-10 do not.

This field displays when you select either File or Partner list.

Source file properties

Default values are provided for source file properties in the user interface for Transfer CFT. Change values to meet your needs.

Prerequisites

- General flow information is defined
- Source or sources are selected
- The source properties page is displayed. See *Transfer CFT source fields in flows* on page 327 for the procedure to access this page.

Fields

Filename

Files

Indicates whether a single file or multiple files are being sent.

Filename

If you selected **Single**, specify the path to the file where the files to be transferred are located.

The value you enter must represent a single file.

The file name can include the following symbolic variables:

- &FDATE, &FTIME, &FYEAR, &FMONTH, &FDAY
- &SPART, &RPART, &PART, &NPART, &GROUP
- &SUSER, &RUSER
- &SAPPL, &RAPPL
- &IDF, &PARM, &IDA
- &NIDF, &IDTU
- &BDATE, &BTIME, &BYEAR, &BMONTH, &BDAY

- &NFNAME, &NFVER

See *Symbolic variables* on page 291 for descriptions of the variables.

Path

If you selected **Multiple**, specify the path to the directory where the files to be transferred are located.

The value you enter can be:

- A directory name – All the files in this directory will be transferred.
- A generic file name, including wildcard characters – Only files that match are transferred. For example, `mydirectory/toto*`.

The maximum length of this field is 64 characters. Enclose the value in quotes to preserve case sensitivity.

This field is required if the target is the initiator of the flow.

File list

This field is displayed if you selected **Multiple** as the Files option.

Specify the name of the file that contains the list of files to be transferred. This file is also referred to as an indirection file. It must contain one file name per record, and that name must start in the first column of the file. The file names contained in the file must not contain an asterisk (*).

Archive name

Name of the file that contains the set of files to be transmitted. Archive files are sent between systems that have the same operating system (grouped mode). The archive file is created automatically by Transfer CFT at the time of the transfer. The file created is a ZIP file on Windows systems and a TAR file on Linux and UNIX systems. Because Windows systems do not have default compression utilities, Transfer CFT for Windows includes zip and unzip utilities.

The file name can include the following symbolic variables:

- &FDATE, &FTIME, &FYEAR, &FMONTH, &FDAY
- &SPART, &RPART, &PART, &NPART, &GROUP
- &SUSER, &RUSER
- &SAPPL, &RAPPL
- &IDF, &PARM, &IDA
- &NIDF, &NFNAME, &IDT
- &BDATE, &BTIME, &BYEAR, &BMONTH, &BDAY

See *Symbolic variables* on page 291 for descriptions of the variables.

Working directory

The path to the directory for sent files in process and temporary files. The working directory specifies a directory other than the default runtime directory for file transfer flows. All files related to the transfer flow — sent and temporary files and scripts — must be part of the working directory tree.

File name sent

Specify the name of the physical file that is to be used during transmission over the network.

If the file name is preceded by an asterisk (*), the target can keep the transmitted name or rename the file. The file name can include the following symbolic variables:

- &FDATE, &FTIME, &FYEAR, &FMONTH, &FDAY
- &SPART, &RPART, &PART, &NPART, &GROUP
- &SUSER, &RUSER
- &SAPPL, &RAPPL
- &IDF, &PARM, &IDA
- &NIDF
- &BDATE, &BTIME, &BYEAR, &BMONTH, &BDAY

See *Symbolic variables* on page 291 for descriptions of the variables.

The file is transferred providing the following conditions are met:

- The target authorizes the source (requester or server) to set the physical name of the file to be received as defined in the Path field.
- The file name specified in this field exists or can be created at the target.

File encoding

The file encoding fields vary depending on the Transfer CFT operating system.

Windows and Linux

File type

Specifies whether the file is a binary or text file.

Select **Binary** to send a mix of file types and ensure there is no custom encoding or transcoding required for the text files. If the text files have custom encoding or transcoding requirements, you must define specific flows for them.

The following fields are displayed for the text file type.

End of record character

Indicates the end of record character used in the file.

Ignore end of file character

This field is displayed only if you selected **CRLF** as the end of record character and if the source Transfer CFT is on a Windows system.

- No - Transfer CFT ends the transfer when it encounters an end-of-file character.
- Yes - Transfer CFT continues the transfer until there is no more data.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, enter the character set in the provided field.

Transcoding

Represents how the data in the file is encoded while it is being sent to the target.

If you select **Custom**, enter the character set in the provided field.

See *Transcoding and character translation* on page 337.

OS/400 (IBM i)

File type

The following describes the available options.

- **Data file** specifies the file is a PF-DTA file.
- **Save file** specifies the file is a SAVF file.
- **Source** specifies the file is a PF-SRC (with header) file.
- **OS 400 specific** specifies the file is a PF-SRC (no header) file.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, the Encoding charset field is displayed where you can enter the character set.

Transcoding

Represents how the data in the file is encoded while it is being sent to the target.

See *Transcoding and character translation* on page 337.

z/OS

File type

The following describes the available options.

- **Autodetect** specifies the file is sent in auto detection mode.

- **Print file with ASA jump codes** specifies the file is print file with ASA jump codes.
- **Print file with machine jump codes** specifies the file is print file with machine jump codes.
- **Spanned variable format** specifies the file is a spanned variable file.
- **ARDSSU** specifies the file is a ADRDSSU file.
- **Binary** specifies the file is a binary file.
- **Text** specifies the file is a text file.
- **Stream text** specifies the file is a text file sent in Stream CFT mode.

Select **Binary** to send a mix of file types, and make sure there is no custom encoding or transcoding required for the file types:

- Print file with ASA jump codes
- Print file with machine jump codes
- Text

If the text files have custom encoding or transcoding requirements, you must define specific flows for them.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, the Encoding charset field is displayed where you can enter the character set.

Transcoding

Represents how the data in the file is encoded while it is being sent to the target.

See *Transcoding and character translation* on page 337.

Record format

Record type

Indicates whether the records in the file are fixed or variable length.

Padding character

This field is displayed if you selected **Fixed** as the record type. Specify the character to use to pad the record. This character is added to the end of the record until it reaches the maximum length as defined in the Maximum record length field. If you do not provide a value, the default character is a space.

Trimming character

This field is displayed if you selected **Variable** as the record type. Specify the character to use to remove padding characters from the end of the record. For example, if the trimming character is a space and there are 5 spaces at the end of the record, all 5 spaces are

removed. If you do not provide a value, the record is unchanged.

Maximum record length

If you select **Default OS value**, Transfer CFT will interpret correctly maximum record length as:

- On Windows, 512 characters
- On Linux or UNIX, 512 characters for text files; 4096 characters for binary files

If you select **Custom**, enter a value in the provided field.

Transcoding and character translation

When defining source and target file properties, you specify how the data files are encoded on each end of the transfer. If the source and target require ASCII-encoded data, you specify ASCII encoding on both source and target file properties. In the source file properties, the Transcoding field defines how data are encoded during the transfer process. If **None** is set for transcoding, ASCII or EBCDIC is the value when the transfer is executed, depending on the partner's operating system. This is important when transferring files with different coding requirements on the source and target systems.

Custom encoding and transcoding

When you specify **Custom** in the Encoding or Transcoding fields, you must also enter the character set (charset) needed to convert the data. Custom values rely on the iconv program and API on UNIX/Linux platforms and on GNU iconv on Windows. This means that the charset value you enter must correspond to a valid iconv value for the host. This value is passed to both the source and target systems when the data flow is deployed.

Most of the character sets supported by Transfer CFT can be used for transcoding. These include UTF-8, UTF-16, UTF-32, UCS-2, ISO8859-1, ASCII, BIG5, cp850, and EBCDIC.

Example 1: Source and target have same encoding requirements

This example shows ASCII to ASCII transfers. For EBCDIC to EBCDIC transfers, enter EBCDIC in the Encoding and Transcoding fields. No character translation is required.

Source

Encoding – ASCII

Transcoding – ASCII

Target

Encoding – ASCII

Example 2: Source is ASCII and target is EBCDIC

This example shows ASCII to EBCDIC transfers.

Source

Encoding – ASCII

Transcoding – Select ASCII if you want the target to perform the translation from ASCII to EBCDIC, or select EBCDIC if you want the source to do it.

Target

Encoding – EBCDIC

Example 3: Custom: UTF-8 to UTF-32

The source sends a file encoded in UTF-8 and the target expects a file in UTF-32. Neither the source nor the target can directly translate from UTF-8 to UTF-32 because the source does not support UTF-32, and the target does not support UTF-8. However, both support UTF-16. Therefore, specify UTF-16 as the transcoding charset to have source convert the file to UTF-16 for the transfer. When the target receives the file, it converts it from UTF-16 to UTF-32.

Source

Encoding – Custom

Encoding charset – UTF-8

Transcoding – Custom

Transcoding charset – UTF-16

Target

Encoding – Custom

Encoding charset – UTF-32

Best practices

Use variable text files whenever possible because there is normally no shrinking or padding with this file type.

When dealing with multi-byte encoding on files with fixed or limited record size, be aware of the following:

- Shrinking a record may cause an unrecoverable error if it occurs in the middle of a multi-byte character.
- Record padding may cause an unrecoverable error if the number of characters to be padded is not a multiple of the pad character (by default, a blank for a text file and a zero for binary files).

- For streamed text files, either variable or fixed length, the CRLF characters used to separate records are from the encoding charset.

Errors

The following table describes some common transcoding errors and corrective actions.

Cause	Action
The charset is incorrect or it is not installed on your local system	Check your local configuration. If the charset is not found, add the charset to your system.
The file you are sending might contain an invalid character sequence or it cannot be transcoded to the charset destination.	Check the file and your configuration.
The file format is incorrect.	Verify the correct file format is defined.

Source processing scripts

The source processing scripts section of the flow definition identify the scripts to execute at specific phases in the flow lifecycle. See *Flow lifecycle* on page 233 for detailed information.

For each phase, you can select whether to execute the default script or a custom script. Default scripts, with the exception of pre-processing scripts, are defined in the configuration of the Transfer CFT involved in the flow. For example, SalesApp1 running on CFT001 is defined as the source in the flow. The default post-processing script defined in the CFT001 configuration is `exec/default_postprocessing.sh` (the actual default value is `exec/&IDF.sh`). If that is the script you want executed in the flow, you select the **Default** option.

To execute a different script, you must select the **Custom** option. Default scripts are defined in the Transfer processing section of the Transfer CFT configuration. See *Transfer processing* on page 175.

Processing scripts are executed on the source during the following phases:

- Pre-processing – Before the file is transferred
- Post-processing – After the file is transferred
- Acknowledgment – After an acknowledgment is sent by the target
- Error – After an error occurs during a transfer

For setting a custom processing script, you can:

- Set an existing script already on the source Transfer CFT.

  or

- Upload a new script. The new script is available on the Transfer CFT after the flow is deployed successfully on the Transfer CFT. The new script is uploaded to Transfer CFT with the default path `$(cft.runtime_dir)/conf/ws_upload`. If the file already exists on Transfer CFT, it is overwritten.

Source pre-processing scripts

A pre-processing script is executed before the file is transferred.

Prerequisites

- General flow information is defined
- Source or sources are selected
- The source properties page is displayed. See *Transfer CFT source fields in flows* on page 327 for the procedure to access this page.

Fields

Script

Indicates whether to execute a pre-processing script. There are no default pre-processing scripts.

File

If you select **Custom**, specify whether to use an existing file or upload a new file.

Filename

If you select **Custom** and select to use an existing file, specify the script to run. This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

State

Indicates the status of the transfer on the source. The script is run only if the transfer is in the specified state.

- Ready - Transfer is available and can start immediately.
- On hold - Transfer is deferred until a remote receive request is accepted, or until a local START command changes this transfer to the ready state.

Apply to broadcast list

This field is displayed if you enabled a broadcast list in source transfer properties.

- On main request – Executes the script only on the main request.
- For each target in the list – Executes the script only for each target in the list.
- Both – Executes the script both for the main request and for each target in the list.

Source post-processing scripts

A post-processing script is executed after the file is transferred.

Prerequisites

- General flow information is defined
- Source or sources are selected
- The source properties page is displayed. See *Transfer CFT source fields in flows* on page 327 for the procedure to access this page.

Fields

Script

Indicate whether to execute the default or a custom post-processing script. The default script is defined in the configuration of the source Transfer CFT in the flow.

File

If you select **Custom**, specify whether to use an existing file or upload a new file.

Filename

If you select **Custom** and select to use an existing file, specify the custom script to run. If the file does not exist, no post-processing is performed even if a default post-processing script is defined in the Transfer CFT configuration, and an error is raised.

This name can include the following symbolic variables:

- &IDF, &PARM

- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

Apply to group of files

This field is displayed if you selected **Multiple** for the **File** parameter in source file properties.

Specify the rule for executing the script on the group of files. When sending a group of files, there are two types of request: a generic request that applies to the entire group, and a specific request. There will be one specific request if the transfer is done in grouped mode, and multiple specific requests (one for each file) if the transfer is done in file-by-file mode.

- On main request – Execute the script only on the main request
- For each file in group – Execute the script for each file in the group but not for the main request
- Both – Execute the script both for the main request and for each file in the group

Apply to broadcast list

This field is displayed if you enabled a broadcast list in source transfer properties.

- On main request – Executes the script only on the main request.
- For each target in the list – Executes the script only for each target in the list.
- Both – Executes the script both for the main request and for each target in the list.

Source acknowledgment scripts

An acknowledgment script is executed after an acknowledgment is received for a sent file.

Prerequisites

- General flow information is defined
- Source or sources are selected

- The source properties page is displayed. See *Transfer CFT source fields in flows* on page 327 for the procedure to access this page.

Fields

Script

Indicates whether to execute the default or a custom acknowledgment script. The default script is defined in the configuration of the source Transfer CFT in the flow.

Filename

If you select **Custom** and select to use an existing file, specify the custom script to run. If the file does not exist, no processing is performed even if a default acknowledgment script is defined in the Transfer CFT configuration, and an error is raised.

This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

State

Indicates whether the transfer must wait for an acknowledgment.

- Require – Transfer must wait for an acknowledgment before it can be considered complete.
- Ignore – Transfer can be considered complete, even if an acknowledgment is not received.

Apply to group of files

This field is displayed if you selected **Multiple** for the **File** parameter in source file properties.

Specify the rule for executing the script on the group of files. When sending a group of files, there are two types of request: a generic request that applies to the entire group, and a specific request. There will be one specific request if the transfer is done in grouped mode, and multiple specific requests (one for each file) if the transfer is done in file-by-file mode.

- On main request – Execute the script only on the main request
- For each file in group – Execute the script for each file in the group but not for the main request
- Both – Execute the script both for the main request and for each file in the group

Apply to broadcast list

This field is displayed if you enabled a broadcast list in source transfer properties.

- On main request – Executes the script only on the main request.
- For each target in the list – Executes the script only for each target in the list.
- Both – Executes the script both for the main request and for each target in the list.

Source error scripts

An error script is executed after an error occurs during a transfer.

Prerequisites

- General flow information is defined
- Source or sources are selected
- The source properties page is displayed. See *Transfer CFT source fields in flows* on page 327 for the procedure to access this page.

Fields

Script

Indicates whether to execute the default or a custom error script. The default script is defined in the configuration of the source Transfer CFT in the flow, and an error is raised..

File

If you select **Custom**, specify whether to use an existing file or upload a new file.

Filename

If you select **Custom** and select to use an existing file, specifies the custom script to run. If the file does not exist, the default error script defined in the Transfer CFT configuration is executed.

This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP

- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

Transfer CFT target fields in flows

Target fields in flows control the behavior of a transfer. You can define target properties when the target is applications or a group of applications associated with Transfer CFTs. Target fields have default values.

The following topics are related to target property types for Transfer CFT:

- *Target transfer properties* on page 345
- *Target file properties* on page 350
- *Target processing scripts* on page 354

The following are the steps to get started.

1. Open the flow in edit mode, if not already opened.

   This might be a partially defined flow you saved previously or an unsaved flow still in the process of being defined.

   To open a flow for editing, click **Flows** on the top toolbar, click the name of a flow and then click **Edit**.
2. Add a target if you haven't already done so.
3. Click the name of the target to display the menu of properties.
4. Click the type of property you want to change.

Target transfer properties

Default values are provided for target transfer properties in the user interface for Transfer CFT. You can edit the defaults to meet your needs.

Prerequisites

- General flow information is defined
- Target or targets are selected

- The target properties page is displayed. See *Transfer CFT target fields in flows* on page 345 for the procedure to access this page.

Fields

Transfer priority

Transfer priorities are equivalent to integer values ranging from 0 (low) to 255 (high). If you select **Custom**, you must enter an integer between 0 and 255. Note that the integer value for medium priority is 128.

When Transfer CFT reaches the maximum number of transfers allowed, it queues transfers. When an ongoing transfer is finished and a slot is available for a new transfer, the system selects the one with the highest priority.

Transfer priority is available when the initiator is the target.

Bandwidth allocation

The amount of bandwidth allocated to this flow. The value you select determines the data transfer rate for this flow.

Transfer state

Indicates the state of the transfer request.

- Ready - Transfer is available and can start immediately.
- On hold - Transfer is deferred until a remote receive request is accepted, or until a local start command changes this transfer to the ready state.
- Kept - Transfer is deferred until a local start command changes this transfer to the ready state.

User ID

The identifier of the transfer owner.

Sending user ID

The identifier of the user sending the transfer.

Receiving user ID

The identifier of the user receiving the transfer.

Detect duplicate transfers

This field is used in detecting duplicate transfers and may contain a list of symbolic variables separated by a period ".". Duplicate transfers can occur, for example, if there is an error in a processing script. Possible variables include:

- &PART
- &IDF

- &DIRECT
- &MODE
- &SAPPL , &RAPPL
- &IDA
- &SUSER , &RUSER
- &FNAME, &NFNAME
- &SYSDATE, &SYSTIME
- &PARM

Example: &PART.&IDF.&IDA.&SAPPL

See *Symbolic variables* on page 291 for descriptions of the variables.

On File not found

Specify what happens when files to transfer are not found.

- Abort transfer - Transfer fails and status is changed to Canceled.
- Ignore transfer - Transfer is successful and status is changed to Completed.

This parameter is available starting with Transfer CFT 3.1.3 Service Pack 4.

No file exists

Specifies the action taken if the received file does not already exist.

- Create – File is created.
- Cancel – Transfer is refused.

Invalid options are disabled. Note that you cannot select **Cancel** in this field and in the File exists field. If that was possible, all transfers would fail.

File exists

Specifies the action taken if the received file exists.

- Delete – Existing file is deleted.
- Cancel – Transfer is refused.
- Overwrite – Existing file is overwritten.
- Overwrite only if empty – Existing file is overwritten only if it has no data.
- Overwrite after receiving temporary file – Existing file is overwritten after receiving the temporary file. This option is available only for Transfer CFT on Unix. The user who performs the transfer must have rights to rename the temporary file name into the file to overwrite.

Invalid options are disabled. You cannot select **Cancel** in this field and in the No file exists field. If that was possible, all transfers would fail.

Aborted transfer

Specifies the action taken if a transfer is terminated due to a file creation error on the target.

- Keep – Transfer remains in the transfer list.
- Delete – Transfer is removed from the transfer list.

Delete file on purge

Indicates the transfer states of files that will be deleted when you remove the associated transfers from the transfer list or when you purge the transfer list. You can select any combination of statuses. If you do not select anything, files are not deleted even when the associated transfers are removed from the transfer list.

- Ready (D) - Transfer is available and can start immediately.
- Transferring (C) - Transfer is being executed.
- On hold (H) - Transfer was interrupted due to an error, such as a network failure, or by a user.
- Kept (K) - Transfer was suspended by Transfer CFT or by a user.
- Transferred (T) - Transfer was completed successfully.
- Executed (X) - Transfer was ended by an application or user.

Purge completed transfer

Indicates whether a completed transfer is purged from the transfer list or kept.

- Yes - Transfer CFT purges the transfer from the transfer list when the status is Completed.
- No - Transfer CFT keeps completed transfers in the transfer list.

Maximum transfer duration

The maximum time in minutes to complete a file transfer before the transfer is canceled. If a transfer times out, you can restart the transfer manually. A transfer will not restart automatically. A value of **0** indicates there is no time limit.

This field is only available for the flow initiator. That is, it is only available for the source when the source is the initiator and it is only available for the target when the target is the initiator.

Activation period

If enabled, you can define the interval when transfers can occur by setting a start date and time and an end date and time.

This field is only available for the flow initiator. That is, it is only available for the source when the source is the initiator and it is only available for the target when the target is the initiator.

Visibility

On file modification

Specify the level of transfer process step details that are sent as events to the Visibility service.

- Default – Events are sent as defined in the configuration of the target Transfer CFT.
- All – Events related to every step of the transfer process are sent.
- First and last – Events related to only the initial and final steps of each transfer process are sent.
- None – No events are sent.

Collecting

A collect list represents a list of sources. Using a collect list enables to you receive a file from all sources in the list with a single command. See *Transfer CFT broadcast and collect* on page 268 for more information.

Collect list

Enables or disables the use of a collect list. If you enable the collect list (chose either File or Partner list) and have fewer than two sources defined, a warning message is displayed.

- File – Using a file in which the list of partners is saved. The file is automatically generated based on the selected sources.
- Partner list – Explicitly using a list type parameter. If there are more than 200 sources defined, a warning message is displayed specifying that the File option should be used instead. Transfer CFT partner list is limited to 200 sources.
- None – Disable the collect list.

When you enable collect list, the following fields are displayed depending on whether you select File or Partner list.

Name

Identifier for the list of sources defined in the flow. The name must be unique among all flows for the target or targets.

This field displays when you select either File or Partner list.

Filename

Name of the collect list file. The new file is available on Transfer CFT after the flow is deployed successfully. The new file is uploaded to Transfer CFT with the default path `$(cft.runtime_dir)/conf/ws_upload`. The file is overwritten if it already exists on Transfer CFT.

This field displays only when you select File.

Unknown source

Specifies the action to take when a source in the collect list is not found.

- Continue – Display an informational message and continue processing
- Ignore – Continue processing without an informational message
- Cancel – Transfer stops at the first error, but all transfers that were started before the error continue. For example, if you have 10 sources in the list and the fourth one is unknown, the file is received from sources 1-3, but not from sources 4-10.

This field displays when you select either File or Partner list.

Target file properties

Default values are provided for target file properties in the user interface for Transfer CFT. Modify the defaults to meet your needs.

Prerequisites

- General flow information is defined
- Target or targets are selected
- The target properties page is displayed. See *Transfer CFT target fields in flows* on page 345 for the procedure to access this page.

Fields

Filename

Specify the file name or full path name for the received file or files. This field is required if the initiator of the flow is the source. Default value: pub\&IDF.&IDTU.&FROOT.RCV

The file name can include the following symbolic variables:

- &FDATE, &FTIME, &FYEAR, &FMONTH, &FDAY
- &SPART, &RPART, &PART, &IPART, &NPART, &GROUP
- &SUSER, &RUSER
- &SAPPL, &RAPPL
- &IDF, &PARM, &IDA
- &NIDF, &IDTU, &IDT
- &BDATE, &BTIME, &BYEAR, &BMONTH, &BDAY
- &NFNAME
- &NFVER

See *Symbolic variables* on page 291 for descriptions of the variables.

Working directory

The path to the directory for received files in process and temporary files. The working directory specifies a directory other than the default runtime directory for file transfer flows. All files related to the transfer flow — received and temporary files and scripts — must be part of the working directory tree.

Temporary file

Specify the name of the temporary file used during the transfer. When the transfer is complete, the temporary file is renamed using the name defined in the Filename field. If you do not specify a value, Transfer CFT creates the file with the name specified in the Filename field. If the File exists parameter is set to Overwrite after receiving temporary file, Temporary file becomes required.

The file name can include the following symbolic variables:

- &FDATE, &FTIME, &FYEAR, &FMONTH, &FDAY
- &SPART, &RPART, &PART, &NPART, &GROUP
- &SUSER, &RUSER
- &SAPPL, &RAPPL
- &IDF, &PARM, &IDA
- &NIDF, &NFNAME, &IDT
- &BDATE, &BTIME, &BYEAR, &BMONTH, &BDAY

See *Symbolic variables* on page 291 for descriptions of the variables.

Receiving file size

The file size in kilobytes allocated when receiving the file. When the value is **0**, the file size is specified by the sender. The parameter is available only for Transfer CFT on z/OS computers.

File encoding

The file encoding fields vary depending on the Transfer CFT operating system.

Windows and Linux

File type

Specify whether the file is a binary or text file.

The following fields are displayed for the text file type only.

End of record character

Indicates the end of record character used in the file.

Ignore end of file character

This field is displayed only if you selected **CRLF** as the end-of-record character and if one of the source Transfer CFTs is on Windows.

- No - Transfer CFT ends the transfer when it encounters an end-of-file character.
- Yes - Transfer CFT continues the transfer until there is no more data.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, enter the character set in the provided field.

OS/400 (IBM i)

File type

The following describes the available options.

- **Data file** specifies the file is a PF-DTA file.
- **Save file** specifies the file is a SAVF file.
- **Source** specifies the file is a PF-SRC (with header) file.
- **OS 400 specific** specifies the file is a PF-SRC (no header) file.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, the Encoding charset field is displayed where you can enter the character set.

z/OS

File type

The following describes the available options.

- **Autodetect** specifies the file is sent in auto detection mode.
- **Print file with ASA jump codes** specifies the file is print file with ASA jump codes.
- **Print file with machine jump codes** specifies the file is print file with machine jump codes.
- **Spanned variable format** specifies the file is a spanned variable file.
- **ARDSSU** specifies the file is a ADRDSSU file.
- **Binary** specifies the file is a binary file.
- **Text** specifies the file is a text file.
- **Stream text** specifies the file is a text file sent in Stream CFT mode.

- **PDSE** specifies the file is a PDSE file.

Select **Binary** to send a mix of file types, and make sure there is no custom encoding or transcoding required for the file types:

- Print file with ASA jump codes
- Print file with machine jump codes
- Text

If the text files have custom encoding or transcoding requirements, you must define specific flows for them.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, the Encoding charset field is displayed where you can enter the character set.

Record format

Record type

Indicates whether the records in the file are fixed or variable length.

Padding character

This field is displayed if you selected **Fixed** as the record type. Specify the character to use to pad the record. This character is added to the end of the record until it reaches the maximum length as defined in the Maximum record length field. If you do not provide a value, the default character is a space.

Trimming character

This field is displayed if you selected **Variable** as the record type. Specify the character to use to remove padding characters from the end of the record. For example, if the trimming character is a space and there are 5 spaces at the end of the record, all 5 spaces are removed. If you do not provide a value, the record is unchanged.

Maximum record length

If you select **Default OS value**, Transfer CFT will interpret correctly maximum record length as:

- On Windows, 512 characters
- On Linux or UNIX, 512 characters for text files; 4096 characters for binary files

If you select **Custom**, enter a value in the provided field.

Target processing scripts

The target processing scripts section of the flow definition identify the files to execute at specific phases in the flow lifecycle. See *Flow lifecycle* on page 233 for detailed information.

For each phase, you can select whether to execute the default script or a custom script. Default scripts are defined in the configuration of the Transfer CFT involved in the flow. For example, ProdCat1 running on CFT100 is defined as the target in the flow. The default post-processing script defined in the CFT100 configuration is `exec/default_postprocessing.sh` (the actual default value is `exec/&IDF.sh`). If that is the script you want executed in the flow, you select the **Default** option.

To execute a different script, you must select the **Custom** option. Default scripts are defined in the Transfer processing section of the Transfer CFT configuration. See *Transfer processing* on page 175.

Processing scripts are executed on the target during the following phases:

- Post-processing – Executed after the file was received
- Acknowledgment – Executed after the file is transferred
- Error – Executed if an error occurs when a file is received

For setting a custom processing script, you can:

- Set an existing script already on the target Transfer CFT.

  or

- Upload a new script. The new script is available on the Transfer CFT after the flow is deployed successfully on the Transfer CFT. The new script is uploaded to Transfer CFT with the default path `$(cft.runtime_dir)/conf/ws_upload`. If the file already exists on Transfer CFT, it is overwritten.

Target post-processing scripts

A target post-processing script is executed after a file is received.

Prerequisites

- General flow information is defined
- Target or targets are selected
- The target properties page is displayed. See *Transfer CFT target fields in flows* on page 345 for the procedure to access this page.

Fields

Script

Indicate whether to execute the default or a custom post-processing script. The default script is defined in the configuration of the target Transfer CFT in the flow.

File

If you select **Custom**, specify whether to use an existing file or upload a new file.

Filename

If you select **Custom** and select to use an existing file, specify the custom script to run. If the file does not exist, no post-processing is performed even if a default post-processing script is defined in the Transfer CFT configuration, and an error is raised.

This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

Apply to collect list

Apply to collect list

This field is displayed if you enabled a collect list in target transfer properties.

- On main request – Executes the script only on the main request.
- For each source in the list – Executes the script only for each source in the list.
- Both – Executes the script both for the main request and for each source in the list.

Target acknowledgment scripts

A target acknowledgment script is executed after the file is received and post-processing is complete.

Prerequisites

- General flow information is defined
- Target or targets are selected
- The target properties page is displayed. See *Transfer CFT target fields in flows* on page 345 for the procedure to access this page.

Fields

Script

Indicate whether to execute a custom acknowledgment script.

Filename

If you select **Custom** and select to use an existing file, specify the custom script to run. If the file does not exist, no processing is performed even if a default acknowledgment script is defined in the Transfer CFT configuration, and an error is raised.

This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

State

Indicates whether the transfer must wait for an acknowledgment.

- Require – Transfer must wait for an acknowledgment before it can be considered complete.
- Ignore – Transfer can be considered complete, even if an acknowledgment is not received.

Apply to collect list

This field is displayed if you enabled a collect list in target transfer properties.

- On main request – Executes the script only on the main request.

- For each source in the list – Executes the script only for each source in the list.
- Both – Executes the script both for the main request and for each source in the list.

Target error scripts

A target error script is executed if an error occurs when a file is received.

Prerequisites

- General flow information is defined
- Target or targets are selected
- The target properties page is displayed. See *Transfer CFT target fields in flows* on page 345 for the procedure to access this page.

Fields

Script

Indicate whether to execute the default or a custom error script. The default script is defined in the configuration of the target Transfer CFT in the flow.

File

Filename

If you select **Custom** and select to use an existing file, specify the custom script to run. If the file does not exist, no processing is performed even if a default error script is defined in the Transfer CFT configuration, and an error is raised.

This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

Transfer CFT legacy flows 25

Legacy flows for Transfer CFTs are a feature for enabling long-time users of Transfer CFT to transition flow management to Central Governance. Legacy flows address the following use cases:

- Flows can be managed in Central Governance for individual Transfer CFTs. In the Central Governance user interface, users can manage partners and send and receive templates for a specific Transfer CFT.
- Users can employ an established procedure to migrate Transfer CFT flow definitions to the Central Governance flow-management process.

Central Governance validates the uniqueness of legacy flow objects against Central Governance flows when Central Governance flows are deployed. Objects must be unique across all types of flows.

Legacy flows are added to the Central Governance user interface when Transfer CFTs register with Central Governance. However, Central Governance does not support some values in legacy flows. If Central Governance cannot map some legacy flow values to available values in legacy flows objects, the values are replaced with default values.

In Central Governance, the objects available in the legacy flows user interface are not part of the flows user interface.

Corresponding Transfer CFT objects

The following table describes how legacy flows in Central Governance correspond to objects in Transfer CFT.

Central Governance object	Transfer CFT object	Description
Partner	CFTPART, CFTTCP	A transfer partner.
Distribution list	CFTDEST	A pseudo-partner referencing a list of partners for performing broadcasting and collecting in a single command.
Send template	CFTSEND	Specifies default values for sending data.
Receive template	CFTRECV	Specifies default values for receiving data.

Flow migration example

In the following example of migrating a legacy flow to a flow in Central Governance, there are two Transfer CFTs, identified as CFT-1 and CFT-2. Both begin as version 2.7.1 and are not registered in Central Governance.

1. CFT-1 and CFT-2 are upgraded to Transfer CFT 3.1.3.
2. CFT-1 and CFT-2 are registered in Central Governance.
3. The following legacy flows are in Central Governance for each Transfer CFT:
    - CFT-1:
        - Send template: FL001
        - Partner: CFT-2
    - CFT-2:
        - Receive template: FL001
        - Partner: CFT-1
4. In Central Governance the user adds a flow with the identifier FL001. The user selects CFT-1 as the source and CFT-2 as the target. The user defines the flow properties and protocol.
5. When the user deploys FL001 Central Governance determines that:
    - FL001 already is defined on CFT-1 and CFT-2.
    - Partners CFT-2 and CFT-1 exist on the Transfer CFTs involved in the flow.
6. Central Governance prompts the user to confirm whether to update the flow on the Transfer CFTs. If the user confirms the FL001 deployment, the send and receive templates for CFT-1 and CFT-2 are no longer available in the legacy flows user interface in Central Governance.

Legacy flow partner objects can conflict with the flow configuration if any of the legacy flow fields have the same value as the corresponding flow field.

Central Governance legacy flow field	Central Governance flow field	CFTUTIL parameter
Name	Transfer CFT name	CFTPART.ID CFTTCP.ID
Partner Access – Remote partner name	Transfer CFT name	CFTPART.NRPART

Partner objects CFTPART and CFTTCP are updated with the information defined in the flow. Alternatively, you can choose to keep the values from the legacy flow partner object by setting the property `migrate.legacy.part.to.flows=true` in the `com.axway.cmp.cgcft-default.cfg` configuration file at `<install directory>/runtime/com.axway.nodes.ume.<UUID>/conf`. All parameters except those in the following table keep the value from the legacy flow object.

Central Governance legacy flow field	Central Governance field which replaces the legacy flow value	CFTUTIL parameter
Partner Access – Host	Host of the partner Transfer CFT.	CFTTCP.HOST
Protocol – Protocol	Protocol name defined on the current Transfer CFT for the flow protocol options: network protocol/security.	CFTPART.CFTPROTID
Protocol – Port in	Port defined in the partner Transfer CFT for the flow protocol options: network protocol/security.	CFTPART.SAP

Migration of flows is completed when there are no more legacy flows objects defined on the Transfer CFTs in Central Governance.

Legacy flows lifecycle

The changing statuses in the lifecycle of legacy flows are displayed for Transfer CFTs in the user interface. The statuses are described in the following tables.

Global statuses

The following global statuses are displayed at the top of the Legacy Flows page for each Transfer CFT.

The global status is for all legacy flows for a specific Transfer CFT. Legacy flows are comprised of objects that include send and receive templates, partners and distribution lists.

If at least one object has a status of	The displayed global status is
Saved, not deployed	Saved, not deployed
Deploying	Deploying
Deployed with errors	Deployed with errors
Removing	Removing
Removed with errors	Removed with errors

If no legacy flow objects meet these conditions, the displayed global status is Deployed.

Object statuses

The following are the statuses for each object in a legacy flow. The objects are send and receive templates, partners and distribution lists.

Status	Status messages	Description
Saved not deployed	Saved at <time> today, not deployed Saved on <date>, not deployed	An object has been added or changed, but has not been deployed.
Deploying	Deploying since <time> today Deploying since <date>	An object is being deployed.
Deployed with errors	Deployed at <time> today Deployed on <date>	An object has been deployed, but errors occurred.
Deployed	Deployed at <time> today Deployed on <date>	An object has been deployed successfully.
Removing	Removing since <time> today Removing since <date>	An object is being removed from Central Governance and Transfer CFT.
Removed with errors	Removed at <time> today Removed on <date>	An object has been removed, but errors occurred.

Deploy and remove actions

The following table indicates when the status of a legacy flow object enables you to perform deploy and remove actions.

Status	Deploy	Remove
Saved not deployed	Yes	Yes
Deploying	No	No
Deployed with errors	Yes	Yes

Status	Deploy	Remove
Deployed	Yes, but only after you change the object	Yes
Removing	No	No
Removed with errors	No	Yes

Manage templates

Use the following procedures to manage send and receive templates. This includes listing, adding, viewing, deploying and removing templates.

Note You can define templates on Transfer CFT directly or by defining flows in Central Governance.

List templates

The following are the steps to navigate to send or receive templates on the Transfer CFT Legacy Flows page for a selected Transfer CFT.

1. Click **Products** on the top toolbar to open the Product List page.
2. Find the Transfer CFT you want and click its name to open its details page.
3. Click **Legacy flows** to open the Transfer CFT Legacy Flows page for the selected Transfer CFT.
4. Select **Send templates** or **Receive templates** to list the names, statuses, file names and descriptions of the templates for the Transfer CFT.

Add template

The starting point for the following steps is when send or receive templates for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

1. Click **Add send template** or **Add receive template** to open the add template page for the selected Transfer CFT.
2. Complete the fields. See *Send template fields* on page 363 or *Receive template fields* on page 375 for information.
3. Click **Save send template** or **Save receive template** to add the template.

Deploy template

The starting point for the following steps is when send or receive templates for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

1. Select a template with a status of Saved, not deployed.
2. Click **Deploy** to deploy the template to the Transfer CFT.

View, edit template

The starting point for the following steps is when send or receive templates for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

1. Click the name of a template to open its details page.
2. Click **Edit** to change the template. See *Send template fields* on page 363 or *Receive template fields* on page 375 for information.
3. Click **Save changes** to save changes to the template.

Remove template

The starting point for the following steps is when send or receive templates for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

Do one of the following to remove a template from the selected Transfer CFT:

- Select a template and click **Remove**.
- Click the name of a template to open its details page. Click **Remove**.

If the template you remove is deployed on Transfer CFT, the template also is removed from it.

Send template fields

The following are the fields to use when adding or editing a send template for Transfer CFT in the Central Governance user interface. See *Manage templates* on page 362 for actions you can perform.

A send template defines:

- The name and local physical characteristics of the file to send.
- The network characteristics of the file to send to a partner.
- The actions to perform locally during and after a transfer. For example, translation, compression, call to a user exit, an end-of-transfer procedure, and so on.
- A default user associated with the transfers

Note You can define templates on Transfer CFT directly or by defining flows in Central Governance.

Name

The template name. The name must be unique across all CFTSEND objects defined on the Transfer CFT.

There cannot be a Central Governance flow with the same identifier that uses the Transfer CFT as a source.

Initiator

The party that initiates the transfer.

- Source - Source system initiates the transfer.
- Target - Target system makes a request to initiate the transfer.

Description

A description of the template

Transfer properties

Transfer priority

Transfer priorities are equivalent to integer values ranging from 0 (low) to 255 (high). If you select **Custom**, you must enter an integer between 0 and 255. Note that the integer value for medium priority is 128.

When Transfer CFT reaches the maximum number of transfers allowed, it queues transfers. When an ongoing transfer is finished and a slot is available for a new transfer, the system selects the one with the highest priority.

Bandwidth allocation

The amount of bandwidth allocated to this flow. The value you select determines the data transfer rate for this flow.

Transfer state

Indicates the state of the transfer request.

- Ready - Transfer is available and can start immediately.
- On hold - Transfer is deferred until a remote receive request is accepted, or until a local start command changes this transfer to the ready state.
- Kept - Transfer is deferred until a local start command changes this transfer to the ready state.

User ID

The identifier of the transfer owner.

Detect duplicate transfers

This field is used in detecting duplicate transfers and may contain a list of symbolic variables separated by a period ".". Duplicate transfers can occur, for example, if there is an error in a processing script. Possible variables include:

- &PART

- &IDF
- &DIRECT
- &MODE
- &SAPPL , &RAPPL
- &IDA
- &SUSER , &RUSER
- &FNAME, &NFNAME
- &SYSDATE, &SYSTIME
- &PARM

Example: &PART.&IDF.&IDA.&SAPPL

See *Symbolic variables* on page 291 for descriptions of the variables.

Compress file

Indicates whether files are compressed before transferred.

Do not enable compression if the type of files being transferred would not benefit from it and possibly result in longer processing times. For example, do not activate compression if transferring JPG or ZIP files.

On file modification

Specify what happens if files are modified during the transfer.

- Continue transfer - Modified file is transferred.
- Stop transfer - Transfer status is changed to Kept.

Action after transfer

Specifies what happens to the file when the transfer has completed.

- Delete - Deletes the file.
- Delete file content - Removes the contents of the file but leaves the end-of-file mark at the beginning of the file.
- None - No action is performed on the file.

Delete file on purge

Indicates the transfer states of files that will be deleted when you remove the associated transfers from the transfer list or when you purge the transfer list. You can select any combination of statuses. If you do not select anything, files are not deleted even when the associated transfers are removed from the transfer list.

- Ready (D) - Transfer is available and can start immediately.
- Transferring (C) - Transfer is being executed.
- On hold (H) - Transfer was interrupted due to an error, such as a network failure, or by a user.

- Kept (K) - Transfer was suspended by Transfer CFT or by a user.
- Transferred (T) - Transfer was completed successfully.
- Executed (X) - Transfer was ended by an application or user.

Additional information

Use this field for any information you want to provide. For example, you can use this field to propagate some business identifiers or values from the source to the target at the protocol level (without having to open the files). Note that in this use case, the attribute must be populated dynamically at runtime for each file transfer.

File properties > filename

Files

Indicates whether a single file or multiple files are being sent.

Filename

Specify the path to the file or to the directory where the files to be transferred are located.

If you selected **Single**, the value you enter must represent a single file.

The file name can include the following symbolic variables:

- &FDATE, &FTIME, &FYEAR, &FMONTH, &FDAY
- &SPART, &RPART, &PART, &NPART, &GROUP
- &SUSER, &RUSER
- &SAPPL, &RAPPL
- &IDF, &PARM, &IDA
- &NIDF, &IDTU
- &BDATE, &BTIME, &BYEAR, &BMONTH, &BDAY
- &NFNAME, &NFVER

See *Symbolic variables* on page 291 for descriptions of the variables.

If you selected **Multiple**, the value you enter can be:

- A directory name – All the files in this directory will be transferred.
- A generic file name, including wildcard characters – Only files that match are transferred. For example, `mydirectory/toto*`.

The maximum length of this field is 64 characters. Enclose the value in quotes to preserve case sensitivity.

This field is required if the target is the initiator of the flow.

File list

This field is displayed if you selected **Multiple** as the Files option.

Specify the name of the file that contains the list of files to be transferred. This file is also referred to as an indirection file. It must contain one file name per record, and that name must start in the first column of the file. The file names contained in the file must not contain an asterisk (*).

Archive name

Name of the file that contains the set of files to be transmitted. Archive files are sent between systems that have the same operating system (grouped mode). The archive file is created automatically by Transfer CFT at the time of the transfer. The file created is a ZIP file on Windows systems and a TAR file on Linux and UNIX systems. Because Windows systems do not have default compression utilities, Transfer CFT for Windows includes zip and unzip utilities.

The file name can include the following symbolic variables:

- &FDATE, &FTIME, &FYEAR, &FMONTH, &FDAY
- &SPART, &RPART, &PART, &NPART, &GROUP
- &SUSER, &RUSER
- &SAPPL, &RAPPL
- &IDF, &PARM, &IDA
- &NIDF, &NFNAME, &IDT
- &BDATE, &BTIME, &BYEAR, &BMONTH, &BDAY

See *Symbolic variables* on page 291 for descriptions of the variables.

File name sent

Specify the name of the physical file that is to be used during transmission over the network.

If the file name is preceded by an asterisk (*), the target can keep the transmitted name or rename the file. The file name can include the following symbolic variables:

- &FDATE, &FTIME, &FYEAR, &FMONTH, &FDAY
- &SPART, &RPART, &PART, &NPART, &GROUP
- &SUSER, &RUSER
- &SAPPL, &RAPPL
- &IDF, &PARM, &IDA
- &NIDF
- &BDATE, &BTIME, &BYEAR, &BMONTH, &BDAY

See *Symbolic variables* on page 291 for descriptions of the variables.

The file is transferred providing the following conditions are met:

- The target authorizes the source (requester or server) to set the physical name of the file to be received as defined in the Path field.

- The file name specified in this field exists or can be created at the target.

File properties > file encoding

The file encoding fields vary depending on the Transfer CFT operating system.

Windows and Linux

File type

Specifies whether the file is a binary or text file.

Select **Binary** to send a mix of file types and ensure there is no custom encoding or transcoding required for the text files. If the text files have custom encoding or transcoding requirements, you must define specific flows for them.

The **Stream text** option is available only in legacy flows and not Central Governance flows.

Text file type

The following fields are displayed for the text file type.

End of record character

Indicates the end of record character used in the file.

Ignore end of file character

This field is displayed only if you selected **CRLF** as the end of record character and if the source Transfer CFT is on a Windows system.

- No - Transfer CFT ends the transfer when it encounters an end-of-file character.
- Yes - Transfer CFT continues the transfer until there is no more data.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, enter the character set in the provided field.

Transcoding

Represents how the data in the file is encoded while it is being sent to the target.

If you select **Custom**, enter the character set in the provided field.

See *Transcoding and character translation* on page 337.

Stream text file type

The following fields are displayed for the stream text file type.

Encoding

Represents how the data in the file to be sent is encoded.

Transcoding

Represents how the data in the file is encoded while it is being sent to the target.

OS/400 (IBM i)

File type

The following describes the available options.

- **Data file** specifies the file is a PF-DTA file.
- **Save file** specifies the file is a SAVF file.
- **Source** specifies the file is a PF-SRC (with header) file.
- **OS 400 specific** specifies the file is a PF-SRC (no header) file.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, the Encoding charset field is displayed where you can enter the character set.

Transcoding

Represents how the data in the file is encoded while it is being sent to the target.

If you select **Custom**, the Transcoding charset field is displayed where you can enter the character set.

See *Transcoding and character translation* on page 337.

z/OS

File type

The following describes the available options.

- **Autodetect** specifies the file is sent in auto detection mode.
- **Print file with ASA jump codes** specifies the file is print file with ASA jump codes.
- **Print file with machine jump codes** specifies the file is print file with machine jump codes.
- **Spanned variable format** specifies the file is a spanned variable file.
- **ARDSSU** specifies the file is a ADRDSSU file.
- **Binary** specifies the file is a binary file.

- **Text** specifies the file is a text file.
- **Stream text** specifies the file is a text file sent in Stream CFT mode.

Select **Binary** to send a mix of file types, and make sure there is no custom encoding or transcoding required for the file types:

- Print file with ASA jump codes
- Print file with machine jump codes
- Text

If the text files have custom encoding or transcoding requirements, you must define specific flows for them.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, the Encoding charset field is displayed where you can enter the character set.

Transcoding

Represents how the data in the file is encoded while it is being sent to the target.

If you select **Custom**, the Transcoding charset field is displayed where you can enter the character set.

See *Transcoding and character translation* on page 337.

File properties > record format

Record type

Indicates whether the records in the file are fixed or variable length.

Padding character

This field is displayed if you selected **Fixed** as the record type. Specify the character to use to pad the record. This character is added to the end of the record until it reaches the maximum length as defined in the Maximum record length field. If you do not provide a value, the default character is a space.

Trimming character

This field is displayed if you selected **Variable** as the record type. Specify the character to use to remove padding characters from the end of the record. For example, if the trimming character is a space and there are 5 spaces at the end of the record, all 5 spaces are removed. If you do not provide a value, the record is unchanged.

Maximum record length

If you select **Default OS value**, Transfer CFT will interpret correctly maximum record length as:

- On Windows, 512 characters
- On Linux or UNIX, 512 characters for text files; 4096 characters for binary files

If you select **Custom**, enter a value in the provided field.

Processing scripts

The source processing scripts section of the flow definition identify the scripts to execute at specific phases in the flow lifecycle. See *Flow lifecycle* on page 233 for detailed information.

For each phase, you can select whether to execute the default script or a custom script. Default scripts, with the exception of pre-processing scripts, are defined in the configuration of the Transfer CFT involved in the flow. For example, SalesApp1 running on CFT001 is defined as the source in the flow. The default post-processing script defined in the CFT001 configuration is `exec/default_postprocessing.sh` (the actual default value is `exec/&IDF.sh`). If that is the script you want executed in the flow, you select the **Default** option.

To execute a different script, you must select the **Custom** option. Default scripts are defined in the Transfer processing section of the Transfer CFT configuration. See *Transfer processing* on page 175.

Processing scripts are executed on the source during the following phases:

- Pre-processing – Before the file is transferred
- Post-processing – After the file is transferred
- Acknowledgment – After an acknowledgment is sent by the target
- Error – After an error occurs during a transfer

For setting a custom processing script, you can:

- Set an existing script already on the source Transfer CFT.

  or

- Upload a new script. The new script is available on the Transfer CFT after the flow is deployed successfully on the Transfer CFT. The new script is uploaded to Transfer CFT with the default path `$(cft.runtime_dir)/conf/ws_upload`. If the file already exists on Transfer CFT, it is overwritten.

Pre-processing

Script

Indicates whether to execute a pre-processing script. There are no default pre-processing scripts.

File

If you select **Custom**, specify whether to use an existing file or upload a new file.

Filename

If you select **Custom** and select to use an existing file, specify the script to run. This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

State

Indicates the status of the transfer on the source. The script is run only if the transfer is in the specified state.

- Ready - Transfer is available and can start immediately.
- On hold - Transfer is deferred until a remote receive request is accepted, or until a local START command changes this transfer to the ready state.

Post-processing

Script

Indicate whether to execute the default or a custom post-processing script. The default script is defined in the configuration of the source Transfer CFT in the flow.

File

If you select **Custom**, specify whether to use an existing file or upload a new file.

Filename

If you select **Custom** and select to use an existing file, specify the custom script to run. If the file does not exist, no post-processing is performed even if a default post-processing script is defined in the Transfer CFT configuration, and an error is raised.

This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID

- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

Apply to group of files

This field is displayed if you selected **Multiple** for the **File** parameter in source file properties.

Specify the rule for executing the script on the group of files. When sending a group of files, there are two types of request: a generic request that applies to the entire group, and a specific request. There will be one specific request if the transfer is done in grouped mode, and multiple specific requests (one for each file) if the transfer is done in file-by-file mode.

- On main request – Execute the script only on the main request
- For each file in group – Execute the script for each file in the group but not for the main request
- Both – Execute the script both for the main request and for each file in the group

Acknowledgment

Script

Indicates whether to execute the default or a custom acknowledgment script. The default script is defined in the configuration of the source Transfer CFT in the flow.

Filename

If you select **Custom** and select to use an existing file, specify the custom script to run. If the file does not exist, no processing is performed even if a default acknowledgment script is defined in the Transfer CFT configuration, and an error is raised.

This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

State

Indicates whether the transfer must wait for an acknowledgment.

- Require – Transfer must wait for an acknowledgment before it can be considered complete.
- Ignore – Transfer can be considered complete, even if an acknowledgment is not received.

Apply to group of files

This field is displayed if you selected **Multiple** for the **File** parameter in source file properties.

Specify the rule for executing the script on the group of files. When sending a group of files, there are two types of request: a generic request that applies to the entire group, and a specific request. There will be one specific request if the transfer is done in grouped mode, and multiple specific requests (one for each file) if the transfer is done in file-by-file mode.

- On main request – Execute the script only on the main request
- For each file in group – Execute the script for each file in the group but not for the main request
- Both – Execute the script both for the main request and for each file in the group

Error

Script

Indicates whether to execute the default or a custom error script. The default script is defined in the configuration of the source Transfer CFT in the flow, and an error is raised..

File

If you select **Custom**, specify whether to use an existing file or upload a new file.

Filename

If you select **Custom** and select to use an existing file, specifies the custom script to run. If the file does not exist, the default error script defined in the Transfer CFT configuration is executed.

This name can include the following symbolic variables:

- &IDF, &PARM

- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

Receive template fields

The following are the fields to use when adding or editing a receive template for Transfer CFT in the Central Governance user interface. See *Manage templates* on page 362 for actions you can perform.

A receive template defines:

- The default name and local physical characteristics of the file to receive.
- The default actions to perform locally during and after the transfer. For example, translation, compression, call to a user exit, an end-of-transfer procedure, and so on.
- The default time slot and default user associated with the transfers.

Note You can define templates on Transfer CFT directly or by defining flows in Central Governance.

Name

The template name. The name must be unique across all CFTRECV objects defined on the Transfer CFT.

There cannot be a Central Governance flow with the same identifier that uses the Transfer CFT as a target.

Description

A description of the template.

Transfer properties

Transfer priority

Transfer priorities are equivalent to integer values ranging from 0 (low) to 255 (high). If you select **Custom**, you must enter an integer between 0 and 255. Note that the integer value for medium priority is 128.

When Transfer CFT reaches the maximum number of transfers allowed, it queues transfers. When an ongoing transfer is finished and a slot is available for a new transfer, the system selects the one with the highest priority.

Bandwidth allocation

The amount of bandwidth allocated to this flow. The value you select determines the data transfer rate for this flow.

Transfer state

Indicates the state of the transfer request.

- Ready - Transfer is available and can start immediately.
- On hold - Transfer is deferred until a remote receive request is accepted, or until a local start command changes this transfer to the ready state.
- Kept - Transfer is deferred until a local start command changes this transfer to the ready state.

User ID

The identifier of the transfer owner.

Detect duplicate transfers

This field is used in detecting duplicate transfers and may contain a list of symbolic variables separated by a period ".". Duplicate transfers can occur, for example, if there is an error in a processing script. Possible variables include:

- &PART
- &IDF
- &DIRECT
- &MODE
- &SAPPL , &RAPPL
- &IDA
- &SUSER , &RUSER
- &FNAME, &NFNAME
- &SYSDATE, &SYSTIME
- &PARM

Example: &PART.&IDF.&IDA.&SAPPL

See *Symbolic variables* on page 291 for descriptions of the variables.

Compress file

Indicates whether files are compressed before transferred.

Do not enable compression if the type of files being transferred would not benefit from it and possibly result in longer processing times. For example, do not activate compression if transferring JPG or ZIP files.

No file exists

Specifies the action taken if the received file does not already exist.

- Create – File is created.
- Cancel – Transfer is refused.

Invalid options are disabled. Note that you cannot select **Cancel** in this field and in the File exists field. If that was possible, all transfers would fail.

File exists

Specifies the action taken if the received file exists.

- Delete – Existing file is deleted.
- Cancel – Transfer is refused.
- Overwrite – Existing file is overwritten.
- Overwrite only if empty – Existing file is overwritten only if it has no data.
- Overwrite after receiving temporary file – Existing file is overwritten after receiving the temporary file. This option is available only for Transfer CFT on Unix. The user who performs the transfer must have rights to rename the temporary file name into the file to overwrite.

Invalid options are disabled. You cannot select **Cancel** in this field and in the No file exists field. If that was possible, all transfers would fail.

Aborted transfer

Specifies the action taken if a transfer is terminated due to a file creation error on the target.

- Keep – Transfer remains in the transfer list.
- Delete – Transfer is removed from the transfer list.

Delete file on purge

Indicates the transfer states of files that will be deleted when you remove the associated transfers from the transfer list or when you purge the transfer list. You can select any combination of statuses. If you do not select anything, files are not deleted even when the associated transfers are removed from the transfer list.

- Ready (D) - Transfer is available and can start immediately.
- Transferring (C) - Transfer is being executed.
- On hold (H) - Transfer was interrupted due to an error, such as a network failure, or by a user.
- Kept (K) - Transfer was suspended by Transfer CFT or by a user.

- Transferred (T) - Transfer was completed successfully.
- Executed (X) - Transfer was ended by an application or user.

File properties

Filename

Specify the file name or full path name for the received file or files. This field is required if the initiator of the flow is the source. Default value: pub\&IDF.&IDTU.&FROOT.RCV

The file name can include the following symbolic variables:

- &FDATE, &FTIME, &FYEAR, &FMONTH, &FDAY
- &SPART, &RPART, &PART, &IPART, &NPART, &GROUP
- &SUSER, &RUSER
- &SAPPL, &RAPPL
- &IDF, &PARM, &IDA
- &NIDF, &IDTU, &IDT
- &BDATE, &BTIME, &BYEAR, &BMONTH, &BDAY
- &NFNAME
- &NFVER

See *Symbolic variables* on page 291 for descriptions of the variables.

Temporary file

Specify the name of the temporary file used during the transfer. When the transfer is complete, the temporary file is renamed using the name defined in the Filename field. If you do not specify a value, Transfer CFT creates the file with the name specified in the Filename field. If the File exists parameter is set to Overwrite after receiving temporary file, Temporary file becomes required.

The file name can include the following symbolic variables:

- &FDATE, &FTIME, &FYEAR, &FMONTH, &FDAY
- &SPART, &RPART, &PART, &NPART, &GROUP
- &SUSER, &RUSER
- &SAPPL, &RAPPL
- &IDF, &PARM, &IDA
- &NIDF, &NFNAME, &IDT
- &BDATE, &BTIME, &BYEAR, &BMONTH, &BDAY

See *Symbolic variables* on page 291 for descriptions of the variables.

File properties > file encoding

The file encoding fields vary depending on the Transfer CFT operating system.

Windows and Linux

File type

Specify whether the file is a binary or text file.

The **Stream text** option is available only in legacy flows and not Central Governance flows.

Text file type

The following fields are displayed for the text file type only.

End of record character

Indicates the end of record character used in the file.

Ignore end of file character

This field is displayed only if you selected **CRLF** as the end-of-record character and if one of the source Transfer CFTs is on Windows.

- No - Transfer CFT ends the transfer when it encounters an end-of-file character.
- Yes - Transfer CFT continues the transfer until there is no more data.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, enter the character set in the provided field.

Transcoding

Represents how the data in the file is encoded while it is being sent to the target.

If you select **Custom**, enter the character set in the provided field.

See *Transcoding and character translation* on page 337.

Stream text file type

The following field is displayed for the stream text file type.

Encoding

Represents how the data in the file to be sent is encoded.

OS/400 (IBM i)

File type

The following describes the available options.

- **Data file** specifies the file is a PF-DTA file.
- **Save file** specifies the file is a SAVF file.
- **Source** specifies the file is a PF-SRC (with header) file.
- **OS 400 specific** specifies the file is a PF-SRC (no header) file.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, the Encoding charset field is displayed where you can enter the character set.

Transcoding

Represents how the data in the file is encoded while it is being sent to the target.

If you select **Custom**, the Transcoding charset field is displayed where you can enter the character set.

See *Transcoding and character translation* on page 337.

z/OS

File type

The following describes the available options.

- **Autodetect** specifies the file is sent in auto detection mode.
- **Print file with ASA jump codes** specifies the file is print file with ASA jump codes.
- **Print file with machine jump codes** specifies the file is print file with machine jump codes.
- **Spanned variable format** specifies the file is a spanned variable file.
- **ARDSSU** specifies the file is a ADRDSSU file.
- **Binary** specifies the file is a binary file.
- **Text** specifies the file is a text file.
- **Stream text** specifies the file is a text file sent in Stream CFT mode.
- **PDSE** specifies the file is a PDSE file.

Select **Binary** to send a mix of file types, and make sure there is no custom encoding or transcoding required for the file types:

- Print file with ASA jump codes
- Print file with machine jump codes
- Text

If the text files have custom encoding or transcoding requirements, you must define specific flows for them.

Encoding

Represents how the data in the file to be sent is encoded.

If you select **Custom**, the Encoding charset field is displayed where you can enter the character set.

Transcoding

Represents how the data in the file is encoded while it is being sent to the target.

If you select **Custom**, the Transcoding charset field is displayed where you can enter the character set.

See *Transcoding and character translation* on page 337.

File properties > record format

Record type

Indicates whether the records in the file are fixed or variable length.

Padding character

This field is displayed if you selected **Fixed** as the record type. Specify the character to use to pad the record. This character is added to the end of the record until it reaches the maximum length as defined in the Maximum record length field. If you do not provide a value, the default character is a space.

Trimming character

This field is displayed if you selected **Variable** as the record type. Specify the character to use to remove padding characters from the end of the record. For example, if the trimming character is a space and there are 5 spaces at the end of the record, all 5 spaces are removed. If you do not provide a value, the record is unchanged.

Maximum record length

If you select **Default OS value**, Transfer CFT will interpret correctly maximum record length as:

- On Windows, 512 characters
- On Linux or UNIX, 512 characters for text files; 4096 characters for binary files

If you select **Custom**, enter a value in the provided field.

Processing scripts

The target processing scripts section of the flow definition identify the files to execute at specific phases in the flow lifecycle. See *Flow lifecycle* on page 233 for detailed information.

For each phase, you can select whether to execute the default script or a custom script. Default scripts are defined in the configuration of the Transfer CFT involved in the flow. For example, ProdCat1 running on CFT100 is defined as the target in the flow. The default post-processing script defined in the CFT100 configuration is `exec/default_postprocessing.sh` (the actual default value is `exec/&IDF.sh`). If that is the script you want executed in the flow, you select the **Default** option.

To execute a different script, you must select the **Custom** option. Default scripts are defined in the Transfer processing section of the Transfer CFT configuration. See *Transfer processing* on page 175.

Processing scripts are executed on the target during the following phases:

- Post-processing – Executed after the file was received
- Acknowledgment – Executed after the file is transferred
- Error – Executed if an error occurs when a file is received

For setting a custom processing script, you can:

- Set an existing script already on the target Transfer CFT.

  or

- Upload a new script. The new script is available on the Transfer CFT after the flow is deployed successfully on the Transfer CFT. The new script is uploaded to Transfer CFT with the default path `$(cft.runtime_dir)/conf/ws_upload`. If the file already exists on Transfer CFT, it is overwritten.

Post-processing

Script

Indicate whether to execute the default or a custom post-processing script. The default script is defined in the configuration of the target Transfer CFT in the flow.

File

If you select **Custom**, specify whether to use an existing file or upload a new file.

Filename

If you select **Custom** and select to use an existing file, specify the custom script to run. If the file does not exist, no post-processing is performed even if a default post-processing script is defined in the Transfer CFT configuration, and an error is raised.

This name can include the following symbolic variables:

- &IDF, &PARM

- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

Acknowledgment

Script

Indicate whether to execute a custom acknowledgment script.

Filename

If you select **Custom** and select to use an existing file, specify the custom script to run. If the file does not exist, no processing is performed even if a default acknowledgment script is defined in the Transfer CFT configuration, and an error is raised.

This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

State

Indicates whether the transfer must wait for an acknowledgment.

- Require – Transfer must wait for an acknowledgment before it can be considered complete.
- Ignore – Transfer can be considered complete, even if an acknowledgment is not received.

Error

Script

Indicate whether to execute the default or a custom error script. The default script is defined in the configuration of the target Transfer CFT in the flow.

File

Filename

If you select **Custom** and select to use an existing file, specify the custom script to run. If the file does not exist, no processing is performed even if a default error script is defined in the Transfer CFT configuration, and an error is raised.

This name can include the following symbolic variables:

- &IDF, &PARM
- &PART, &RPART, &SPART, &GROUP
- &RUSER, &SUSER, &USERID
- &RAPPL, &SAPPL
- &NIDF

See *Symbolic variables* on page 291 for descriptions of the variables.

Choose file

If you select **Custom** and select to upload a new file, select the file. The destination path is configured on Transfer CFT by the parameter `copilot.webservices.upload_directory`.

Manage partners

Use the following procedures to manage partners. This includes listing, viewing, adding, editing, deploying and removing partners.

List partners

The following are the steps to navigate to partners on the Transfer CFT Legacy Flows page for a selected Transfer CFT.

1. Click **Products** on the top toolbar to open the Product List page.
2. Find the Transfer CFT you want and click its name to open its details page.
3. Click **Legacy flows** to open the Transfer CFT Legacy Flows page for the selected Transfer CFT.
4. Select **Partners** to list the names, statuses and descriptions of the partners for the Transfer CFT.

Add partner

The starting point for the following steps is when partners for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

1. Click **Add partner** to open the Add Partner page for the selected Transfer CFT.
2. Complete the fields. See *Partner fields* on page 385 for information.
3. Click **Save** to add the partner.

Deploy partner

The starting point for the following steps is when partners for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

1. Select a partner with a status of Saved, not deployed.
2. Click **Deploy** to deploy the partner to the Transfer CFT.

View, edit partner

The starting point for the following steps is when partners for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

1. Click the name of a partner to open its details page.
2. Click **Edit** to change the partner. See *Partner fields* on page 385 for information.
3. Click **Save** to save changes to the partner.

Remove partner

The starting point for the following steps is when partners for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

Do one of the following to remove a partner from the selected Transfer CFT:

- Select a partner and click **Remove**.
- Click the name of a partner to open its details page. Click **Remove**.

If the partner you remove is deployed on Transfer CFT, the partner also is removed from it.

Partner fields

This topic describes the fields when adding or editing a partner in legacy flows. See *Manage partners* on page 384 for actions you can perform.

The user interface specifies default values for fields with them. You can change default values as needed.

Note Partners and distribution lists can be defined directly on Transfer CFT or on Central Governance via flows.

Name

The partner identifier. The name must be unique for all partners and distribution lists defined on Transfer CFT.

Description

Optionally, a description of the partner.

Partner access

Relay

Local identifier of a relay partner.

Local partner name

Identifier Transfer CFT uses to identify itself to a partner.

The partner name must be unique across all partners and broadcast and collect lists on Transfer CFT.

There cannot be a Central Governance flow with the same broadcast list name that uses the instance of Transfer CFT as the source.

There cannot be a Central Governance flow with the same collect list name that uses the instance of Transfer CFT as the target.

Local partner password
Confirm local partner password

Password Transfer CFT uses to identify itself to a partner.

Remote partner name

Identifier of the remote partner Transfer CFT. If you do not enter a value, the remote partner name defaults to the partner name entered previously.

The remote partner name must be unique among all nrpart partner values for the instance of Transfer CFT.

Remote partner password
Confirm remote partner password

Password of the remote partner Transfer CFT.

Host

Host of the remote partner Transfer CFT. See *Operating systems and deployment correspondence* on page 483.

Operating system

Operating system of the remote partner Transfer CFT.

Protocol

Security profile

Local security profile used for transfers with the remote partner.

Protocol

Local protocol used for transfers with the remote partner.

Port

Port of the remote partner Transfer CFT.

Network sessions

The following fields are for customizing sessions allocated to the partner. A value must be set in the Host field to enable editing these fields. The value of each field is independent of the other fields.

Incoming connections

Maximum number of sessions for inbound connections (server mode).

Outgoing connections

Maximum number of sessions for outbound connections (requester mode).

Total connections

Maximum number of communication sessions for the partner.

Manage distribution lists

Use the following procedures to manage distribution lists. This includes viewing, adding, editing, deploying and removing lists.

Note Partners and distribution lists can be defined directly on Transfer CFT or on Central Governance via flows.

List distribution lists

The following are the steps to navigate to distribution lists on the Transfer CFT Legacy Flows page for a selected Transfer CFT.

1. Click **Products** on the top toolbar to open the Product List page.
2. Find the Transfer CFT you want and click its name to open its details page.
3. Click **Legacy flows** to open the Transfer CFT Legacy Flows page for the selected Transfer CFT.
4. Select **Distribution lists** to list the names, statuses and descriptions of the distribution lists for the Transfer CFT.

Add distribution list

The starting point for the following steps is when distribution lists for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

1. Click **Add distribution list** to open the Add Distribution List page for the selected Transfer CFT.
2. Complete the fields. See *Distribution list fields* on page 389 for information.
3. Click **Save** to add the partner.
4. Optionally, click **Deploy** to deploy the list to the Transfer CFT. You can skip this step and deploy the list later if you prefer.

Deploy distribution list

The starting point for the following steps is when distribution lists for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

1. Select a distribution list with a status of Saved, not deployed.
2. Click **Deploy** to deploy the list to the Transfer CFT.

View, edit distribution list

The starting point for the following steps is when distribution lists for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

1. Click the name of a distribution list to open its details page. If the list has not been deployed, you can click **Deploy** to do so.
2. Click **Edit** to change the distribution list. See *Distribution list fields* on page 389 for information.
3. Click **Save** to save changes to the distribution list.

4. Optionally, click **Deploy** to deploy the list to the Transfer CFT. You can skip this step and deploy the list later if you prefer.

Remove distribution list

The starting point for the following steps is when distribution lists for a selected Transfer CFT are displayed on the Transfer CFT Legacy Flows page.

Do one of the following to remove a distribution list from the selected Transfer CFT:

- Select a distribution list and click **Remove**.
- Click the name of a distribution list to open its details page. Click **Remove**.

If the removed list had been deployed on Transfer CFT, the list also is removed from the Transfer CFT.

Distribution list fields

This topic describes the fields when adding or editing a distribution list in legacy flows. See *Manage distribution lists* on page 387 for actions you can perform.

Note Partners and distribution lists can be defined directly on Transfer CFT or on Central Governance via flows.

Name

The distribution list identifier. The name must be unique for all partners and distribution, broadcast and collect lists defined on the Transfer CFT.

There cannot be a Central Governance flow that uses the Transfer CFT as a source and has a broadcast list with same name.

There cannot be a Central Governance flow that uses the Transfer CFT as a target and has a collect list with same name.

Type

Defines the partner list as one of the following:

- Partner list - Explicitly using a list type parameter.
- File - Using a file in which the list of partners is saved.

These methods are mutually exclusive. A partner included in a list cannot itself be a broadcasting list.

If you select file as the type, you can do one of the following:

- Use existing file - Specify a file already on the Transfer CFT.

- Upload new file - The new file is available on Transfer CFT after the distribution list is deployed successfully on it. The new file is uploaded to Transfer CFT with the default path `$(cft.runtime_dir)/conf/ws_upload`. The file is overwritten if it already exists on Transfer CFT.

Unknown partner

Specifies the action to take when a target in the distribution list is not found.

- Continue – Display an informational message and continue processing.
- Ignore – Continue processing without an informational message.
- Cancel – The transfer stops at the first error, but all transfers started before the error occurred continue. For example, if there are 10 targets in the list and the fourth one is unknown, targets 1-3 receive the file, but targets 4-10 do not.

Processing scripts submission

Apply pre-processing
Apply post-processing
Apply acknowledgment scripts

These fields specify when to apply processing scripts.

- On main request – Executes the script only on the main request.
- For each partner in the list – Executes the script only for each partner in the list.
- Both – Executes the script both for the main request and for each partner in the list.

Environment promotion and staging 26

This documentation provides guidelines and examples for environment promotion and staging in Central Governance, with a focus on flows.

Environment promotion and staging relies on the import and export commands of CLI. Before reading this documentation, see *Command line interface* on page 78 and review the following commands:

- appExport
- appImport
- partnerExport
- partnerImport
- flowExport
- flowImport
- flowDeploy

Guidelines

While reviewing the following promotion and staging guidelines, keep in mind that Central Governance does not have environment awareness and that an instance of a product can register with only one instance of Central Governance.

These guidelines provide the ideal deployment pattern for staging. Applying them assures the simplest possible promotion without the need for external tooling or manual actions to adapt the export files. Some of these guidelines might not apply to you, because of your existing product infrastructure or the nature of your business. For example, typically, a retailer cannot have as many stores in testing as in production.

- Install one instance of Central Governance per environment. For example, if you have development, testing, pre-production and production environments, you need four instances of Central Governance.
- The same applies for registered products. If you have four instances of Central Governance, install and register one instance of a product per Central Governance environment. Do not register the same instance of a product in multiple Central Governance environments.
- Have one-to-one mapping in the number and names of applications between environments. For instance, the General Ledger application should have the same name in all environments, but all other details, including the host name, can differ from one environment to another.

- If, in the case of Transfer CFTs, application groups are used in flows, typically in a corporate-to-store pattern, have one-to-one mapping in the number and names of application groups between environments. For instance, the group containing all stores should have the same name in all environments, but all other details can differ from one environment to another.
- Have one-to-one mapping in the number, names and identifiers of flows between environments.
- Have one-to-one mapping in the number and names of partners between environments. For instance, your food supplier should have the same name in all environments, but all other details, such as communication profiles, can differ from one environment to another.
- It is strongly recommended to:
    - Use the same communication profile aliases between environments and ensure that communication profile properties are compatible.
    - Use the same credential aliases for certificates and SSH keys.
    - Use the same PGP keys aliases used by SecureTransport processing steps.

Having one-to-one mapping in the number and names of products between environments is not required or recommended. The only recommendation relates to when relays are used. In such cases, having the same name across environments is strongly recommended.

Flow promotion use cases

All of the following use cases presume product parameters within a flow are identical in the source and the target environments. If not, the flow should be edited manually in the target Central Governance after import or the export file should be adapted outside of Central Governance, manually or via automated tooling.

Application to application

Sample flows:

- Application 1 (Transfer CFT 1) > Application 2 (Transfer CFT 2)
- Application 1 (Transfer CFT 1) > Application Group 1
- Application 2 (Transfer CFT 2), Application 3 (Transfer CFT 3) > Application 1 (Transfer CFT 1)

New flow: Same application and Transfer CFT names

Assuming applications and Transfer CFTs exist and have the same names in both Central Governance instances, but the flow doesn't exist in the target Central Governance.

- Export the flow from the source Central Governance with the following command:

```
flowExport -n <flow name>
```

- Import the flow in the target Central Governance with the following command:

```
flowImport –f <file name>
```

The flow is imported, but not deployed, in the target Central Governance.

The same logic applies if promoting multiple flows at once. In such case provide the list of flow names, separated by commas, or provide a name pattern matching the flows to export. This comment also applies to all of the following use cases.

Existing flow: Same application and Transfer CFT names

Assuming the flow exists in the target Central Governance, the only difference from the preceding case is to overwrite the flow upon import.

- Export the flow from the source Central Governance with the following command:

  ```
  flowExport –n <flow name>
  ```

- Import the flow in the target Central Governance with the following command:

  ```
  flowImport –f <file name> -o
  ```

The flow is imported, but not deployed, in the target Central Governance.

New flow: Same application names, different Transfer CFT names

Assuming applications and Transfer CFTs exist in both Central Governance instances, but the Transfer CFTs have different names, and the flow doesn't exist in the target Central Governance.

- Export the flow from the source Central Governance with the following command:

  ```
  flowExport –n <flow name>
  ```

- Import the flow in the target Central Governance with the following command:

  ```
  flowImport –f <file name> -ai.
  ```

Without the **–ai** (**--allowincomplete**) option, the flow cannot be imported correctly because its definition points to an instance of Transfer CFT that does not exist in the target Central Governance. With this option, the target Central Governance ignores the Transfer CFTs referenced in the file and instead includes in the flow the Transfer CFTs linked to the applications already defined in the target Central Governance.

The flow is imported, but not deployed, in the target Central Governance.

Existing flow: Same application names, different Transfer CFT names

Assuming the flow exists in the target Central Governance, the only difference from the preceding case is to overwrite the flow upon import.

- Export the flow from the source Central Governance with the following command:

```
flowExport -n <flow name>
```

- Import the flow in the target Central Governance with the following command:

```
flowImport -f <file name> -ai -o
```

The flow is imported, but not deployed, in the target Central Governance.

Different number of applications and Transfer CFTs

Both of the following cases assume the number of applications and Transfer CFTs in the flow are different in the source and target environments. The typical use case is corporate-to-store, with many stores in the production environment and few stores in other environments. Best practice in is to rely on application groups in flows.

New flow

Assuming the application groups exist in both Central Governance instances and the flow doesn’t exist in the target Central Governance.

- Export the flow from the source Central Governance with the following command:

```
flowExport -n <flow name>
```

- Import the flow in the target Central Governance with the following command:

```
flowImport -f <file name>
```

Thanks to the abstraction layer provided by the application groups, it doesn't matter that the application groups differ in number or names in the source and target instances. The imported flow points to the existing application groups in the target Central Governance.

The flow is imported, but not deployed, in the target Central Governance.

Existing flow

Assuming the flow exists in the target Central Governance, the only difference from the preceding case is to overwrite the flow upon import.

- Export the flow from the source Central Governance with the following command:

```
flowExport -n <flow name>
```

- Import the flow in the target Central Governance with the following command:

```
flowImport -f <file name> -o
```

The flow is imported, but not deployed, in the target Central Governance.

Application to application with relays

Sample flow:

- Application 1(Transfer CFT 1) > Transfer CFT relay >Application 2 (Transfer CFT 2)

For applications, use the cases in *Application to application* on page 392.

Relays are linked correctly if relay names match existing Transfer CFTs. You need to use the **–ai** (**--allowincomplete**) option if you have a different Transfer CFT relay names or a different number of relays or both.

Application to business and business to application

Sample flows:

- Application (SecureTransport) > Partner 1
- Partner 1 > Application (SecureTransport)

Note An application with SecureTransport in an application group is not a supported use case.

Partners

When imported flows have partners that exist in the target environment, the flows are relinked to the partners based on the partner names. Partners in the source and target environments should have the same names and the same or compatible server communication profiles (see *Prerequisites for promoting flows* on page 398). Credentials can differ provided aliases are the same.

Best practice is using the same partner names between the environments. If partner names differ between environments, change SecureTransport configuration in imported flows to match the target partners. You must check and update receive, file processing and send properties.

One of the following cases applies when partners do not exist in the target environment. You must first import partners to the target environment because flow import does not support importing partners.

New partners

Assuming partners do not exist in the target environment.

- Export the partners from the source Central Governance with the following command:

  ```
  partnerExport –n <partner names>
  ```

  Or export all partners with:

  ```
  partnerExport
  ```

- Import the partner in the target Central Governance with the following command:

  ```
  partnerImport –f <filename>
  ```

The partners are imported in the target Central Governance

Existing partners

Assuming partners exist in the target Central Governance, the only difference from the preceding case is to overwrite the partner upon import.

- Export the partners from the source Central Governance with the following command:

```
partnerExport -n <partner names>
```

- Import the flow in the target Central Governance with the following command:

```
partnerImport -f <file name> -o
```

The partners are imported and updated in the target Central Governance.

Applications

The following cases apply when application names are the same, but names of SecureTransports are the same or different.

New flow: Same application and SecureTransport names

Assuming applications and SecureTransport exist and have the same names in both Central Governance instances, but the flow doesn't exist in the target Central Governance.

- Export the flow from the source Central Governance with the following command:

```
flowExport -n <flow name>
```

- Import the flow in the target Central Governance with the following command:

```
flowImport -f <file name>
```

The flow is imported, but not deployed, in the target Central Governance.

The same logic applies if promoting multiple flows at once. In this case provide the list of flow names, separated by commas, or provide a name pattern matching the flows to export. This comment also applies to all of the following use cases.

Existing flow: Same application and SecureTransport names

Assuming the flow exists in the target Central Governance, the only difference from the preceding case is to overwrite the flow upon import.

- Export the flow from the source Central Governance with the following command:

```
flowExport -n <flow name>
```

- Import the flow in the target Central Governance with the following command:

```
flowImport -f <file name> -o
```

The flow is imported, but not deployed, in the target Central Governance.

New flow: Same application names, different SecureTransport names

Assuming applications and SecureTransports exist in both Central Governance instances, but the SecureTransports have different names and the flow doesn't exist in the target Central Governance.

- Export the Flow from the source Central Governance with the following command:

```
flowExport -n <flow name>
```

- Import the flow in the target Central Governance with the following command:

```
flowImport -f <file name> -ai.
```

Without the **–ai** (**--allowincomplete**) option, The flow cannot be imported correctly because its definition points to an instance of SecureTransport that does not exist in the target Central Governance. With this option, the target Central Governance ignores the SecureTransports referenced in the file and instead includes in the flow the SecureTransports linked to the applications already defined in the target Central Governance.

The flow is imported, but not deployed, in the target Central Governance.

Existing flow: Same application names, different SecureTransport names

Assuming the flow exists in the target Central Governance, the only difference from the preceding case is to overwrite the flow upon import.

- Export the flow from the source Central Governance with the following command:

```
flowExport -n <flow name>
```

- Import the flow in the target Central Governance with the following command:

```
flowImport -f <file name> -ai -o
```

The flow is imported, but not deployed, in the target Central Governance.

Application to business and business to application with SecureTransport relay

Sample flows:

- Application (Transfer CFT) > SecureTransport relay > Partner 1, Partner 2
- Partner 1, Partner 2 > SecureTransport relay > Application (Transfer CFT)

For partners see the use cases in *Partners* on page 395.

For applications see the use cases in *Applications* on page 396.

For relays you must use the same SecureTransport relay name.

Deploying promoted flows

You can deploy promoted flows in the Central Governance user interface. However, the preferred method is via the command-line if there is a policy to avoid using the UI in a production environment. If so, the only remaining operation is to run the command:

```
flowDeploy -n <flow names>
```

Prerequisites for promoting flows

There are prerequisites for exporting and importing flows when promoting flows from one Central Governance environment to another. Promotion is when, for example, you export flows from a development or staging environment and import them to a production environment. The promoted flows remain in the source environment unless you remove them after promoting to the target environment.

Review these topics before promoting flows using CLI or the user interface. For CLI see *flowExport* on page 85 and *flowImport* on page 85. For the UI see *Back up flows from UI* on page 294. As usual, users must have roles with proper privileges to run specific actions. See *Permissions enforcement* on page 80.

The import algorithm

Central Governance uses an algorithm that imports objects in the following sequence:

1. Application groups
2. Participants (all sources and targets in flows, including partners, applications, application groups, unmanaged products)
3. Credentials
4. Profiles
5. Flows

The first two steps are triggered when using parameters like **importapplications** or **importunmanagedproducts** with *flowImport* on page 85. If the parameters are not used, it is presumed participants already exist in Central Governance, and links to them are made as described in *Conditions about participants* on page 401 and *Conditions about file-transfer middleware* on page 401. If conditions are unmet, import fails unless the **allowincomplete** parameter is used. If **allowincomplete** is used, participants that do not exist are removed as flow sources, targets and relays and flow status is recalculated.

Next, import of credentials and profiles is attempted. Lastly, flows are imported. For profiles:

- Existing client communication profiles are checked for compatibility. Client communication profiles that do not exist are created if compatible.
- Existing server communication profiles are checked for compatibility. Importing stops if server communication profiles do not exist or are not compatible.

The compatibility checks Central Governance performs are described in the next topic.

Conditions about protocols and communication profiles

Communication profiles in exported flow files must be compatible with profiles in the instance of Central Governance that imports the flows. This applies to client and server communication profiles when, upon importing, Central Governance finds a client or server communication profile with the same name.

When a flow is imported, Central Governance creates client communication profiles when they do not exist in the target environment. Conversely, server communication profiles are never created upon importing flows; they are reused if they exist with the same name.

Central Governance checks for compatibility when it detects a communication profile in an imported flow with the same name of an existing profile. The type of communication profile (client or server) must match and the protocol must be the same. The profiles must have the same protocol details, as outlined in the following tables.

HTTP details			
Property	**Property value**	**Server communication profile**	**Client communication profile**
SSL/TLS	None Client optional Server only Mutual authentication	Enable SSL/TLS = No Enable SSL/TLS = Yes ; Client authentication required: Optional Enable SSL/TLS = Yes ; Client authentication required: No Enable SSL/TLS = Yes ; Client authentication required: Yes	Enable SSL/TLS = No Enable SSL/TLS = Yes ; No certificate authentication or certificate Enable SSL/TLS = Yes ; No certificate authentication Enable SSL/TLS = Yes ; certificate
HTTP method	PUT POST GET	Must support PUT Must support POST Must support GET	PUT POST GET
FIPS	YES NO	YES NO	YES NO

PeSIT details			
Property	**Property value**	**Server communication profile**	**Client communication profile**
Network protocol	TCP pTCP UDT	TCP pTCP UDT	TCP pTCP UDT
SSL/TLS	None Client optional Server only Mutual authentication	Enable SSL/TLS = No Enable SSL/TLS = Yes ; Client authentication required: Optional Enable SSL/TLS = Yes ; Client authentication required: No Enable SSL/TLS = Yes ; Client authentication required: Yes	Enable SSL/TLS = No Enable SSL/TLS = Yes ; No certificate authentication or certificate Enable SSL/TLS = Yes ; No certificate authentication Enable SSL/TLS = Yes ; certificate
FIPS	YES NO	YES No	YES NO

SFTP details			
Property	**Property value**	**Server communication profile**	**Client communication profile**
Client authentication	Public key Password Password or public key	Public key Password Password or public key	Public key Password Password or public key
FIPS	YES NO	YES NO	YES NO

FTP details			
Property	**Property value**	**Server communication profile**	**Client communication profile**
Connection mode	Active Passive	Active or Both Passive or Both	Active Passive

FTP details			
Property	**Property value**	**Server communication profile**	**Client communication profile**
SSL/TLS	None Client optional Server only Mutual authentication	Enable SSL/TLS = No Enable SSL/TLS = Yes ; Client authentication required: Optional Enable SSL/TLS = Yes ; Client authentication required: No Enable SSL/TLS = Yes ; Client authentication required: Yes	Enable SSL/TLS = No Enable SSL/TLS = Yes ; No certificate authentication or certificate Enable SSL/TLS = Yes ; No certificate authentication Enable SSL/TLS = Yes ; certificate
FIPS	YES NO	YES NO	YES NO

When a server communication profile for a partner, product or unmanaged product is used by a protocol in a flow, the profile, based on name, must exist in Central Governance. If not the flow import fails, except when the **allowincomplete** parameter is used, which results in the protocol becoming undefined in Central Governance.

Conditions about participants

Participants in a flow to be imported must exist in the target environment. Participants are used as sources and targets in flows, including partners, applications and application groups. The following rules are enforced upon importing:

- An application exists in Central Governance if an application with the same name and host exists.
- An application group or partner exists in Central Governance if an application group or partner with the same name exists.

These conditions can be bypassed by using the **allowincomplete** option. In this case the participants are removed from the source or target of the flow, and the flow status is recalculated according to the new flow content.

Conditions about file-transfer middleware

File-transfer middleware — products or unmanaged products used in flows as sources, relays or targets — must exist in the target environment when flows are imported. Central Governance enforces the following rules upon importing:

- A product exists in Central Governance if a product with the same name and same product type exists and if the product status indicates the product is registered successfully.
- An unmanaged product exists in Central Governance if an unmanaged product with the same name exists.

These conditions can be bypassed by using the **allowincomplete** option. In this case the file-transfer middleware is removed from the source, target or relay of the flow, and the flow status is recalculated according to the new flow content.

Summary of export and import actions

The following table is a summary of export and import actions with CLI. Note that FGAC is sometimes required. See *Fine-grained access control* on page 122 for more about FGAC.

Action	Option used	Required privilege	Comment
Export a flow		View Flow and FGAC	
Export an application		View Application and FGAC	If the user exports a flow linked to an application and does not have the View Application privilege, or does not the right to view this application, the flow is exported with the reference to this application, but not its definition.
Export a group		View Group and FGAC	If the user exports a flow linked to a group and does not have the View Group privilege, or does not have the right to view this group, the flow is exported with the reference to this group, but not its definition.
Export an unmanaged product		View Unmanaged Product	If the user exports a flow linked to an unmanaged product and does not have the View Unmanaged Product privilege, the flow is exported with the reference to this unmanaged product, but not its definition.

Action	Option used	Required privilege	Comment
Export a partner		View Partner	If the user exports a flow linked to a partner and does not have the View Partner privilege, the flow is exported with the reference to this partner, but not its definition.
Import a flow		Create and View Flow and FGAC and View {object}	If the user imports a flow linked to an object (application, group, partner, unmanaged product) on which the user has no rights, the flow is not imported.
Import a flow	allowincomplete	Create and View Flow and FGAC	If the user imports a flow linked to an object on which the user has no rights, the flow is imported, but without this object, and the flow is incomplete.
Import an application	importapplications	Create and View Application and FGAC	If the user tries to import a flow containing an application on which the user has no rights, both the flow and the application are not imported.
Import an application	allowincomplete	Create and View Application and FGAC	If the user tries to import a flow containing an application on which the user has no rights: - The flow is imported but without the application (due to the **allowincomplete** option). - The application is not imported.
Import a partner	importpartners	Create and View Partner	If the user tries to import a flow containing a partner but does not have the create permission, both the flow and the partner are not imported.

Action	Option used	Required privilege	Comment
Import a partner	allowincomplete	Create and View Partner	If the user tries to import a flow containing a partner but does not have the create permission: - The flow is imported but without the partner (due to the **allowincomplete** option) - The partner is not imported
Import a group	importapplications	Create and View Group and FGAC	If the user tries to import a flow containing a group on which the user has no rights, both the flow and the group are not imported.
Import a group	allowincomplete	Create and View Group and FGAC	If the user tries to import a flow containing a group on which he has no right: - the flow is imported but without the group (due to the allowincomplete option) - the group is NOT imported
Import an unmanaged product	importunmanagedproducts	Create and View Unmanaged Product	If the user tries to import a flow containing an unmanaged product but does not have the create permission, both the flow and the unmanaged product are not imported.
Import an unmanaged product	allowincomplete	Create and View Unmanaged Product	If the user tries to import a flow containing an unmanaged product but does not have the create permission: - The flow is imported but without the unmanaged product (due to the **allowincomplete** option). - The unmanaged product is not imported.
Overwrite an existing flow	overwrite	Modify and View Flow and FGAC	

Alert rules 27

An alert is an event that occurs in Central Governance, such as a registration failure for a product. An alert rule can be triggered when the alert event matches the conditions in the alert rule. All alert rules send email notifications to the recipients defined in the rule.

An alert rule consists of conditions and notification sections:

- Conditions section defines the conditions for triggering the alert rule.
- Notification section contains the message to be sent to the specified recipients.

See *Edit alert rule messages, recipients* on page 409 for more information about conditions and notifications.

You can make copies of predefined alert rules and change conditions and notification information.

Note Although you can edit predefined alert rules, best practice is to make copies of predefined rules and change the copies. Copies of predefined rules are recognized as user-defined rules. If you change predefined rules and later install an upgrade for Central Governance, the rules revert to their default configurations. However, your user-defined rules are unaffected when you apply an upgrade.

Central Governance provides the following predefined alert rules.

Flow error

A flow error alert is triggered when errors occur during the execution of a flow. The alert message contains the following information:

- Flow name
- Reason for the failure, which includes a return code and message
- Source application
- Target application
- Source product
- Target product
- File name
- A link to view transfer details

Product configuration deployment error

Triggered when a configuration deployment fails on a product. The alert message contains the following information:

- Name of the product and product type
- Name of the host where the product is installed
- Date and time of the deployment
- Reason for the failure
- A link to view configuration details

Flow deployment error

Triggered when a flow deployment fails on a product. The alert message contains the following information:

- Name of affected flow
- Name of the product and product type
- Name of the host where the product is installed
- Date and time of the deployment
- Reason for the failure
- A link to view flow details

Product failure

Triggered when a product is not running. The alert message contains the following information:

- Name of the product and product type
- Name of the host where product is installed
- Status of the product
- Reason for the failure
- A link to view the product details

Product registration error

Triggered when a product fails to register successfully. The alert message contains the following information:

- Name of the product and product type
- Name of the host where the product is installed
- Status of the product, which is always "registered in error."
- Reason for the failure
- A link to view the product details

Use Alert Rule List page

Click **Alert Rules** on the top toolbar to open the Alert Rule List page. Use the page to perform the following actions.

Sort rules

Click a column heading to sort rules by name, subscription, status or source in ascending or descending order.

Filter rules

Use the Filter Add drop-down list to filter the list of displayed rules by name, subscription, status, source or description.

Activate or deactivate a rule

Select one or more alert rules and click **Activate** or **Deactivate**. All rules are inactive by default.

Activating and deactivating also are available options when you click the name of a rule to open its details page.

See *Why deactivate an alert rule* on page 408.

Subscribe or unsubscribe to a rule

This adds or removes your email address to the list of recipients of alert messages. Your user account must include your email address to subscribe.

- To subscribe, select one or more alert rules and click **Subscribe**. An envelope icon is displayed next to the alert rule name to indicate you are subscribed. A user with the proper privilege can verify by clicking the alert rule name and checking for the email address in the To field under the Notification section.
- To unsubscribe, select one or more subscribed rules and click **Unsubscribe**. The envelope icon disappears and your email address is removed from the To field under the Notification section.

Users with the default IT Manager and Middleware Manager roles can subscribe and unsubscribe. Both roles let users view or edit the list of email recipients for an alert.

Subscribing and unsubscribing also are available options when you click the name of a rule to open its details page.

Copy a rule

Select one or more rules and click **Copy** to duplicate the rule or rules. The names of copied rules are appended with Copy [n] to indicate they are copies of the original. For example, Flow error - Copy 1.

You can make a copy of a copy. The name of such a copy is appended with Copy [n] - Copy [n].

A copied rule is identified on the Alert Rule List page as user defined under the Source column. A copied rule is inactive by default. You can edit a user defined rule just as you can a predefined rule. See *Edit alert rule messages, recipients* on page 409 for more information.

Copying a rule also is an available option when you click the name of a rule to open its details page.

Edit a rule

Click the name of a rule to open its details page and then click **Edit**. See *Edit alert rule messages, recipients* on page 409 for more information.

Remove a rule

You can remove only copies of alert rules, which are identified as user-defined rules. You cannot remove original rules, which are identified as predefined rules.

Select one or more copies of a rule, click **Remove** and click the **Remove** option in the confirmation window.

Removing a copy of a rule also is an available option when you click the name of a rule to open its details page.

Why deactivate an alert rule

The following explains why you might want to deactivate a rule.

As the IT Manager, you have scheduled network maintenance on Saturday and know Central Governance and a product will have communication problems for about one hour. If the product failure alert rule remains active during the maintenance period, you will receive many meaningless notifications. To avoid this, deactivate the rule before starting the maintenance. Reactivate it after maintenance is completed. If you remove the rule instead of deactivating it, you would have to reconfigure it entirely.

If Central Governance detects an alert while the rule is disabled, no email notifications are sent. When you reactivate the rule, email notifications are sent only for the alerts detected after the rule is reactivated.

You can activate and deactivate alert rules on the Alert Rule List page or when viewing the details of a specific alert. See *Use Alert Rule List page* on page 407.

Edit alert rule messages, recipients

You can view or edit the messages Central Governance sends to recipients of triggered alert rules and view or change lists of message recipients.

A recipient can be any person with an email address.

Note Although you can edit predefined alert rules, best practice is to make copies of predefined rules and change the copies. Copies of predefined rules are recognized as user-defined rules. If you change predefined rules and later install an upgrade for Central Governance, the rules revert to their default configurations. However, your user-defined rules are unaffected when you apply an upgrade.

Rule editing steps

The following are the steps for editing an alert rule.

1. Click **Alert Rules** on the top toolbar to open the Alert Rule List page.
2. Click a rule name to open its details page.
3. Click **Edit**.
4. Enter the changes you want and click **Save changes** when done.

See:

- *Conditions fields* on page 409 for information about the Conditions section.
- *Notification fields* on page 411 for information about the Notifications section

Conditions fields

The following are the conditions for each default alert. Except for the Status condition of the product failure alert, all conditions are set to **Any** by default, but can be reset to specific single or multiple values.

Flow error

Application name

Name of one or more applications, source or target.

File name

Name of one or more file names or a pattern using the asterisk (*) wildcard character.

Flow name

Name of one or more flows.

Product name

Name of one or more products.

Product configuration deployment error

Product name

Name of one or more products.

Product type

The types of products the rule applies to.

Flow deployment error

Flow name

Name of one or more flows.

Product name

Name of one or more products.

Product type

The types of products the rule applies to.

Product failure

Product name

Name of one or more products.

Product type

The type of products the rule applies to.

Status

One or more of the following statuses: stopping, stopped, unreachable, partially started, started in error, stopped in error, in progress.

Product registration error

There are no conditions defined for this alert.

Notification fields

The following describes the fields in the Notification section.

From

An email address for the message sender. The default value is an invalid no-reply address.

To

One or more recipient addresses. If multiple addresses, separate each address with a comma. For example:

```
name1@domain1.com,name2@domain2.com
```

An alert message is sent to each address.

Subject

The subject line of the alert message. You can use values derived from the context of the rule itself. See *Use context values in notifications* on page 411.

Message

The alert message. A full-featured HTML editor is provided for composing the body of the notification email. You can use values derived from the context of the rule itself. See *Use context values in notifications* on page 411.

Use context values in notifications

When editing an alert rule you can use values derived from the context of the rule, including the functions and data associated with the rule, in the subject and message of your email notifications. The INSERT drop-downs to the right of the Subject and Message fields are for inserting these values, which are comprised of attributes, conditions and variables.

When you click the INSERT drop-down for the subject or message, an additional drop down and a scrollable list are displayed. The first drop down enables you to select the type of context value to add: attributes, conditions or variables.

If you select attributes, a second drop down enables you to select the Tracked Object associated with the rule from which the attributes are retrieved. The list has the available attributes.

If you select conditions or variables, the available conditions or variables are displayed. No other drop-down menus are displayed.

When you select a context value item, it is displayed in the subject or message of the notification.

Central Governance replaces the attributes, conditions and variables in the subject and message with values just before sending alert messages.

The Visibility user documentation describes how to use these. To access the documentation, select **Flows > Monitoring** to open the Visibility user interface. Select **Help > Help** and go to **Event processing rules > View the rules and edit rules details > Use context values in alert rule notifications**.

Deployment monitoring 28

The Deployment List page is for reviewing the status of objects in the process of deploying or updating, have deployed or updated successfully, or have failed. You can use a retry action for failed deployments and updates.

Deployment monitoring concepts

Central Governance monitors as objects are being deployed and updated. Never-deployed objects are not tracked on the Deployment List page.

When objects deployed previously are changed, you must redeploy them to update the page. For example, you change a port in the configuration of one Transfer CFT, but the change is reflected on the Deployment List page only when the configuration is redeployed.

Monitoring for SecureTransport

On the Deployment List page you can monitor deployments of flows on each SecureTransport involved in the flow.

Central Governance does not deploy the following objects to SecureTransport and monitoring is not applicable: configurations, policies, legacy flows and updates.

Flows have global statuses not represented on the Deployment List page. Global statuses are listed in the Status columns on the Flow List page. In contrast, the specific status of a policy or flow for each linked product is displayed on the Deployment List page.

Monitoring for Transfer CFT

For Transfer CFT you can monitor deployments of:

- Configurations – The state of the last deployment of the Transfer CFT configuration on an instance of Transfer CFT.
- Policies – The state of a Transfer CFT policy deployment on each Transfer CFT linked to the policy.
- Flows – The state of a flow deployment on each Transfer CFT the flow uses.
- Updates - The state of an update applied to a product.

Policies, flows and updates have global statuses not represented on the Deployment List page. For example, global statuses are listed in the Status columns on the Policy List and Flow List pages. In contrast, the specific status of a policy or flow for each linked Transfer CFT is displayed on the Deployment List page.

Transfer CFT configuration deployment monitoring

Only one table row is displayed for each Transfer CFT configuration on the Deployment List page. The deployment is reported when the instance of Transfer CFT is registered successfully in Central Governance.

The status of an instance of Transfer CFT is not the same as the deployment status of its configuration. For example, the status of an instance of Transfer CFT might be **Stopped** on the Product List page, but the status of its configuration might be **Deployed** on the Deployment List page.

When a user deploys a configuration for an instance of Transfer CFT, the deployment state, date and time, and status are updated for the Transfer CFT on the Deployment List page.

Changing a configuration affects reporting on the Deployment List page only when the changed configuration is deployed on Transfer CFT.

If the Transfer CFT is linked to a policy and the policy has not been deployed on it, the policy is deployed when a user deploys its configuration, and this is reported on the Deployment List page. Moreover, if a policy deployment fails (deployed in error status), the record associated with the policy on the Transfer CFT also is updated on the Deployment List page.

If Central Governance deploys a configuration to Transfer CFT and tries unsuccessfully to restart it, Central Governance rolls back to the last successfully deployed configuration. At the same time, Central Governance retains the new configuration that failed to deploy. This enables you to try again to redeploy the new configuration.

If Transfer CFT is stopped or a user deploys a configuration to it using the **Deploy, no restart** option, an invalid configuration is detected only when attempting to start the product. If invalid, a rollback is not performed and the product remains in Stopped status. Review the product's logs for clues to resolve the problem.

When an instance of Transfer CFT is removed from the Product List page, Central Governance deletes the record of its configuration deployment from the Deployment List page. If the removed Transfer CFT was linked to a policy, Central Governance also deletes those records from the page.

Transfer CFT policy deployment monitoring

When a policy is deployed, its configuration is deployed on all instances of Transfer CFT linked to the policy. Central Governance adds a table row on the Deployment List page for each Transfer CFT. Central Governance also updates the configuration deployment of each Transfer CFT affected by the deployed policy. For example, if the policy is deployed to three instances of Transfer CFT:

- A table row of type **Policy** is added for each of the three Transfer CFT targets.
- A table row of type **Configuration** is updated for each of the three Transfer CFT targets.

Changing a policy affects reporting on the Deployment List page only when the changed policy is deployed on Transfer CFT.

When linking a policy to a new instance of Transfer CFT, the page is updated when the policy is deployed on it.

When unlinking a policy from an instance of Transfer CFT, Central Governance deletes the row for the policy on the Transfer CFT target. When a policy is deleted, Central Governance deletes all table rows associated with it.

Flow deployment monitoring

When a flow is deployed, its configuration is deployed on all instances of registered products linked to the flow. A table row of type **Flow** is added on the Deployment List page for each affected product.

Changing a flow affects reporting on the page only when the changed flow is deployed on the product. For example, when an instance of Transfer CFT is added or removed as a source, target or relay in a flow, the changes are reported on the page when the changed flow is deployed or removed.

If Transfer CFTs or SecureTransports are removed as sources or targets in flows, the flow configuration also is removed from the affected product If the configuration on the removed product fails, the flow status becomes **Deployed in error**, and on the Deployment List page the target product has the status **Removed in error**.

When a flow is deleted, Central Governance deletes all table rows associated with it on the Deployment List page. If a flow is **Removed in error**, all deployment entries remain on the Deployment List page until the flow is removed successfully from all targets.

Product updates

Central Governance can apply updates remotely to products that have been registered successfully. Updates are files containing service packs, patches and version upgrades for products.

See *Product updates* on page 159 for details.

Predefined filters for deployment monitoring

Select a predefined filtered view by clicking a category on the left side of the Deployment List page. The default sorting is the most recent by date. The predefined views are:

- **All** is an unfiltered view of all deployments.
- **In Error (n)** is a view of all failed deployments with a status of **Deployed in error**, with **n** the number that failed. The error reason in the Message column is the same as the popup text for deployments with **Deployed in error** status in the All view.
- **Flows** is a view of all deployments of the type Flow.

- **Policies** is a view of all policy deployments.
- **Configurations** is a view of all configuration deployments.
- **Updates** is a view of updates applied to products.

The following describes the columns in the tables for monitoring deployments on the page. The columns are common to the predefined filtered views you can select on the left side of the page. You can click **Filter** to apply filters by column headings, date/time, target and so on.

Date/Time

The date and time when the deployment process was completed. However, for a deployment with a **Deploying** status, this is the starting date and time of the deployment.

Status

The most recent status of the deployment.

For an error status, place the cursor over the status to view a popup description of the error and reasons for failure. You can attempt to redeploy failed deployments or updates by selecting an item in this state and clicking **Retry**.

The Configurations view also can display a status of **Deployed, needs restart** for a product that needs to be restarted after its configuration was changed and deployed with the **Deploy, no restart** option. The option lets you restart a product when you want rather than at the same time a configuration is deployed. Restarting is required for the configuration changes to become effective on the product. Go to the Product List page and restart the product when you are ready. Select **Products** on the top toolbar to open the page.

Target

The name of the product where the configuration or update is deployed. Click the value to open the details page for the product.

Item

The name of the deployed item. Click the value to open the details page for it. The details page that is displayed depends on the value in the Type column.

Type

The type of deployment or update.

Message

For deployments with an error status, a message provides reasons for the failure.

Retry configurations, policies, flows, updates

You can redeploy a failed deployment, with the statuses **Deployed in error** and **Removed in error**, on the Deployment List page. This includes failed flows, policies, configurations and product updates. The retry action is available for deployments and updates with error statuses.

The status **Removed in error** identifies a failure to remove a flow configuration from a product. It can occur when a flow is removed from Central Governance or when a flow is deployed after a product was deleted from the flow. A flow is removed from Central Governance only when the flow configuration has been removed successfully from all the targeted products.

For failed deployments of type:

- Configuration, the action deploys the configuration again.
- Policy, the action redeploys the policy only on the selected instances of Transfer CFT. Redeploying a policy also redeploys the configuration of the associated Transfer CFT. See *Policies* on page 207.
- Flow, the action redeploys the flow only on the selected instances of products. See *Defining flows* on page 272.
- Update, the action tries again to install an update on the target products. See *Product updates* on page 159.

The following steps do not apply to products in the Configurations view that have a status of **Deployed, needs restart**. You need to restart such products for their most recently deployed configurations to become effective. Go to the Product List page and restart the products when you are ready. Select **Products** on the top toolbar to open the page.

1. Select **Administration > Deployments** to open the Deployment List page.
2. Select one or more table rows with a status of **Deployed in error** or **Removed in error**. For failed product updates, the status is **Uploaded in error** or **Updated in error**.
3. Click **Retry**.

   Note that the retry action on a failed deployment does not re-deploy everything, just the selected part. For deployments, the status changes to **Deploying** before changing to **Deployed** for a successful deployment.

   For failed flow removals, the status changes to **Removing**. The target product is removed from the Deployment List page when the removal succeeds. If the flow removal triggered the initial action and there are no more targets to remove, the flow also is removed from Central Governance.

   For product updates, **Updated** indicates success.

Appendix A: Transfer CFT capacity planning

The following topics provide guidelines to help you perform capacity planning for implementing Transfer CFTs in a Central Governance environment. Also provided are results of Axway performance tests for Central Governance when managing Transfer CFTs.

The guidelines are based on results of specific performance and scalability tests of Central Governance and on knowledge gained during testing and development of Central Governance. Many untested factors, including network performance and various workload characteristics, can affect the performance of a production deployment. Use this documentation to plan your deployment, but it is essential to test your plan to verify it handles your workload and satisfies your performance objectives.

Planning steps

The following are recommended high-level steps for Transfer CFT capacity planning in a Central Governance environment.

1. Identify and document your Transfer CFT workload and topology.
2. Determine your performance objectives.
3. Relate your workload, topology and objectives to the characteristics of the general guidelines in this capacity planning documentation.
4. Develop a deployment plan based on the guidelines and test results.
5. Implement a Central Governance test environment and load-test it to validate your deployment plan.

Performance factors

Axway has determined key factors for Central Governance performance. The following describes these performance factors and their roles in capacity planning.

For capacity planning with Central Governance you need to:

- Know the number of Transfer CFTs.
- Analyze the complexities of the Transfer CFT flows, or topologies, that Central Governance manages.
- Determine your performance objectives.

You can then relate your requirements and objectives to the Axway capacity planning guidelines and performance test results.

Workload characteristics

You can determine or estimate workload characteristics by measuring your current workload and projecting future workload based on plans for business services that use Central Governance.

Factors to define or estimate include:

Number of Transfer CFTs

Each Transfer CFT is an object in the Central Governance database. The object requires memory for storage and processor time for status updates. Memory and processor time also are required when deploying Transfer CFT configuration.

Complexity of flows

The logical link between a source and destination Transfer CFT is defined as a path. Each path in a data flow is an object in the Central Governance database. The object requires memory for storage and consumes processor and network resources during deployment.

The complexity of your flows is a function of the number of paths. This affects the performance of Central Governance when deploying the flows on Transfer CFTs. To accurately plan the capacity of your Central Governance deployment, you must characterize your organizational needs and create Transfer CFT flows that provide the desired balance of performance, scalability and maintainability.

Performance objectives

Your performance objectives are defined by the level of service you provide to your business partners. Business requirements typically drive service objectives.

Use your performance objectives to evaluate your capacity plan. You must define these objectives to have the criteria for determining whether your capacity plan is sufficient.

Performance objectives can include:

Transfer CFT flow configuration deployment duration

The time required to deploy a flow configuration can be affected by the number of the Transfer CFTs included in the flow and the number of paths in the flow. Complex flows with many paths might take longer to deploy when compared to simple flows with fewer paths.

Transfer CFT policy deployment duration

The time required to deploy policy configuration can be affected by the number of Transfer CFTs managed by Central Governance.

Planning guidelines

The following are guidelines for capacity planning. These are based on Axway's performance and scalability tests of Central Governance. See *Performance benchmarks* on page 421 for results of the Axway tests. Use these guidelines to develop your own capacity plan based on your processing requirements and deployment infrastructure.

Performance of Central Governance is affected by:

- Processing power of servers
- Transfer CFT topology

Transfer CFT topology is embodied in flows in a Central Governance environment. Axway tested a type of flow to collect performance characteristics and establish benchmarks. The corporate-to-store flow mirrors a common use case. Note that flows with the fewest paths result in faster performance and greater scalability.

A common corporate-to-store flow involves:

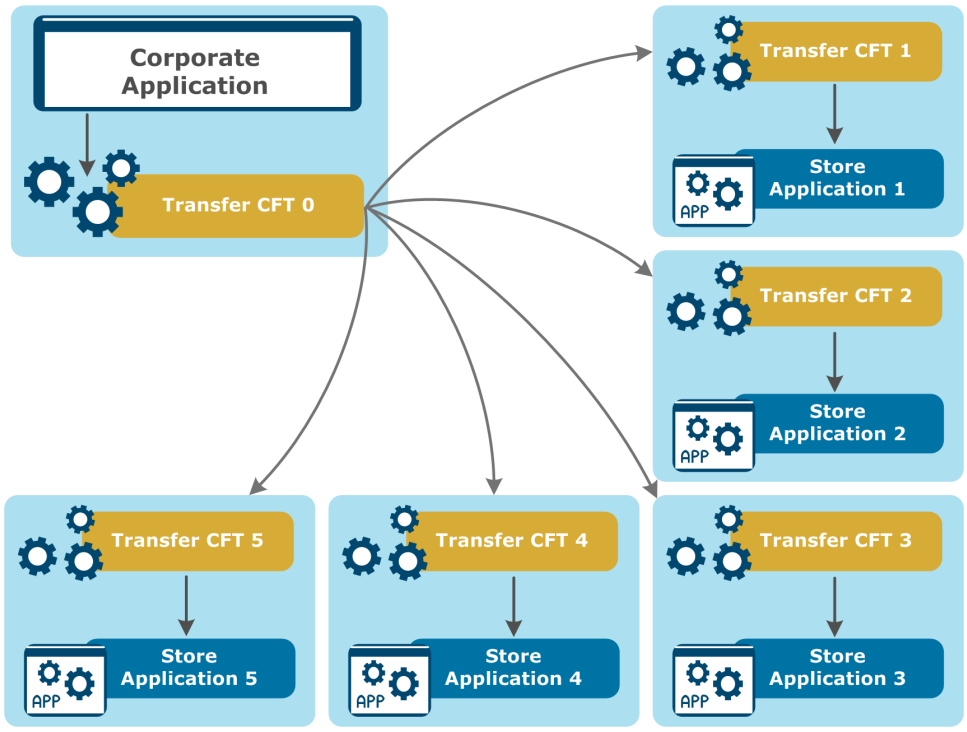
- Stores sending daily sales and inventory data to corporate headquarters
- Headquarters sending daily updated prices lists to stores

Details of such data exchanges are:

- At headquarters, a corporate application is integrated with a single instance of Transfer CFT
- The corporate application can exchange data, via Transfer CFT, with all stores
- Each store has a single instance of Transfer CFT on premise
- The stores can exchange data with the corporate application
- The stores cannot exchange data among themselves

This corporate-to-store flow is scalable because the number paths for *n* stores is *n*.

The following graphic illustrates a simple corporate-to-store flow.



Performance benchmarks

Axway executes test scenarios to evaluate performance characteristics and establish benchmarks for Central Governance. The scenarios attempt to replicate common and extreme use cases. These are starting points for performance planning and cannot substitute for thorough performance evaluation of your environment.

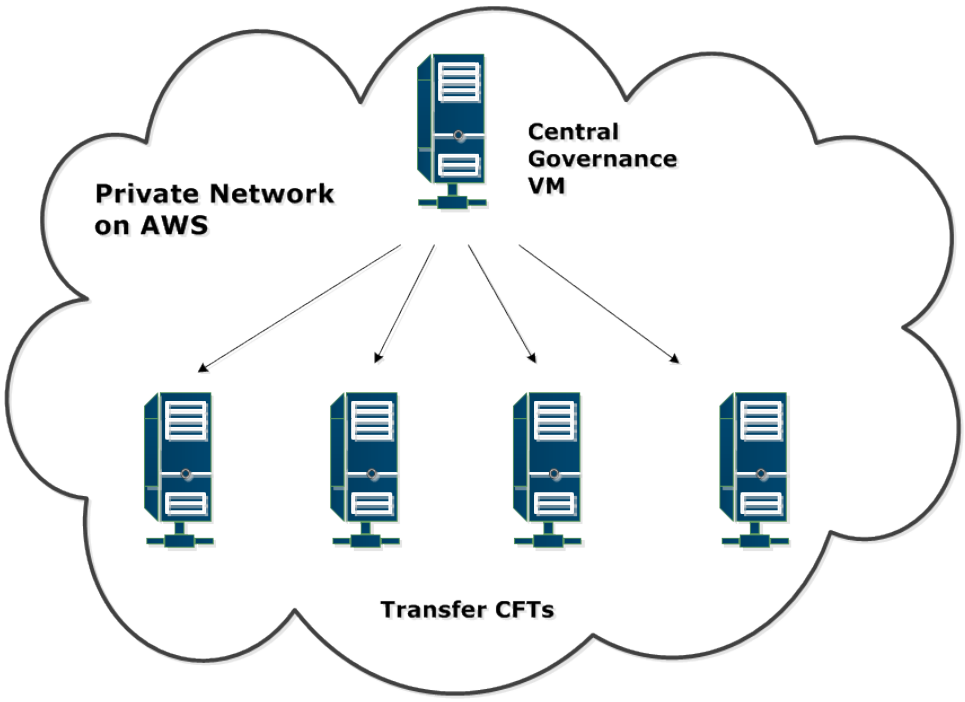
Test environment

The test environment for executing performance scenarios is managed in an isolated virtual environment.

All virtual machines in the test environment communicate on a private network. This ensures collected performance benchmarks reflect the performance of Central Governance, unaffected by the state of the network. This is an important consideration when evaluating performance in real-world installations.

All tests were executed on the cloud via Amazon Web Services (AWS) for scalability. Amazon Elastic Compute Cloud (Amazon EC2) c3.4xlarge was used. See http://aws.amazon.com/ec2/instance-types/ for more info about EC2 instances.

The following illustrates the test environment.



Central Governance

Central Governance is installed on a virtual machine with the following specifications.

Specification	Value
Number of virtual CPUs	15
CPU frequency	15 GHz
Memory	30 GB
Storage	320 GB solid-state drive (SSD)

The following software is running on the VM.

- Red Hat Enterprise Linux 6.5 operating system
- JRE 1.8 update 40

In addition, the ulimit environment variables on each VM are:

- ulimit -n 4096 (number of opened files)
- ulimit -s unlimited (stack size)
- ulimit –u unlimited (number of processes)

Transfer CFT

Transfer CFTs 3.1.3 are installed on virtual machines. Up to 150 Transfer CFTs are on each VM.

Test scenarios

The following describes the performance test scenarios and benchmarks.

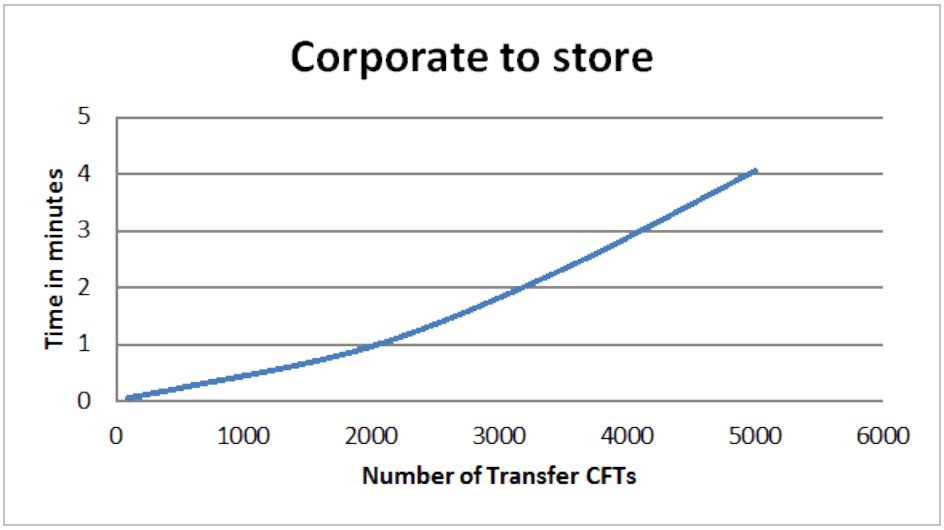
Transfer CFT flow configuration deployments

Flow deployment duration is the time required to deploy the configuration of flows. The Transfer CFTs are registered in Central Governance with a stopped status. The flows are created and saved in Central Governance, but not deployed. The representational state transfer application programming interface (REST API) is used to deploy the flow configuration on the Transfer CFTs. The time between issuing the deploy command and when the flow has a deployed status is recorded as the deployment duration.

The following table shows the results of deploying flows. Times are minutes. All durations have ±1 second margin of error. The Transfer CFTs are not started.

Number of Transfer CFTs	Minutes
100	0.06
500	0.23
2000	0.96
3500	2.32
5000	4.05

The following graph illustrates the scalability of flow deployments by comparing deployment time for different numbers of Transfer CFTs.



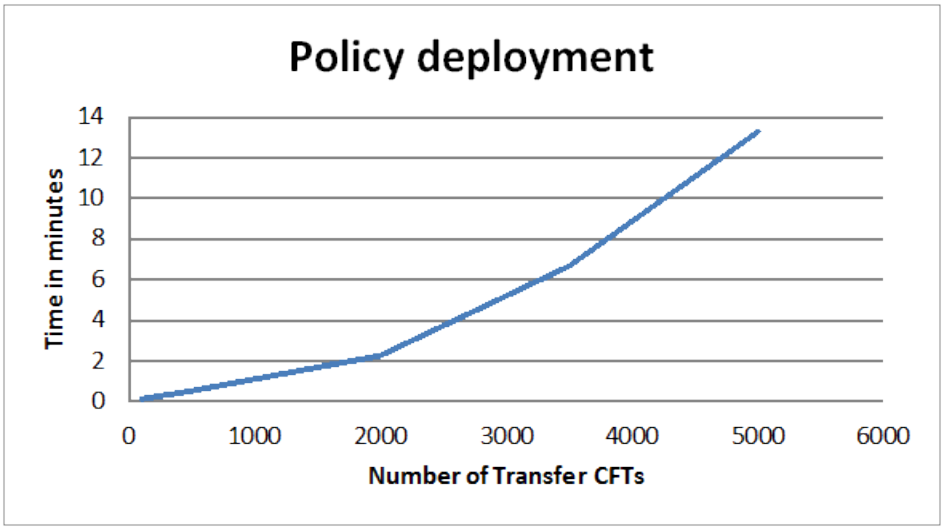
Transfer CFT policy configuration deployment

Policy deployment duration is the time required to deploy the configuration to Transfer CFTs. The Transfer CFTs are registered in Central Governance with a stopped status. The policy is created and saved in Central Governance, but not deployed. The web API is used to deploy the policy configuration on the Transfer CFTs. The time between issuing the deploy command and when the policy has a deployed status is recorded as the deployment duration.

The following table and chart show the results of deploying a policy to Transfer CFTs. Times are minutes. All durations have ±1 second margin of error. The Transfer CFTs are not started.

Number of Transfer CFTs *	Minutes
100	0.09
500	0.50
2000	2.21
3500	6.63
5000	13.30

* Transfer CFTs are not started.



Recommendations

The following table provides recommendations for RAM in gigabytes for Central Governance based on the number of Transfer CFTs it manages.

Number of Transfer CFTs	RAM GB
Up to 250	3
251 to 500	5
501 to 750	7
751 to 1,000	9
1,000 to 3,500	15
More than 3,500	30

The following ulimit settings are required for Linux:

- ulimit -n 4096 (number of opened files)
- ulimit -s unlimited (stack size)
- ulimit –u 65536 or higher (number of processes)

When trying to install Central Governance, the installer exits when these requirements are unmet.

Appendix B: Transfer CFT corresponding parameters

The following topics have tables that list and describe fields in Central Governance that correspond to parameters in Transfer CFT.

Transfer CFT configuration in Central Governance and CFTUTIL

The following tables describe the Transfer CFT fields that can be configured in the Central Governance user interface and the corresponding Transfer CFT CFTUTIL parameters. Default Central Governance values are underlined.

See Transfer CFT command list and syntax in the Transfer CFT user guide for a list of Transfer CFT commands, syntax and parameters.

Network > protocols

CG field	CG values	CFTUTIL parameter	Description
Interface	Any | Any IPv4 | Any IPv6 | Address	CFTNET - HOST = INADDR_ANY | IN6ADDR_ANY | IN4ADDR_ANY | string	Networking IP address of the local resource (an entity through which connections can be established).

CG field	CG values	CFTUTIL parameter	Description
Network protocol	Enable | Disable	CFTPARM - NET (network)	Shows whether the corresponding network protocol is active. Once a network protocol section is available in Central Governance, a NET object of type TCP is created on Transfer CFT with: ID=<name of the network protocol:TCP/PTCP or UDT>1 CLASS= the class corresponding to the selected network type: • for TCP, class =1 • for UDT, class=2 • for pTCP, class=3 If the network protocol is enabled, the CFTNET object is referenced in the list of active networks of the current CFTPARM and it is taken into account during transfers execution. UDT and pTCP are not supported on Transfer CFTs running on z/OS and IBM i computers.
Ports out	5000-65535 (5000-65535)	CFTNET - SRCPORTS = (6000-6009,6010-6019,6020-6030)	Definition of outgoing port ranges. The list of ports can contain up to 16 ranges.
Connections	1-2000 (128)	CFTNET - MAXCNX	The maximum number of simultaneous connections that Transfer CFT accepts to establish on a given network resource.
PeSIT / SSL	No security | SSL _ DEFAULT	CFTPROT - SSL	The security profile for a given protocol definition.
Protocol	Enable | Disable	CFTPARM - PROT ('prot' list)	Once a protocol section is available in Central Governance, a PROT object of type PeSIT is created on Transfer CFT with: ID=<ID_CFTNET>+<if No security:1;2> NET=<ID_CFTNET> If the protocol is enabled, the CFPROT object is referenced in the list of active protocols of the current CFTPARM and it is taken into account during transfers execution.
Port in	1025-65535	CFTPROT - SAP	Number of the local monitoring port on which the monitor can be called.

Network > general

CG field	CG values	CFTUTIL parameter	Description
Maximum simultaneous transfers	Linux and Windows: 2-1000 (128) z/OS and OS/400: 2-990 (128)	CFTPARM - MAXTRANS	The maximum number of simultaneous connections that Transfer CFT accepts to establish for a network resource.
Disconnect timeout	0-3600 (60)	CFTPROT - DISCTR, DISCTC	The wait timeout for either a response to the protocol connection request or to the partner in the connection, before disconnecting.
Attempts to restart transfer	0-32767 (5)	CFTPROT - RESTART	The maximum number of times that Transfer CFT will attempt to restart a transfer.
IPv6 mode	Client | Server | Both | None	UCONF - ipv6.disable_connect, ipv6.disable_listen	IPv6 resolution for host names when Transfer CFT is acting as a client, a server, both, or none.

Network > general > keep alive between transfers

CG field	CG values	CFTUTIL parameter	Description
Client	0-3600 (10)	CFTPROT - DISCTD	The time to keep the session active between transfer activity on the client.
Server	0-3600 (60)	CFTPROT - DISCTS	The time to keep the session active between transfer activity on the server.

Network > pTCP

CG field	CG values	CFTUTIL parameter	Description
Number of parallel connections	1-1024 (10)	UCONF - acceleration.ptcp.<netid>.nb_connections	The maximum number of striped connections.

CG field	CG values	CFTUTIL parameter	Description
Packet size	(3000)	UCONF - acceleration.ptcp.<netid>.packet_size	pTCP packet size in bytes
Buffer size	(10)	UCONF - acceleration.ptcp.<netid>.buffer_size	Internal acceleration buffer size in MB.

Network > UDT

CG field	CG values	CFTUTIL parameter	Description
Buffer size	(10)	UCONF - acceleration.udt.<netid>.buffer_ size	Internal acceleration buffer size in MB.

Network > PeSIT tuning > transmission

CG field	CG values	CFTUTIL parameter	Description
Compression	Yes | No	CFTPROT - SCOMP, RCOMP Yes - 15 No - 0	Use compression on file transfers.
Inactivity timeout	0-3600 (60)	CFTPROT - rto	Network monitoring timeout in seconds, excluding the protocol connection/disconnection/break phase. 0 means infinite.

Network > PeSIT tuning > synchronization

CG field	CG values	CFTUTIL parameter	Description
Acknowledgment window size	0-16 (3)	CFTPROT - SCHKW, RCHKW	The window size setting the number of sync points that can occur.
Data transferred between sync points	0-32767 (32767)	CFTPROT - SPACING, RPACING	The number of KB transferred between sync points.

Bandwidth allocation

CG field	CG values	CFTUTIL parameter	Description
Enable	Yes | No	UCONF - cft.server.bandwidth.enable	Manage data rates and the network bandwidth used for incoming and outgoing data in your flows.
Global data rate	Unlimited | Limited	UCONF - cft.server.bandwidth.cos.0.max_rate_in, cft.server.bandwidth.cos.0.max_rate_out	Specifies limits on the rates of incoming and outgoing data. Allows setting, in kilobytes per second, the maximum for the rates of incoming and outgoing data.
Maximum incoming (global data rates is limited)		UCONF - cft.server.bandwidth.cos.0.max_rate_in	The maximum limit for incoming data transfer rates.
Maximum outgoing (global data rates is limited)		UCONF - cft.server.bandwidth.cos.0.max_rate_out	The maximum limit for outgoing data transfer rates.

Bandwidth allocation > priority

Bandwidth allocation per priority level.

CG field	CG values	CFTUTIL parameter
High	80%	UCONF - cft.server.bandwidth.cos.1.weight_in cft.server.bandwidth.cos.1.weight_out
Medium	15%	UCONF - cft.server.bandwidth.cos.2.weight_in cft.server.bandwidth.cos.2.weight_out
Low	5%	UCONF - cft.server.bandwidth.cos.3.weight_in cft.server.bandwidth.cos.3.weight_out

Transfer processing

CG field	CG values	CFTUTIL parameter	Description
User for file access	Transfer CFT system account | USERID variable	CFTPARM - USERCTRL = NO | YES	Specifies the account that is used to read/write files transferred.
User for script execution	Transfer CFT system account | USERID variable	UCONF - cft.server.exec_as_user = NO | YES	Specifies the account that is used to execute scripts. This parameter is not supported on Transfer CFTs running on z/OS and IBM i computers.
For an unknown flow	Use the system default | Reject request	UCONF - cft.default_idf.enable = YES | NO	The action to take if the flow is unknown.
Transmit files individually	Always | When necessary	UCONF - cft.server.force_heterogeneous_mode = YES | NO	Whether the transmission of a group of files is done by individual file, or grouped when possible.
When requesting all files	Stop on error | Continue	CFTPARM - RCVALLER = STOP (Stop on error) | CONTINUE (Continue)	The action to take if any of the transfers fail.

Transfer processing > default scripts > source | target

Acknowledgment is only for source.

CG field	CFTUTIL parameter	Description
Post-processing	CFTPARM – EXECSF | EXECRF	The file to execute after the file is sent | received.
Acknowledgment	CFTPARM - EXECSFA	The file to execute after an acknowledgement is received for a sent file.
Error	CFTPARM – EXECSE | EXECRE	The file to execute after an error occurs during a file transfer

Transfer request mode > asynchronous

CG field	CG values	CFTUTIL parameter	Description
Time between scans	1-6 (60)	CFTCOM - TYPE = FILE - WSCAN	Interval in seconds to scan the transfer request communication file.

Transfer request mode > synchronous

CG field	CG values	CFTUTIL parameter	Description
Enable	Yes | No	CFTCOM - TYPE = TCPIP	Add a communication media of type synchronous.
Host	string, max 64, 127.0.0.1 by default	CFTCOM - HOST	The host receiving commands. If HOST is updated at deployment, ADDRLIST is set to the empty value.
Port	1025-65535 (1765)	CFTCOM - PORT	The port to receive commands on.
Maximum connections	1-1024 (256)	UCONF - cft.server.cftcoms.max_ connection	Number of connections for the media communication.
Secured connections	Enable | Disable	CFTCOM - PROTOCOL = XHTTP (Disable) | XHTTPS (Enable)	Use security for the request/reply protocol on the network.
Session timeout	0-86400 seconds or 0-1440 minutes (60 seconds)	CFTCOM - DISCTS	Interval in seconds or minutes the Transfer CFT waits before closing an idle connection.

Transfer list

CG field	CG values	CFTUTIL parameter	Description
Number of entries in memory	1-32000 (1000)	UCONF - cft.server.catalog.cache_size	The maximum number of entries in the memory buffer.

CG field	CG values	CFTUTIL parameter	Description
Update during transfer	Enabled | Disabled	CFTCAT - UPDAT = 0 (Disabled)| 1..32767 (Enabled)	Update the transfer list while a transfer is occurring
(Update during transfer is Enabled)	1-32000 (1) sync points between updates	CFTCAT - UPDAT	The number of synchronization points that occur during a transfer before updating the transfer list.
Synchronize list file when written	Yes | No	UCONF - cft.server.catalog.sync.enable	Force the transfer list file to synchronize when Transfer CFT processes write to it.

Transfer list > entry retention

CG field	CG values	CFTUTIL parameter	Description
Purge	Automatic | Manual	UCONF - cft.purge.periodicity	Periodically remove older entries in the transfer list. The transfer list will grow indefinitely if manual purge is selected and start-up purge is disabled.
Purge is manual		UCONF - cft.purge.periodicity,value=0	
Purge is automatic	1-999 (1) days between purges	UCONF - cft.purge.periodicity	
Purge at startup	Yes | No	UCONF - cft.purge.enable_ on_start	Defines if the system should purge the transfer list at start up.
Purge increment	10 (minimum 10)	UCONF - cft.purge.quantity	Defines how many transfers to delete from the transfer list file per step.
Keep aborted transfers	Yes | No	CFTCAT - RKERROR (keep | delete)	Automatically delete aborted transfers without waiting for a purge.

Transfer list > entry retention > retention period

The period is expressed in number of days, hours or minutes.

CG field	CG values	CFTUTIL parameter	Description
Completed incoming transfers	1-999 (10	UCONF -cft.purge.rx cft.purge.rt	Period after which the entries of incoming transfers that were successfully executed are purged.
Incomplete incoming transfers	1-999 (10	UCONF - cft.purge.rh	Period after which the entries of uncompleted incoming transfers are purged.
Completed outgoing transfers	1-999 (10	UCONF - cft.purge.sx cft.purge.st	Period after which the entries of outgoing transfers that were successfully executed are purged.
Incomplete outgoing transfers	1-999 (10	UCONF - cft.purge.sh	Period after which the entries of uncompleted outgoing transfers are purged.

CRONJOBs

CG field	CG value	CFTUTIL parameter	Description
Name	string, max 32, empty by default	CFTCRON, id	
Status	Active / Inactive	CFTCRON, state	To activate CRONJOB, State should be ACTIVE. To disable a CRONJOB, State should be NOACTIVE.
Description	string, max 80, empty by default	CFTCRON, comment	Free comment. This comment is displayed and can be used to indicate a specific item of information (e.g. customer name, etc.)
Filename	string, max 512, empty by default	CFTCRON, exec	Upload / Specify the script to be executed.
Schedule	string, max 512, empty by default	CFTCRON, time	*CRONJOB schedule syntax* on page 185

CG field	CG value	CFTUTIL parameter	Description
User ID	string, max 32, empty by default	CFTCRON, userid	The user for this job procedure.
Additional Information	string, max 512, empty by default	CFTCRON, parm	The PARM to be used in the job execution.

Access and security > access management

CG field	CG values	CFTUTIL parameter	Description
Access type	None | Transfer CFT internal | Central Governance	am.type = none (None) | internal (Transfer CFT internal) | passport (Central Governance)	Type of access management to use in Transfer CFT.
Create process as user	Yes | No	copilot.misc.createprocessasuser	Specifies whether Transfer CFT Copilot user must have system rights.

Access type set to Central Governance

CG field	CG values	CFTUTIL parameter	Description
Organization	<any organization from Central Governance>	am.passport.cg.organization	Central Governance organization with users who can operate Transfer CFT.
Superusers		am.passport.superuser	List of users separated by comma with unlimited privileges on Transfer CFT.
Check permission for transfer execution	Yes | No	am.passport.userctrl.check_ permissions_on_transfer_ execution	Check whether the user has permissions to execute transfers.

Access type set to Transfer CFT internal

CG field	CG values	CFTUTIL parameter	Description
Group database	SYSTEM | XFBADM	am.internal.group_database	Type of database where group members are defined. On Windows, only SYSTEM group database is supported. This parameter is not supported on Transfer CFTs running on z/OS and IBM i computers.
Admin		am.internal.role.admin	List of groups mapped to the administrator role.
Application		am.internal.role.application	List of groups mapped to the application role.
Designer		am.internal.role.designer	List of groups mapped to the designer role.
Helpdesk		am.internal.role.helpdesk	List of groups mapped to the helpdesk role.
Partner manager		am.internal.role.partnermanager	List of groups mapped to the partner manager role.

Access and security > security > FIPS

CG field	CG values	CFTUTIL parameter	Description
Enable	Yes | No	cft.fips.enable_compliance	Activate FIPS security.

Visibility

CG field	CG values	CFTUTIL parameter
Enable	Yes | No	UCONF - sentinel.xfb.enable

Visibility > servers

CG field	CG values	CFTUTIL parameter	Description
Main server	Internal | External	sentinel.trkipaddr = <CG visibility host> sentinel.trkipport = <CG visibility server port>	
Host (Main server is External)		UCONF - sentinel.trkipaddr	Host for the main Sentinel server.
Port (Main server is External)	1-65535 (1305)	UCONF - sentinel.trkipport	Port for the main Sentinel server.
Backup server	Internal | External | None	UCONF - sentinel.*	
Host (Backup server is External)		UCONF - sentinel.trkipaddr_bkup	Host for the backup Sentinel server.
Port (Backup server is External)	1-65535 (1305)	UCONF - sentinel.trkipport_bkup	Port for the backup Sentinel server.

Visibility > events

CG field	CG values	CFTUTIL parameter	Description
Transfer steps reported	All | First and last | None	UCONF - sentinel.xfb.transfer = ALL (All) sentinel.xfb.transfer = SUMMARY (first and last) sentinel.xfb.transfer = NO (None)	Level of detail for message content.
Transfer status frequency (transfer steps reported is All)	Every 60 seconds	UCONF - sentinel.xfb.transfer_progress_ period in seconds	Specify how often the transfer status is updated in seconds or minutes.
Minimum log level	Error | Fatal | Warning | Info | No log events	UCONF - sentinel.xfb.log = EF (Error) | F (Fatal) | WEF (Warning) | IWEF (Info) | empty (no logs)	Minimum severity level of the messages to display.

CG field	CG values	CFTUTIL parameter	Description
Buffer capacity	10000	UCONF - sentinel.xfb.buffer_size (in number of messages)	Maximum number of messages in Sentinel buffer
When buffer is full	Drop new messages | Shut down	UCONF - sentinel.xfb.shut sentinel.xfb.shut = 0 => Drop new messages sentinel.xfb.shut = 95 => Shut down	Discard messages that exceed the buffer capacity, or shut down Transfer CFT when the visibility buffer is full

Logging

CG field	CG values	CFTUTIL parameter	Description
Entry size	Linux, Windows and z/OS: 28-1024 (160) bytes OS/400: 28-256 (160) bytes	CFTLOG - length	Size of each entry in the logging file in bytes.
Timestamp precision	1 second | 10 ms | 100 ms	UCONF - cft.cftlog.time_ precision	The preciseness of the time displayed in the log, in seconds, 10 milliseconds or 100 milliseconds.

Logging > file rotation

CG field	CG values	CFTUTIL parameter	Description
Number of files in rotation	1-999 (3)	UCONF - cft.cftlog.backup_ count	Number of log files used in the rotation process. This parameter is not supported on Transfer CFTs running on z/OS computers.
Daily rotation time	00:00:00 daily time	CFTLOG - switch	Time of day to rotate files.
Rotate based on size	Yes | No	CFTLOG - maxrec = 0 (No) | 1..9999 KB (Yes)	Rotate log files when they reach a specific size.

CG field	CG values	CFTUTIL parameter	Description
(Rotate based on size is Yes)	Every 1-9999 (1024) KB	CFTLOG - maxrec	The size based on which the file rotates.
Rotate on stop	Yes | No	UCONF: cft.cftlog.switch_on_ stop	Rotate files when Transfer CFT stops.

Folder monitoring

CG field	CG value	CFTUTIL parameter	Description
Enable	Yes | No	UCONF - folder_monitoring.enable = NO | YES	Specifies whether folder monitoring is enabled.
Folder name	string max 100, <empty>	UCONF - folder_monitoring.folders =1,2,... (each index corresponds to a Folder name in Central Governance)	Specifies the friendly name of each folder monitoring as identified in Central Governance. For each Folder name, an incremented index is added in the folders list. The index <i> is referred for all parameters under a given folder monitoring instance.
Directory to scan	String max 512, <empty>	UCONF - folder_ monitoring.folders.<i>.scan_dir	Path to the top-level directory to monitor.
Directory where files are tracked	String max 512, <empty>	UCONF - folder_ monitoring.folders.<i>.work_dir	Path to the top-level directory for transferred files.
Flow identifier	First sub-folder | Second sub-folder | Custom	UCONF - folder_ monitoring.folders.<i>.idf = <CFTSEND ID> (0) - First sub-folder (1) - Second sub-folder <value> - Custom	Identifier of the flow used in the transfer.

CG field	CG value	CFTUTIL parameter	Description
Partner	First sub-folder | Second sub-folder | Custom	UCONF - folder_ monitoring.folders.<i>.part = <CFTPART ID> (0) - First sub-folder (1) - Second sub-folder <value> - Custom	Partner who receives the file.
Scan sub-directories	Yes | No	UCONF - folder_ monitoring.folders.<i>.enable_ subdir = YES | NO	Monitor the directory tree starting with the top-level directory.
Number of files to scan	Unlimited | Limited	UCONF - folder_ monitoring.folders.<i>.file_count = -1	Whether the transmission of a group of files is done by individual file or grouped when possible.
(Number of files to scan is limited)	1 – 2147483647 (100)	UCONF - folder_ monitoring.folders.<i>.file_count	Maximum number of files to scan for submission.
Time between scans	1 - 3600 (60 seconds)	UCONF - folder_ monitoring.folders.<i>.interval	Seconds between scans.
Time before scanned files are submitted	0 – 3600 (5 seconds)	UCONF - folder_ monitoring.folders.<i>.file_idle_ delay	Files that have not changed during this interval can be submitted.
Method	Move | File	UCONF - folder_ monitoring.folders.<i>.method	Specifies whether submitted files are moved or kept in the scan directory and tracked with a state file.
Append timestamp to submitted files (Method is Move)	Yes | No	UCONF - folder_ monitoring.folders.<i>.renaming_ method Yes – TIMESTAMP No - NONE	After the submission, the file is moved and renamed by appending the timestamp.

CG field	CG value	CFTUTIL parameter	Description
Resubmit changed files (Method is File)	Yes | No	UCONF - folder_ monitoring.folders.<i>.resubmit_ changed_file = YES | NO	Specifies whether a file is submitted again if a change is detected.
Include file template	String max 1024, <empty>	UCONF - folder_ monitoring.folders.<i>.file_ include_filter	Only files matching this pattern are monitored.
Exclude file template	String max 1024, <empty>	UCONF - folder_ monitoring.folders.<i>.file_ exclude_filter	Files matching this pattern are excluded.
Minimum size	Unlimited | Limited	UCONF - folder_ monitoring.folders.<i>.file_size_ min = -1 (Unlimited)	No minimum size limit for files to be submitted.
(Minimum size is limited)	1 - 2147483647 (1024)	UCONF - folder_ monitoring.folders.<i>.file_size_ min	Minimum size of files that can be submitted.
Maximum size	Unlimited | Limited	UCONF - folder_ monitoring.folders.<i>.file_size_ max = -1 (Unlimited)	No maximum size limit for files to be submitted.
(Maximum size is limited)	1 - 2147483647 (1024)	UCONF - folder_ monitoring.folders.<i>.file_size_ max	Maximum size of files that can be submitted.

Transfer CFT legacy flows in Central Governance and CFTUTIL

The following tables describe the Transfer CFT fields that can be configured in the Central Governance legacy flows user interface and the corresponding Transfer CFT CFTUTIL parameters.

Distribution list

CG field	CG value	CFTUTIL parameter	Description
Name	String, max 32 c.	CFTDEST, ID	Identifier of the distribution list.
Type -> Partner list	200 partners, each one has maximum 32 c.	CFTDEST, PART	Explicit list of the identifiers of the various partners.
Type -> File -> Use existing file	string, max 512	CFTDEST, FNAME	Name of the file containing the list of partner identifiers. The file already exists on Transfer CFT.
Type -> File -> Upload new file	file not empty, filename max 512,	CFTDEST, FNAME	Name of the file to upload on Transfer CFT containing the list of partner identifiers.
Unknown partner	Cancel (default) ; Continue ; Ignore	CFTDEST, NOPART Cancel -> Abort ; Continue-> Continue; Ignore-> Ignore	Select the action to perform when one of the partners is not defined.
Processing scripts submission -> Apply pre-processing	On main request (default) ; For each target in the list ; Both	CFTDEST, EXECPRE On main request -> DEST ; For each target in the list -> CHILDREN ; Both -> PART	Preprocessing procedure submit mode type: On main request – Executes the script only on the main request. For each target in the list – Executes the script only for each target in the list. Both – Executes the script both for the main request and for each target in the list.
Processing scripts submission -> Apply post-processing	On main request (default) ; For each target in the list ; Both	CFTDEST, EXEC On main request -> DEST ; For each target in the list -> CHILDREN ; Both -> PART	End of transfer procedure submit mode type: On main request – Executes the script only on the main request. For each target in the list – Executes the script only for each target in the list. Both – Executes the script both for the main request and for each target in the list.

CG field	CG value	CFTUTIL parameter	Description
Processing scripts submission -> Apply acknowledgement processing	On main request (default) ; For each target in the list ; Both	CFTDEST, EXECA On main request -> DEST ; For each target in the list -> CHILDREN ; Both -> PART	Acknowledgment procedure submit mode type: On main request – Executes the script only on the main request. For each target in the list – Executes the script only for each target in the list. Both – Executes the script both for the main request and for each target in the list.

Partners

CG field	CG values	CFTUTIL parameter	Description
Name	String, 32	CFTPART.ID CFTTCP.ID	
Description	String, 80	CFTPART.COMMENT	
Partner Access - Relay	String, 32	CFTPART.IPART If IPART is not empty, IMAXTIME = 0	
Partner Access -- Local partner name	String, 24	CFTPART.NSPART	Represents the identifier the Transfer CFT uses to identify itself to a partner.
Partner Access -- Local partner password	String, 8	CFTPART.NSPASSW	
Partner Access – Remote partner name	String, 24	CFTPART.NRPART	Represents the identifier of the remote partner Transfer CFT.
Partner Access – Remote partner password	String, 8	CFTPART.NRPASSW	

CG field	CG values	CFTUTIL parameter	Description
Partner Access – Host	String : a list of entries separated by comma. Maximum 4 entries. Each entry can have maximum 50 characters.	CFTTCP.HOST	
Partner Access – Operating system	List with values. See *Operating systems and deployment correspondence* on page 483.	CFTPART.SYST	
Protocol – Security profile	String, 32	CFTPART.SSL	
Protocol – Protocol		CFTPART.CFTPROTID	
Protocol – Port in	STRING LIST max_length=32 max_entries=4	CFTPART.SAP	
Protocol > Network sessions > Incoming connections	Unix, Windows, IBM i: 0-1000 z/OS: 0 - 990	CFTTCP.CNXIN	Maximum number of sessions for input connections
Protocol > Network sessions > Outgoing connections	Unix, Windows, IBM i: 0-1000 z/OS: 0 - 990	CFTTCP.CNXOUT	Maximum number of sessions for output connections
Protocol > Network sessions > Total connections	Unix, Windows, IBM i: 0-1000 z/OS: 0 - 990	CFTTCP.CNXINOUT	Maximum number of communication sessions
Hidden field	n/a	CFTTCP.CLASS	The network protocol selected. 1 - TCP 2 - UDT 3 - pTCP

Send template > transfer properties

CG field	CG values	CFTUTIL parameter	Description
Transfer priority	LOW, MEDIUM, HIGH, CUSTOM	CFTSEND, pri LOW ->0 MEDIUM->128 HIGH-> 255 CUSTOM-> integer between 0...255	Transfer priorities are equivalent to integer values ranging from 0 (low) to 255 (high). When Transfer CFT reaches the maximum number of transfers allowed, it queues transfers. When an ongoing transfer is finished and a slot is available for a new transfer, the system selects the one with the highest priority.
Bandwidth allocation	LOW, MEDIUM, HIGH	CFTSEND, cos LOW ->3 MEDIUM->2 HIGH-> 1	The amount of bandwidth allocated to this flow. The value you select determines the data transfer rate for this flow.
Transfer state	Ready, On hold, Kept	CFTSEND, state Ready (D) -> DISP On hold (H) ->HOLD Kept (K) ->KEEP	Indicates the state of the transfer request.: Ready, On Hold, Kept. Field is available in UI only if the Initiator is the source. If Target is the initiator, in source side the transfer state is ready and the field cannot be configured in Central Governance UI.
User id	string, max 32, empty by default	CFTSEND, userid	Identifier of the transfer owner. If this parameter is not defined, its default value is the system userid of the Transfer CFT.
Detect duplicate transfers	string max 512, empty by default	CFTSEND, duplicat	This field is used in detecting duplicate transfers and may contain a list of symbolic variables separated by a period ".".
Compress file	Yes, No	CFTSEND, ncomp Yes → 15 No → 0	Indicates whether files are compressed before they are transferred.
On file modification	Continue transfer, Stop transfer	CFTSEND , fdisp Continue transfer -> SHR Stop transfer-> CHECK	Available only in send template. Specify what happens if files are modified during the transfer.

CG field	CG values	CFTUTIL parameter	Description
Action after transfer	Delete file , Delete file content, None	CFTSEND, faction Delete file -> Delete Delete file content -> Erase None-> None: default	Specifies what happens to the file in source side when the transfer is complete. Delete file – Deletes the file. Delete file content – Removes the contents of the file but leaves the "end of file" mark at the beginning of the file. None – No action is performed on the file.
Delete file on purge	Ready (D) , Transferring (C), On Hold (H), Kept (K), Transferred (T), Executed (X)	CFTSEND, fdelete Ready (D) ->D Transferring (C) -> C On Hold (H) -> H Kept (K) -> K Transferred (T) -> T Executed (X) -> X	Indicates the transfer states of files that will be deleted when you remove the associated transfers from the transfer list or when you purge the transfer list. You can select any combination of statuses. If you do not select anything, files are not deleted even when the associated transfers are removed from the transfer list. Ready – The transfer is available and can start immediately. Transferring – The transfer is being executed. On hold – The transfer was interrupted due to an error, such as a network failure, or by a user. Kept – The transfer was suspended by Transfer CFT or by a user. Transferred – The transfer was successfully completed. Executed – The transfer was ended by an application or user.
Additional information	string max 512, empty by default	CFTSEND, parm	Use this field for any information you want to provide.

Send template > file properties > files

Indicates whether you are sending one or more files.

CG field	CG values	CFTUTIL parameter	Description
Single -> Path	string max 512	CFTSEND, fname	Indicate the single file to be sent.
Multiple -> Path	string max 512	CFTSEND, fname	If you selected Multiple, the value you enter can be: A directory name – All the files in this directory will be transferred. A generic file name, including wildcard characters – Only files that match are transferred. For example, mydirectory/toto*.

CG field	CG values	CFTUTIL parameter	Description
Multiple -> File list	string max 512	CFTSEND, selfname	This field is displayed if you selected Multiple in the Files field. Specify the name of the file that contains the list of files to be transferred. This file is also referred to as an indirection file. It must contain one file name per record, and that name must start in the first column of the file. The file names contained in the file must not contain an asterisk (*). When specifying a file here, the Path field is also required.
Multiple -> Archive name	string max 512	CFTSEND, wfname	Name of the file that contains the set of files to be transmitted. Archive files are sent between systems that have the same operating system (grouped mode). The archive file is created automatically by Transfer CFT at the time of the transfer. The file created will be a zip file on Windows systems and a tar file on Linux/UNIX systems. Because Windows systems do not have default compression utilities, Transfer CFT for Windows includes zip and unzip utilities.
File name sent	string max 512	CFTSEND, nfname	Specify the name of the physical file that is to be used during transmission over the network.

Send template > file properties > file type (Linux and Windows)

CG field	CFTUTIL parameter	Description
Binary	CFTSEND, ftype=B, fcode=BINARY, ncode=BINARY	Specify whether the file is a binary file.
Text	CFTSEND, see *Transfer CFT configuration for FTYPE on Windows and Linux* on page 473	Specify whether the file is a text file.
Stream text	CFTSEND, ftype = J	Specify whether the file is a text file sent in Stream CFT mode.
Record format	CFTSEND, see *Transfer CFT configuration for record format* on page 475	Indicate whether the records in the file are fixed or variable length.

Send template > file properties > file type (IBM i)

CG field	CFTUTIL parameter	Description
Data file	CFTSEND, ftype=D, fcode=BINARY, ncode=BINARY	Specify whether the file is a PF-DTA file.
Save file	CFTSEND, ftype=Z, fcode=BINARY, ncode=BINARY	Specify whether the file is a SAVF file.
Source	CFTSEND, ftype=S	Specify whether the file is a PF-SRC (with header) file.
OS 400 specific	CFTSEND, ftype=E	Specify whether the file is a PF-SRC (no header) file.

Send template > file properties > file type (z/OS)

CG field	CFTUTIL parameter	Description
Autodetect	CFTSEND, ftype= "", fcode=<empty>, ncode=<empty>	Specify whether the file is sent in auto detection mode.
Print file with ASA jump codes	CFTSEND, ftype= A	Specify whether the file is Print file with ASA jump codes.
Print file with machine jump codes	CFTSEND, ftype= M	Specify whether the file is Print file with machine jump codes.
Spanned variable format	CFTSEND, ftype= S, fcode= BINARY, ncode= BINARY	Specify whether the file is a spanned variable file.
ARDSSU	CFTSEND, ftype= 1, fcode= BINARY, ncode= BINARY	Specify whether the file is a ADRDSSU file.
Binary	CFTSEND, ftype= B, fcode= BINARY, ncode= BINARY	Specify whether the file is a binary file.
Text	CFTSEND, ftype= T	Specify whether the file is a text file.
Stream Text	CFTSEND, ftype= J	Specify whether the file is a text file sent in Stream CFT mode.

Send template > processing scripts > pre-processing

CG field	CG values	CFTUTIL parameter	Description
Script -> Filename	Upload new file -> script to upload Use existing file -> Filename field: string of max 512c	CFTSEND, preexec	Specify the script to be executed before the file is transferred. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.
State	Ready, On hold	CFTSEND, prestate Ready ->DISP : default On hold -> HOLD	Indicate the status of the transfer on the source. The script is run only if the transfer is in the specified state. Ready – Indicates that the transfer is available and can start immediately. On hold – Indicates that the transfer is deferred until a remote receive request is accepted, or until a local START command changes this transfer to the ready state.

Send template > processing scripts > post-processing

CG field	CG values	CFTUTIL parameter	Description
Script -> Filename	Upload new file -> script to upload Use existing file -> Filename field: string of max 512c	CFTSEND, exec	Specify the script to be executed after the file is transferred. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.
Apply to group of files	On main request, For each file in group, Both	CFTSEND, execsub On main request -> LIST For each file in group -> SUBF Both -> FILE	This field is displayed if you configure multiple files in send template transfer properties.

Send template > processing scripts > acknowledgment

CG field	CG values	CFTUTIL parameter	Description
Script -> Filename	Upload new file -> script to upload Use existing file -> Filename field: string of max 512c	CFTSEND, ackexec	Specify the script to be executed after an acknowledgement is received for a sent file. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.
State	Require, Ignore	CFTSEND, ackstate Require -> REQUIRE Ignore -> IGNORE	Indicate if the transfer must wait for an acknowledgement. Require – The transfer must wait for an acknowledgement before it can be considered complete. Ignore – The transfer can be considered complete, even if an acknowledgement is not received.
Apply to group of files	On main request, For each file in group, Both	CFTSEND, execsub On main request -> LIST For each file in group -> SUBF Both -> FILE	This field is displayed if you configure multiple files in send template transfer properties.

Send template > processing scripts > error

CG field	CG values	CFTUTIL parameter	Description
Script -> Filename	Upload new file -> script to upload Use existing file -> Filename field: string of max 512c	CFTSEND, exece	Specify the script to be executed after an error occurs during a transfer. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.

Receive template > transfer properties

CG field	CG values	CFTUTIL parameter	Description
Transfer priority	LOW, MEDIUM, HIGH, CUSTOM	CFTRECV, pri LOW ->0 MEDIUM->128 HIGH-> 255 CUSTOM-> integer between 0...255 Empty by default	Transfer priorities are equivalent to integer values ranging from 0 (low) to 255 (high). When Transfer CFT reaches the maximum number of transfers allowed, it queues transfers. When an ongoing transfer is finished and a slot is available for a new transfer, the system selects the one with the highest priority.
Bandwidth allocation	LOW, MEDIUM, HIGH	CFTRECV, cos LOW ->3 MEDIUM->2 HIGH-> 1	The amount of bandwidth allocated to this flow. The value you select determines the data transfer rate for this flow.
Transfer state	Ready, On hold, Kept	CFTRECV, state Ready (D) -> DISP On hold (H) ->HOLD Kept (K) ->KEEP	Indicates the state of the transfer request.: Ready, On Hold, Kept.
User id	string, max 32, empty by default	CFTRECV, userid	Identifier of the transfer owner. If this parameter is not defined, its default value is the system "userid" of the Transfer CFT.

CG field	CG values	CFTUTIL parameter	Description
Detect duplicate transfers	string max 512, empty by default	CFTRECV, duplicat	This field is used in detecting duplicate transfers and may contain a list of symbolic variables separated by a period ".".
Compress file	Yes, No	CFTRECV, ncomp Yes → 15 No → 0	Indicates whether files are compressed before they are transferred.
No file exists	Create, Cancel	CFTRECV, see *Transfer CFT configuration for no file exists, file exists* on page 476	Specifies the action taken if the received file does not already exist. Create – The file is created. Cancel – The transfer is refused.
File exists	Delete, Cancel, Overwrite, Overwrite only if empty	CFTRECV, see *Transfer CFT configuration for no file exists, file exists* on page 476	Specifies the action taken if the received file exists. Delete – The existing file is deleted. Cancel – The transfer is refused. Overwrite – The existing file is overwritten. Overwrite only if empty – The existing file is overwritten only if it contains no data.
Aborted transfer	Keep, Delete	CFTRECV, rkerror	Specifies the action taken if a transfer is terminated due to a file creation error on the target. Keep – The transfer remains in the transfer list. Delete – The transfer is removed from the transfer list.
Delete file on purge	Ready (D) , Transferring (C), On Hold (H), Kept (K), Transferred (T), Executed (X)	CFTRECV, fdelete Ready (D) ->D Transferring (C) -> C On Hold (H) -> H Kept (K) -> K Transferred (T) -> T Executed (X) -> X	Indicates the transfer states of files that will be deleted when you remove the associated transfers from the transfer list or when you purge the transfer list. You can select any combination of statuses. If you do not select anything, files are not deleted even when the associated transfers are removed from the transfer list. Ready – The transfer is available and can start immediately. Transferring – The transfer is being executed. On hold – The transfer was interrupted due to an error, such as a network failure, or by a user. Kept – The transfer was suspended by Transfer CFT or by a user. Transferred – The transfer was successfully completed. Executed – The transfer was ended by an application or user.

Receive template > file properties > files

Indicates whether you are sending a single file or multiple files.

CG field	CG values	CFTUTIL-parameter	Description
Filename	string max 512, for CFT hosted on z/OS and IBM i: defaulted to blank; for other Linux and Windows: pub\&IDF.&IDTU.&FROOT.RCV	CFTRECV, fname	Specify the file name or full path name for the received file or files. Default value: for CFT hosted on z/OS and IBM i defaulted to: blank; for other Linux and Windows: pub\&IDF.&IDTU.&FROOT.RCV
Temporary file	string max 512	CFTRECV, wfname	Specify the name of the temporary file used during the transfer. When the transfer is complete, the temporary file is renamed using the name defined in the Filename field. If you do not specify a value, Transfer CFT directly creates the file with the name specified in the Filename field.

Receive template > file properties > file type

CG field	CFTUTIL parameter	Description
Binary	CFTRECV, ftype=B, fcode=BINARY	Specify whether the file is a binary file.
Text	CFTRECV, see *Transfer CFT configuration for FTYPE on Windows and Linux* on page 473	Specify whether the file is a text file.
Stream text	CFTRECV, ftype = J	Specify whether the file is a text file sent in Stream CFT mode.
Record format	CFTRECV, see *Transfer CFT configuration for record format* on page 475	Indicate whether the records in the file are fixed or variable length.

Receive template > file properties > file type (IBM i)

CG field	CFTUTIL parameter	Description
Data file	CFTRECV, ftype=D, fcode=BINARY	Specify whether the file is a PF-DTA file.
Save file	CFTRECV, ftype=Z, fcode=BINARY	Specify whether the file is a SAVF file.
Source	CFTRECV, ftype=S	Specify whether the file is a PF-SRC (with header) file.
OS 400 specific	CFTRECV, ftype=E	Specify whether the file is a PF-SRC (no header) file.

Receive template > file properties > file type (z/OS)

CG field	CFTUTIL parameter	Description
Autodetect	CFTRECV, ftype= "", fcode=<empty>	Specify whether the file is sent in auto detection mode.
Print file with ASA jump codes	CFTRECV, ftype= A	Specify whether the file is Print file with ASA jump codes.
Print file with machine jump codes	CFTRECV, ftype= M	Specify whether the file is Print file with machine jump codes.
Spanned variable format	CFTRECV, ftype= S, fcode= BINARY	Specify whether the file is a spanned variable file.
ARDSSU	CFTRECV, ftype= 1, fcode= BINARY	Specify whether the file is a ADRDSSU file.
Binary	CFTRECV, ftype= B, fcode= BINARY	Specify whether the file is a binary file.
Text	CFTRECV, ftype= T	Specify whether the file is a text file.
Stream Text	CFTRECV, ftype= J	Specify whether the file is a text file sent in Stream CFT mode.

Receive template > processing scripts > post-processing

CG field	CG values	CFTUTIL parameter	Description
Script -> Filename	if Custom, Filename field: string of max 512c	CFTRECV, exec	Specify the script to be executed after the file is received. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.

Receive template > processing scripts > acknowledgment

CG field	CG values	CFTUTIL parameter	Description
Script -> Filename	if Custom, Filename field: string of max 512c	CFTRECV, ackexec	Specify the script to be executed after the file is received and post-processing is complete. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.
State	Require, Ignore	CFTRECV, ackstate Require -> REQUIRE Ignore -> IGNORE	Indicate if the transfer must wait for an acknowledgement. Require – The transfer must wait for an acknowledgement before it can be considered complete. Ignore – The transfer can be considered complete, even if an acknowledgement is not received.

Receive template > processing scripts > error

CG field	CG values	CFTUTIL parameter	Description
Script -> Filename	if Custom, Filename field: string of max 512c	CFTRECV, exece	Specify the script to be executed if an error occurs when a file is received. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.

Transfer CFT configuration for FTYPE=TEXT

According to the following flow settings table, CFTSEND and CFTRECV FTYPE can have different values: O, F, X or T for FTYPE=TEXT.

CG fields and Transfer CFT OS/ ftype value	O	O	F	X	T
File type	Text	Text	Text	Text	Text
End of record character	CRLF	CRLF	CRLF	LF	BOTH
Transfer CFT OS	Linux	WINDOWS	WINDOWS	doesn't matter	doesn't matter
Ignore end of file character	doesn't matter	NO	YES	doesn't matter	doesn't matter

Transfer CFT configuration for encoding/transcoding

According to the following flow settings table, four fields from CFTSEND and CFTRECV are configured for each Transfer CFT selected in flow target: fcode, ncode, fcharset, ncharset.

CG field	CFTSEND / CFTRECV	CG value -> CFTSEND/CFTRECV field value
Encoding	fcode	ASCII -> ASCII EBCDIC -> EBCDIC CUSTOM -> <empty>

CG field	CFTSEND / CFTRECV	CG value -> CFTSEND/CFTRECV field value
Encoding charset	fcharset	For file type Text: if Encoding=ASCII/EBCDIC => fcharset is not set else fcode is not set and fcharset= the value set in field and string max 32 For file type Stream Text: fcharset is not set
Transcoding	ncode	None -> <empty> ASCII -> ASCII EBCDIC -> EBCDIC CUSTOM -> <empty>
Transcoding charset	ncharset	For file type Text: if Transcoding =None/ASCII/EBCDIC => ncharset is not set else ncode is not set and ncharset= the value set in field and string max 32 For file type Stream Text: ncharset is not set

CG field	CFTRECV	CG value -> CFTRECV field value
Transcoding	ncharset	For file type Text: None -> “” ASCII -> ASCII EBCDIC -> EBCDIC CUSTOM -> else, the value set in field, string max 32 For file type Stream Text: fcharset = “”
Transcoding charset	ncharset	The value set in field, string max 32

Transfer CFT configuration for record format

CG field	CFTSEND/ CFTRECV	CG value -> CFTSEND/CFTRECV field value	CG default value	Comment
Record format	frecfm	Fixed -> F Variable -> V CFTSEND/CFTRECVfrecm can have values: U, <empty> not available on CG, after the CFT registration it will be mapped to the CG UI as follows U, empty -> Variable	Variable	
Trimming character	fpad	16 characters	<empty>	
Maximum record length	flrecl	if option is checked => 0 value must be set in CFT else=> the next input text is enabled and user may enter a value between 0...32767	Default OS value, flrecl=0	If maximum is checked, at deployment, value set in CFTSEND will be 0, and CFT will interpret it as : for CFT on windows => 512 for CFT on unix => 512 for text files (FTYPE=T, O or X) => 4096 for binary files (FTYPE=B)

Transfer CFT configuration for no file exists, file exists

Two CFTRECV fields correspond to the two CG fields.

The Comments column are remarks about file creation rule on receiver side regarding the existence of the file in the same location with the same name.

CG values	CFTRECV, fdisp	CFTRECV, faction	Comments
No file exists=CREATE File exists=DELETE	both	delete	If no file exists, the file will be created. If file exists it will be deleted and recreated (no matter if it is empty or not).

CG values	CFTRECV, fdisp	CFTRECV, faction	Comments
No file exists=CREATE File exists=OVERWRITE	both	erase	If no file exists, the file will be created. If file exists it will be overwritten (no matter if it is empty or not).
No file exists=CREATE File exists=OVERWRITE ONLY IF EMPTY	both	verify	If no file exists, the file will be created. If file exists and it is not empty, the transfer is aborted. If file exists but it is empty, the file will be overwritten.
No file exists=CREATE File exists=CANCEL	new	verify	If no file exists, the file will be created. If file exists the transfer will be aborted (no matter if it is empty or not).
No file exists=CANCEL File exists=DELETE	old	delete	If no file exists, the transfer is aborted. If file exists the file will be deleted and recreated (no matter if it is empty or not).
No file exists=CANCEL File exists=OVERWRITE	old	erase	If no file exists, the transfer is aborted. If file exists the file will be overwritten (no matter if it is empty or not).
No file exists=CANCEL File exists=OVERWRITE ONLY IF EMPTY	old	verify	If no file exists, the transfer is aborted. If file exists and it is not empty, the transfer is aborted. If file exists but it is empty, the file will be overwritten.

Flow configurations in Central Governance and CFTUTIL

The following tables list and describe corresponding flow configurations in Central Governance and Transfer CFT. Default Central Governance values are underlined.

Flow definition: Source

The following tables describe the fields and parameters available in flow definition, source side, in Central Governance and CFTUTIL.

Transfer properties

CG field	CG values	CFTUTIL parameter	Description
Transfer priority	LOW, MEDIUM, HIGH, CUSTOM	CFTSEND, pri LOW >0 MEDIUM>128 HIGH> 255 CUSTOM> integer between 0...255	Transfer priorities are equivalent to integer values ranging from 0 (low) to 255 (high). When Transfer CFT reaches the maximum number of transfers allowed, it queues transfers. When an ongoing transfer is finished and a slot is available for a new transfer, the system selects the one with the highest priority. Transfer priority is available when the source is the initiator.
Bandwidth allocation	LOW, MEDIUM, HIGH	CFTSEND, cos LOW >3 MEDIUM>2 HIGH> 1	The amount of bandwidth allocated to this flow. The value you select determines the data transfer rate for this flow.
Transfer state	Ready, On hold, Kept	CFTSEND, state Ready (D) > DISP On hold (H) >HOLD Kept (K) >KEEP	Indicates the state of the transfer request.: Ready, On Hold, Kept. Field is available in UI only if the initiator is the source. If target is the initiator, in source side the transfer state is ready and the field cannot be configured in the Central Governance UI.
User id	string, max 32, empty by default	CFTSEND, userid	Identifier of the transfer owner. If this parameter is not defined, its default value is the system userid of the Transfer CFT.
Sender User id	string, max 32, empty by default	CFTSEND, suser	Identifier of the user sending the file.
Receiver User id	string, max 32, empty by default	CFTSEND, ruser	Identifier of the user receiving the file.
Detect duplicate transfers	string max 512, empty by default	CFTSEND, duplicat	This field is used in detecting duplicate transfers and may contain a list of symbolic variables separated by a period ".".

CG field	CG values	CFTUTIL parameter	Description
On file not found	Abort transfer, Ignore transfer	CFTSEND, filenotfound Abort transfer > ABORT Ignore transfer > IGNORE	Specify whether the flow fails if the file to transfer is not found.
On file modification	Continue transfer, Stop transfer	CFTSEND , fdisp Continue transfer > SHR Stop transfer> CHECK	Available only in source side. Specify what happens if files are modified during the transfer.
Action after transfer	Delete file , Delete file content, None	CFTSEND, faction Delete file > Delete Delete file content > Erase None> None: default	Specifies what happens to the file in source side when the transfer is complete. Delete file – Deletes the file. Delete file content – Removes the contents of the file but leaves the "end of file" mark at the beginning of the file. None – No action is performed on the file.
Delete file on purge	Ready (D) , Transferring (C), On Hold (H), Kept (K), Transferred (T), Executed (X)	CFTSEND, fdelete Ready (D) >D Transferring (C) > C On Hold (H) > H Kept (K) > K Transferred (T) > T Executed (X) > X	Indicates the transfer states of files that will be deleted when you remove the associated transfers from the transfer list or when you purge the transfer list. You can select any combination of statuses. If you do not select anything, files are not deleted even when the associated transfers are removed from the transfer list. Ready – The transfer is available and can start immediately. Transferring – The transfer is being executed. On hold – The transfer was interrupted due to an error, such as a network failure, or by a user. Kept – The transfer was suspended by Transfer CFT or by a user. Transferred – The transfer was successfully completed. Executed – The transfer was ended by an application or user.
Purge completed transfer	Yes, No	CFTSEND, delete	Indicates whether a completed transfer is purged from the transfer list or kept.
Additional information	string max 512, empty by default	CFTSEND, parm	Use this field for any information you want to provide.

CG field	CG values	CFTUTIL parameter	Description
Maximum transfer duration	[0; 32767]	CFTSEND, MAXDURATION	Zero (0) indicates a file transfer never times out.
Start date		CFTSEND, MINDATE	Example of the mapping: 01 Apr 2014 --> CFT: 20140401 11 Jan 2017 --> CFT: 20170111 If the value is not indicated, Transfer CFT populates it with 10000101.
Start time		CFTSEND, MINTIME	Example of the mapping: 3:12:11 --> CFT : 13121100 00:00:01 --> CFT : 00000100 MINTIME default 00:00:00
End date		CFTSEND, MAXDATE	If a value is not specified, Transfer CFT uses 99991231.
End time		CFTSEND, MAXTIME	MAXTIME default 23:59:59
On file modification	Default, All, First and last, None	CFTSEND, trk Default > UNDEFINED All > ALL First and last > SUMMARY None > NO	Specify the level of transfer process step details to send as events to the Visibility service.

PeSIT properties

CG field	CG values	CFTUTIL parameter	Description
Compress file	Yes, No	CFTSEND, ncomp Yes → 15 No → 0	Indicates whether files are compressed before they are transferred. Same value will be used for the compression on target side for the field CFTRECV, ncomp.

File properties > files

Indicates whether you are sending one or more files.

CG field	CG values	CFTUTIL parameter	Description
Single -> Path	string max 512	CFTSEND, fname	Indicate the single file to be sent.
Multiple > Path	string max 512	CFTSEND, fname	If you selected Multiple, the value you enter can be: A directory name – All the files in this directory will be transferred. A generic file name, including wildcard characters – Only files that match are transferred. For example, mydirectory/toto*.
Multiple > File list	string max 512	CFTSEND, selfname	This field is displayed if you selected Multiple in the Files field. Specify the name of the file that contains the list of files to be transferred. This file is also referred to as an indirection file. It must contain one file name per record, and that name must start in the first column of the file. The file names contained in the file must not contain an asterisk (*). When specifying a file here, the Path field is also required.
Multiple > Archive name	string max 512, &IDF.&idtu.rcv	CFTSEND, wfname	Name of the file that contains the set of files to be transmitted. Archive files are sent between systems that have the same operating system (grouped mode). The archive file is created automatically by Transfer CFT at the time of the transfer. The file created will be a zip file on Windows systems and a tar file on Linux/UNIX systems. Because Windows systems do not have default compression utilities, Transfer CFT for Windows includes zip and unzip utilities.
File name sent	string max 512	CFTSEND, nfname	Specify the name of the physical file that is to be used during transmission over the network.
Working directory	string max 512	CFTSEND/WORKINGDIR	Indicates the path to the directory for sent files in process and temporary files.

File properties > file type

CG field	CFTUTIL parameter	Description
Binary	CFTSEND, ftype=B, fcode=BINARY, ncode=BINARY	Specify whether the file is a binary file.
Text	CFTSEND, see *Transfer CFT configuration for FTYPE on Windows and Linux* on page 473	Specify whether the file is a text file.
Record format	CFTSEND, see *Transfer CFT configuration for record format* on page 475	Indicate whether the records in the file are fixed or variable length.

Processing scripts > pre-processing

CG field	CG values	CFTUTIL parameter	Description
Script > Filename	Upload new file > script to upload Use existing file > Filename field: string of max 512c	CFTSEND, preexec	Specify the script to be executed before the file is transferred. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.
State	Ready, On hold	CFTSEND, prestate Ready >DISP : default On hold > HOLD	Indicate the status of the transfer on the source. The script is run only if the transfer is in the specified state. Ready – Indicates that the transfer is available and can start immediately. On hold – Indicates that the transfer is deferred until a remote receive request is accepted, or until a local START command changes this transfer to the ready state.
Apply to broadcast list	On main request, For each target in the list, Both	CFTDEST, execpre On main request > DEST For each target in the list > CHILDREN Both > PART	This field is displayed if you enabled a broadcast list in source transfer properties. On main request – Executes the script only on the main request. For each target in the list – Executes the script only for each target in the list. Both – Executes the script both for the main request and for each target in the list.

Processing scripts > post-processing

CG field	CG values	CFTUTIL parameter	Description
Script > Filename	Upload new file > script to upload Use existing file > Filename field: string of max 512c	CFTSEND, exec	Specify the script to be executed after the file is transferred. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.
Apply to group of files	On main request, For each file in group, Both	CFTSEND, execsub On main request -> LIST For each file in group -> SUBF Both -> FILE	This field is displayed if you configure multiple files in source transfer properties. Values – On main request | For each target in the list | Both On main request – Executes the script only on the main request. For each target in the list – Executes the script only for each target in the list. Both – Executes the script both for the main request and for each target in the list.
Apply to broadcast list	On main request, For each target in the list, Both	CFTDEST, exec On main request > DEST For each target in the list > CHILDREN Both > PART	This field is displayed if you configure multiple files in source transfer properties. On main request – Executes the script only on the main request. For each target in the list – Executes the script only for each target in the list. Both – Executes the script both for the main request and for each target in the list.

Processing scripts > acknowledgment

CG field	CG values	CFTUTIL parameter	Description
Script > Filename	Upload new file > script to upload Use existing file > Filename field: string of max 512c	CFTSEND, ackexec	Specify the script to be executed after an acknowledgement is received for a sent file. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.
State	Require, Ignore	CFTSEND, ackstate Require > REQUIRE Ignore > IGNORE	Indicate if the transfer must wait for an acknowledgment. Require – The transfer must wait for an acknowledgment before it can be considered complete. Ignore – The transfer can be considered complete, even if an acknowledgment is not received.
Apply to group of files	On main request, For each file in group, Both	CFTSEND, execsub On main request -> LIST For each file in group -> SUBF Both -> FILE	This field is displayed if you configure multiple files in source transfer properties. Values – On main request | For each target in the list | Both On main request – Executes the script only on the main request. For each target in the list – Executes the script only for each target in the list. Both – Executes the script both for the main request and for each target in the list.
Apply to broadcast list	On main request, For each target in the list, Both	CFTDEST, execa On main request > DEST For each target in the list > CHILDREN Both > PART	This field is displayed if you enabled a broadcast list in source transfer properties. On main request – Executes the script only on the main request. For each target in the list – Executes the script only for each target in the list. Both – Executes the script both for the main request and for each target in the list.

Processing scripts > error

CG field	CG values	CFTUTIL parameter	Description
Script > Filename	Upload new file > script to upload Use existing file > Filename field: string of max 512c	CFTSEND, exece	Specify the script to be executed after an error occurs during a transfer. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.

Flow definition: Target

The following tables describe the fields and parameters available in flow definition, target side, in Central Governance and CFTUTIL.

Transfer properties

CG field	CG values	CFTUTIL parameter	Description
Transfer priority	LOW, MEDIUM, HIGH, CUSTOM	CFTSEND, pri LOW >0 MEDIUM>128 HIGH> 255 CUSTOM> integer between 0...255	Transfer priorities are equivalent to integer values ranging from 0 (low) to 255 (high). When Transfer CFT reaches the maximum number of transfers allowed, it queues transfers. When an ongoing transfer is finished and a slot is available for a new transfer, the system selects the one with the highest priority. The same priority is used for the transfer on the target side. Transfer priority is available when the target is the initiator.
Bandwidth allocation	LOW, MEDIUM, HIGH	CFTSEND, cos LOW >3 MEDIUM>2 HIGH> 1	The amount of bandwidth allocated to this flow. The value you select determines the data transfer rate for this flow. The same bandwidth allocation is used for the transfer on the target side.

CG field	CG values	CFTUTIL parameter	Description
Transfer state	Ready, On hold, Kept	CFTRECV, state Ready (D) > DISP On hold (H) >HOLD Kept (K) >KEEP	Indicates the state of the transfer request: Ready, On Hold, Kept. Field is available in UI only if the initiator is the target. If the initiator is the source, on the target side the transfer state has the value Ready and the field cannot be configured in the Central Governance UI.
User id	string, max 32, empty by default	CFTRECV, userid	Identifier of the transfer owner. If this parameter is not defined, its default value is the system userid of the Transfer CFT.
Sender User id	string, max 32, empty by default	CFTSEND, suser	Identifier of the user sending the file.
Receiver User id	string, max 32, empty by default	CFTSEND, ruser	Identifier of the user receiving the file.
Detect duplicate transfers	string max 512, empty by default	CFTRECV, duplicat	This field is used in detecting duplicate transfers and may contain a list of symbolic variables separated by a period ".".
On file not found	Abort transfer, Ignore transfer	CFTSEND, filenotfound Abort transfer > ABORT Ignore transfer > IGNORE	Specify whether the flow fails if the file to transfer is not found.
No file exists	Create, Cancel	CFTRECV, see *Transfer CFT configuration for no file exists, file exists* on page 476	Specifies the action taken if the received file does not already exist. Create – The file is created. Cancel – The transfer is refused.
File exists	Delete, Cancel, Overwrite, Overwrite only if empty	CFTRECV, see *Transfer CFT configuration for no file exists, file exists* on page 476	Specifies the action taken if the received file exists. Delete – The existing file is deleted. Cancel – The transfer is refused. Overwrite – The existing file is overwritten. Overwrite only if empty – The existing file is overwritten only if it contains no data.

CG field	CG values	CFTUTIL parameter	Description
Aborted transfer	Keep, Delete	CFTRECV, rkerror	Specifies the action taken if a transfer is terminated due to a file creation error on the target. Keep – The transfer remains in the transfer list. Delete – The transfer is removed from the transfer list.
Delete file on purge	Ready (D) , Transferring (C), On Hold (H), Kept (K), Transferred (T), Executed (X)	CFTRECV, fdelete Ready (D) >D Transferring (C) > C On Hold (H) > H Kept (K) > K Transferred (T) > T Executed (X) > X	Indicates the transfer states of files that will be deleted when you remove the associated transfers from the transfer list or when you purge the transfer list. You can select any combination of statuses. If you do not select anything, files are not deleted even when the associated transfers are removed from the transfer list. Ready – The transfer is available and can start immediately. Transferring – The transfer is being executed. On hold – The transfer was interrupted due to an error, such as a network failure, or by a user. Kept – The transfer was suspended by Transfer CFT or by a user. Transferred – The transfer was successfully completed. Executed – The transfer was ended by an application or user.
Purge completed transfer	Yes, No	CFTSEND, delete	Indicates whether a completed transfer is purged from the transfer list or kept.
Maximum transfer duration	[0; 32767]	CFTRECV, MAXDURATION	Zero (0) indicates a file transfer never times out.
Start date		CFTRECV, MINDATE	Example of the mapping: 01 Apr 2014 --> CFT: 20140401 11 Jan 2017 --> CFT: 20170111 If the value is not indicated, Transfer CFT populates it with 10000101.
Start time		CFTRECV, MINTIME	Example of the mapping: 3:12:11 --> CFT : 13121100 00:00:01 --> CFT : 00000100 MINTIME default 00:00:00
End date		CFTRECV, MAXDATE	If a value is not specified, Transfer CFT uses 99991231.
End time		CFTRECV, MAXTIME	MAXTIME default 23:59:59

CG field	CG values	CFTUTIL parameter	Description
On file modification	Default, All, First and last, None	CFTSEND, trk Default > UNDEFINED All > ALL First and last > SUMMARY None > NO	Specify the level of transfer process step details to send as events to the Visibility service.

File properties > files

Indicates whether you are sending a single file or multiple files.

CG field	CG values	CFTUTIL-parameter	Description
Filename	string max 512, Default value: pub\&IDF.&IDTU.&FROOT.RCV	CFTRECV, fname	Specify the file name or full path name for the received file or files. This field is required if the initiator of the flow is the source. Default value: pub\&IDF.&IDTU.&FROOT.RCV
Temporary file	string max 512	CFTRECV, wfname	Specify the name of the temporary file used during the transfer. When the transfer is complete, the temporary file is renamed using the name defined in the Filename field. If you do not specify a value, Transfer CFT directly creates the file with the name specified in the Filename field. If the File exists parameter is set to Overwrite after receiving temporary file, Temporary file becomes required.
Working directory	string max 512	CFTRECV/WORKINGDIR	Indicates the path to the directory for received files in process and temporary files.
Receiving file size	[0; 2147483647]	CFTRECV, FSPACE	Specify the size allocation in kilobytes of the incoming file. If set to **0**, Transfer CFT allocates space according to the file size specified by the sender at the protocol level. The parameter is available only for z/OS computers.

File properties > file type

CG field	CFTUTIL parameter	Description
Binary	CFTRECV, ftype=B, fcode=BINARY, fcharset="", ncharset=""	Specify whether the file is a binary file.
Text	CFTRECV, see *Transfer CFT configuration for FTYPE on Windows and Linux* on page 473	Specify whether the file is a text file.
Record format	CFTRECV, see *Transfer CFT configuration for record format* on page 475	Indicate whether the records in the file are fixed or variable length.

Processing scripts > post-processing

CG field	CG values	CFTUTIL parameter	Description
Script > Filename	if Custom, Filename field: string of max 512c	CFTRECV, exec	Specify the script to be executed after the file is received. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.
Apply to group of files	On main request, For each file in group, Both	CFTSEND, execsub On main request -> LIST For each file in group -> SUBF Both -> FILE	This field is displayed if you enabled a broadcast list in source transfer properties. Values – On main request | For each target in the list | Both On main request – Executes the script only on the main request. For each target in the list – Executes the script only for each target in the list. Both – Executes the script both for the main request and for each target in the list.

CG field	CG values	CFTUTIL parameter	Description
Apply to collect list	On main request, For each source in the list, Both	CFTDEST, exec On main request > DEST For each source in the list > CHILDREN Both > PART	This field is displayed if you enabled a collect list in target transfer properties. On main request – Executes the script only on the main request. For each source in the list – Executes the script only for each source in the list. Both – Executes the script both for the main request and for each source in the list.

Processing scripts > acknowledgment

CG field	CG values	CFTUTIL parameter	Description
Script > Filename	if Custom, Filename field: string of max 512c	CFTRECV, ackexec	Specify the script to be executed after the file is received and post-processing is complete. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.
State	Require, Ignore	CFTRECV, ackstate Require > REQUIRE Ignore > IGNORE	Indicate if the transfer must wait for an acknowledgment. Require – The transfer must wait for an acknowledgment before it can be considered complete. Ignore – The transfer can be considered complete, even if an acknowledgement is not received.
Apply to collect list	On main request, For each source in the list, Both	CFTDEST, execa On main request > DEST For each source in the list > CHILDREN Both > PART	This field is displayed if you enabled a collect list in target transfer properties. On main request – Executes the script only on the main request. For each source in the list – Executes the script only for each source in the list. Both – Executes the script both for the main request and for each source in the list.

Processing scripts > error

CG field	CG values	CFTUTIL parameter	Description
Script > Filename	if Custom, Filename field: string of max 512c	CFTRECV, exece	Specify the script to be executed if an error occurs when a file is received. • z/OS: Filename length not to exceed 10 chars • IBM i: Filename name must contain the sequence of blocks 8 chars + "." • IBM i + z/OS: Upload only EBCDIC-encoded files. Upload of other files results in deployment failure.

Transfer CFT configuration for FTYPE on Windows and Linux

According to the following flow settings table, CFTSEND and CFTRECV FTYPE can have different values: O, F, X or T for FTYPE=TEXT.

CG fields and Transfer CFT OS/ ftype value	O	O	F	X	T
File type	Text	Text	Text	Text	Text
End of record character	CRLF	CRLF	CRLF	LF	BOTH
Transfer CFT OS	Linux	WINDOWS	WINDOWS	doesn’t matter	doesn’t matter
Ignore end of file character	doesn’t matter	NO	YES	doesn’t matter	doesn’t matter

Transfer CFT configuration for FTYPE on IBM i

According to the following flow settings table, CFTSEND and CFTRECV FTYPE for IBM i can have the following values: D, E, S, Z. Encoding and transcoding fields are configurable when fcode and ncode support other than BINARY.

Central Governance File type value	Transfer CFT ftype value	Central Governance encoding/transcoding and Transfer CFT fcode/ncode value
Data file (default)	D	See *Transfer CFT configuration for encoding/transcoding* on page 474
Save file	E	See *Transfer CFT configuration for encoding/transcoding* on page 474
Source	S	fcode=BINARY, ncode=BINARY
OS400 specific	Z	fcode=BINARY, ncode=BINARY

Transfer CFT configuration for FTYPE on Z/OS

According to the following flow settings table, CFTSEND and CFTRECV FTYPE for Z/OS can have the following values: A, M, S, 1, _, B, T. Or it can be auto-detected. FTYPE=_ only is available for CFTRECV. Encoding and transcoding fields are available for configuration when fcode and ncode support other than BINARY.

Central Governance File type value	Transfer CFT ftype value	Central Governance encoding/transcoding and Transfer CFT fcode/ncode value
Autodetect (default)	<empty>	Not defined
Print file with ASA jump codes	A	See *Transfer CFT configuration for encoding/transcoding* on page 474
Print file with machine jump codes	M	See *Transfer CFT configuration for encoding/transcoding* on page 474
Spanned variable format	S	fcode=BINARY, ncode=BINARY
ARDSSU	1	fcode=BINARY, ncode=BINARY
PDSE (only for receiver)	_	fcode=BINARY, ncode=BINARY
Binary	B	fcode=BINARY, ncode=BINARY
Text	T	See *Transfer CFT configuration for encoding/transcoding* on page 474

Transfer CFT configuration for encoding/transcoding

According to the following flow settings table, four fields from CFTSEND and CFTRECV are configured for each Transfer CFT selected in flow target: fcode, ncode, fcharset, ncharset.

CG field	CFTSEND / CFTRECV	CG value -> CFTSEND/CFTRECV field value
Encoding	fcode	ASCII -> ASCII EBCDIC -> EBCDIC CUSTOM -> <empty>
Encoding charset	fcharset	For file type Text: if Encoding=ASCII/EBCDIC => fcharset is not set else fcode is not set and fcharset= the value set in field and string max 32 For file type Stream Text: fcharset is not set
Transcoding	ncode	None -> ' ' ASCII -> ASCII EBCDIC -> EBCDIC CUSTOM -> <empty> (Windows and Linux only)
Transcoding charset	ncharset	For file type Text: if Transcoding =None/ASCII/EBCDIC => ncharset is not set else ncode is not set and ncharset= the value set in field and string max 32 For file type Stream Text: ncharset is not set

Transfer CFT configuration for record format

CG field	CFTSEND/ CFTRECV	CG value -> CFTSEND/CFTRECV field value	CG default value	Comment
Record format	frecfm	Fixed -> F Variable -> V	Variable	

CG field	CFTSEND/ CFTRECV	CG value -> CFTSEND/CFTRECV field value	CG default value	Comment
Unpadding character	fpad	16 characters	<empty>	If character typed in padding/unpadding character is not visible, the character name must be displayed. (If space character is typed as unpadding character, label SPACE is displayed.) To use a special character as a trimming or padding character, set the hexadecimal value as follows: • x'08FD' (base UTF-16) up to 8 characters if the charset is 32 bits (like UTF-32LE) • x'084FD1AE4' (base UTF-32) up to 16 characters
Maximum record length	flrecl	if option is checked => 0 value must be set in CFT else=> the next input text is enabled and user may enter a value between 0...32767	option checked, flrecl=0	If maximum is checked, at deployment, value set in CFTSEND will be 0, and CFT will interpret it as : for CFT on windows => 512 for CFT on unix => 512 for text files (FTYPE=T, O or X) => 4096 for binary files (FTYPE=B)

Transfer CFT configuration for no file exists, file exists

Two CFTRECV fields correspond to the two CG fields.

The Comments column are remarks about file creation rule on receiver side regarding the existence of the file in the same location with the same name.

CG values	CFTRECV, fdisp	CFTRECV, faction	Comments
No file exists=CREATE File exists=DELETE	both	delete	If no file exists, the file will be created. If file exists it will be deleted and recreated (no matter if it is empty or not).
No file exists=CREATE File exists=OVERWRITE	both	erase	If no file exists, the file will be created. If file exists it will be overwritten (no matter if it is empty or not).

CG values	CFTRECV, fdisp	CFTRECV, faction	Comments
No file exists=CREATE File exists=OVERWRITE ONLY IF EMPTY	both	verify	If no file exists, the file will be created. If file exists and it is not empty, the transfer is aborted. If file exists but it is empty, the file will be overwritten.
No file exists=CREATE File exists=CANCEL	new	verify	If no file exists, the file will be created. If file exists the transfer will be aborted (no matter if it is empty or not).
No file exists=CANCEL File exists=DELETE	old	delete	If no file exists, the transfer is aborted. If file exists the file will be deleted and recreated (no matter if it is empty or not).
No file exists=CANCEL File exists=OVERWRITE	old	erase	If no file exists, the transfer is aborted. If file exists the file will be overwritten (no matter if it is empty or not).
No file exists=CANCEL File exists=OVERWRITE ONLY IF EMPTY	old	verify	If no file exists, the transfer is aborted. If file exists and it is not empty, the transfer is aborted. If file exists but it is empty, the file will be overwritten.
No file exists=CANCEL File exists=OVERWRITE AFTER RECEIVING TEMPORARY FILE	new	rename	If no file exists, the transfer is aborted. If file exists, the file is overwritten after receiving the temporary file by renaming the temporary file with the file name. This option is available only for Transfer CFT on Unix.

Combination forbidden: No file exists=CANCEL, File exists=CANCEL

Transfer CFT partners in flows

The following topics define Transfer CFT partners as used in Central Governance flows.

When a flow using Transfer CFTs is deployed, Central Governance deploys on each Transfer CFT the definition of the partners defined in the flow. There are two Transfer CFT partner objects involved: CFTPART and CFTTCP.

- For each CFTPART, Central Governance sets values for the fields ID, NRPART, NRPASSW, NSPART, NSPASSW, PROT, SAP, SSL (if mutual authentication is used)
- For each CFTTCP, Central Governance sets values for the fields ID, CLASS, HOST.

All the other fields of CFTTCP and CFTPART have the default values.

You can overwrite the following fields with values from the partner template configuration file.

Transfer CFT field	Partner template parameter	Partner template value	Default Central Governance value if property is not set
CFTPART - ID	Name of the Transfer CFT partner.	%HOSTNAME% (product host as it appears in the product list)	STRING max_length=32
CFTTCP – CNXIN, CNXOUT, CNXINOUT	cft.partner.cnxin cft.partner.cnxout cft.partner.cnxinout	0..1000	CNXIN = '64', CNXOUT = '64', CNXINOUT = '64'
CFTTCP – RETRYW, RETRYM, RETRYN	cft.partner.retryw cft.partner.retryn cft.partner.retrym	0..32767	RETRYW = '7', RETRYM = '12', RETRYN = '6',

When a broadcast or collect list is used in flows, Central Governance deploys a new object, CFTDEST. This is in addition to partners definition (CFTPART and CFTTCP objects) for each Transfer CFT partner.

Transfer CFT partner definition

This topic has tables for defining Transfer CFT partners:

- CFTPART fields and corresponding Central Governance fields
- CFTTCP fields and corresponding Central Governance fields

CFTPART

The following table lists the CFTPART fields and corresponding Central Governance fields for defining Transfer CFT partners.

- The same protocol must always be used between two Transfer CFTs. Only the security option can change.
- There is a Central Governance limitation. If you change the protocol in a flow, the configuration associated to the previous protocol is not removed on the Transfer CFTs involved in the flow. The new configuration is added in the list of available communications modes with the partner.

CFTPART field	Corresponding field in Central Governance	Type
ID	Name of the Transfer CFT partner.	STRING max_length=32
NRPART	Name of the Transfer CFT partner.	STRING max_length=24
NRPASSW	PeSIT password of the Transfer CFT. It is the reversed value of the Transfer CFT name.	STRING max_length=8

CFTPART field	Corresponding field in Central Governance	Type
NSPART	Name of the current Transfer CFT.	STRING max_length=24
NSPASSW	PeSIT password of the Transfer CFT. It is the reversed value of the partner Transfer CFT name.	STRING max_length=8
PROT	Protocol name defined on the current Transfer CFT for the flow protocol options: network protocol/security.	STRING LIST max_length=32 max_entries=4
SAP	Port defined in the partner Transfer CFT for the flow protocol options: network protocol/security.	STRING LIST max_length=32 max_entries=4 If the Transfer CFT partner is only receiving data (is the target, Direction = Sender pushes files and acknowledgments are not enabled), the SAP field is not set.
SYST	Operating system of the unmanaged product.	STRING

CFTTCP

The following table lists the CFTTCP fields and corresponding Central Governance fields for defining Transfer CFT partners.

If the flow has relays defined, additional rules are applied. *Flow with relays* on page 482.

CFTTCP field	Corresponding field in Central Governance	Type
ID	Name of the Transfer CFT partner.	STRING, max_length=32
HOST	Host of the current Transfer CFT.	string list max_length=512 max_entries=4 If the Transfer CFT partner is only receiving data (is the target, Direction = Sender pushes files and acknowledgments are not enabled), the HOST is set to INADDR_ANY.
CLASS	Class of the network protocol defined on the current Transfer CFT for the flow protocol option: network protocol. 1 - TCP 2 - UDT 3 - TCP	Number min=0 max=64

Unmanaged product partner definition

This topic has tables for defining unmanaged product partners:

- CFTPART fields and corresponding Central Governance fields
- CFTTCP fields and corresponding Central Governance fields

CFTPART

The following table lists the CFTPART fields and corresponding Central Governance fields for defining unmanaged product partners.

If the flow has relays defined, additional rules are applied. See *Flow with relays* on page 482.

CFTPART field	Corresponding field in Central Governance	Type
ID	PeSIT login of the unmanaged product.	STRING max_length=24
NRPART	PeSIT login of the unmanaged product.	STRING max_length=24
NRPASSW	PeSIT password of the unmanaged product.	STRING max_length=8
NSPART	Name of the current Transfer CFT.	STRING max_length=32
NSPASSW	Reversed value of the current Transfer CFT name.	STRING max_length=40
PROT	Protocols defined on the Transfer CFT for the flow protocol options: network protocol/security.	
SAP	Ports defined in the unmanaged product for the flow protocol options: network protocol/security.	STRING LIST max_length=32 max_entries=4

If mutual authentication is used, the CFTSSL object from the CFT with the ID=SSL_Default is updated. The alias of the new certificate is added to the value of the field ROOTCID.

CFTTCP

The following table lists the CFTTCP fields and corresponding Central Governance fields for defining unmanaged product partners.

CFTTCP field	Corresponding field in Central Governance	Type
ID	PeSIT login of the unmanaged product.	STRING max_length=24

CFTTCP field	Corresponding field in Central Governance	Type
HOST	Hosts of the unmanaged product.	STRING list max_ length=512 max_entries=4
CLASS	Class of the network protocol defined on the current Transfer CFT for the flow protocol option: network protocol. 1 - TCP 2 - UDT 3 - pTCP	Number <1> min=0 max=64

Flow using distribution list

When a broadcast or collect list is used in flows, Central Governance deploys a new object, CFTDEST. This is in addition to partners definition (CFTPART and CFTTCP objects) for each Transfer CFT partner.

- Broadcast list for each Transfer CFT defined in source
- Collect list for each Transfer CFT defined in target

CFTDEST field	Corresponding field in Central Governance	Type
ID	Name of the broadcast or collect list defined in flow UI.	STRING max_length=32
FNAME	Broadcast list / Collect list -> File Name of the file containing the names of the Transfer CFTs or the PeSIT logins of the partners. The file is generated by Central Governance based on the sources or targets selected. For a broadcast list, the file contains the partners selected in the flow target. For a collect list, the file contains the partners selected in the flow source.	STRING max_length=512 The number of partners in the file is not limited.

CFTDEST field	Corresponding field in Central Governance	Type
PART	Broadcast list / Collect list -> Partner list The names of the Transfer CFTs or the PeSIT logins of the partners. For a broadcast list, the field contains the partners selected in the flow target. For a collect list, the field contains the partners selected in the flow source.	33*200 STRING LIST each item: max_length=32 list: max_entries=200 If more than 200 Transfer CFTs are selected as partners, Central Governance warns that it is recommended to use the File option as Transfer CFT PART supports maximum 200 partners.
NOPART	Flow Source, Transfer Properties section -> Unknown target Flow Target, Transfer Properties section -> Unknown source Possible values: Central Governance: / Transfer CFT: Cancel / Abort Continue / Continue Ignore / Ignore	

Flow with relays

A flow in Central Governance with defined relays is deployed to force routing of the file by the intermediate partner (explicit store and forward). The IPART parameter defines the immediate partner of the initial sender.

The following are additional rules for defining partners in flows.

On Transfer CFT source

- Partner for the Transfer CFT target (normal definition) +
    - If relays exist:
        - ipart=<first relay>
        - OMINTIME=0
        - OMAXTIME=0
- + If relays exists , partner for the first Transfer CFT relay (normal definition)

On each relay

- Partner for the Transfer CFT before (normal definition)
- Partner with the Transfer CFT target +:
    - If next Transfer CFT is not the Transfer CFT target:
        - ipart=<next partner>
        - OMINTIME=0
        - OMAXTIME=0
    - Partner for the Transfer CFT defined in the ipart parameter (normal definition) (if ipart Transfer CFT is different then target)
- Partner with the Transfer CFT source +:
    - if Transfer CFT before is not the Transfer CFT source:
        - ipart=<Partner before>
        - OMINTIME=0
        - OMAXTIME=0

On Transfer CFT target

- Partner with the Transfer CFT source:
    - If Transfer CFT relay exists:
        - Ipart=<partner before >
        - OMINTIME=0
        - OMAXTIME=0
- + If relays exist, CFTPART for the last relay (normal definition)

Operating systems and deployment correspondence

The following table lists Central Governance values and correspondence with middleware components.

Central Governance value	Transfer CFT CFTPART:SYST
aix-power-32 (default value)	UNIX
aix-power-64	UNIX
hpux-ia64-64	UNIX
hpux-parisc-32	UNIX

Central Governance value	Transfer CFT CFTPART:SYST
hpux-parisc-64	UNIX
linux-ia64-64	UNIX
linux-s390-32	UNIX
linux-s390-64	UNIX
linux-x86-32	UNIX
linux-x86-64	UNIX
sun-sparc-32	UNIX
sun-sparc-64	UNIX
sun-x86-32	UNIX
sun-x86-64	UNIX
win-ia64-64	WINNT
win-x86-32	WINNT
win-x86-64	WINNT
MVS	MVS
OS2200	OS2200
os400	OS400
vms-ia64	VMS
vms-alpha	VMS
Guardian-IA64	GUARD
Guardian-Risc	GUARD

Any other value that is found on Transfer CFT when importing the Transfer CFT partners or upgrading from Central Governance 1.0.2 to 1.1 is imported or migrated to aix-power-32.

Appendix C: SecureTransport corresponding fields

The following topics have tables that list and describe fields in Central Governance that correspond to fields in SecureTransport.

Protocol fields in Central Governance and SecureTransport

The following tables map protocol fields in Central Governance and SecureTransport.

PeSIT

Central Governance field	SecureTransport field	Central Governance values	Comment
Protocol	Operations > Server control > PeSIT Server > Enable PeSIT over <TCP/pTCP> <Secure/Plain>	PeSIT	PeSIT servers corresponding to the Central Governance server communication profiles are enabled and started on SecureTransport.
Port	Operations > Server control > PeSIT Server > Enable PeSIT over <TCP/pTCP> <Secure/Plain> Socket > Port	[1..65535]	

Central Governance field	SecureTransport field	Central Governance values	Comment
Network protocol	Operations > Server control > PeSIT Server > Enable PeSIT over: CG: TCP + Enable SSL/TLS = No => ST: Enable PeSIT over Plain Socket CG: TCP + Enable SSL/TLS = Yes => ST: Enable PeSIT over Secured Socket CG: pTCP + Enable SSL/TLS = No => ST: Enable PeSIT over pTCP Plain Socket CG: pTCP + Enable SSL/TLS = Yes => ST: Enable PeSIT over pTCP Secured Socket	TCP | pTCP	According to whether SSL is activated and according to the network protocol.
PeSIT login	N/A in the static configuration	Text, maximum 24 characters	
Password	N/A in the static configuration	Text, maximum 8 characters	Represents the PeSIT password corresponding to the PeSIT login used in flows when SecureTransport acts as PeSIT server.
Enable SSL/TLS	Operations > Server control > PeSIT Server > Enable PeSIT over TCP/pTCP over secured socket.	Yes | No	
Client authentication required	Access > Secure Socket Layer > Client Certificate Authentication > PeSIT CG: Yes - ST: Mandatory CG: No - ST: Disabled CG: Optional - ST: Optional	Yes | No | Optional	
Private certificate	Operations > Server control > PeSIT Server > Enable PeSIT over <TCP/pTCP> Secure Socket > SSL Key Alias	Use existing private certificate | Upload new private certificate	

Central Governance field	SecureTransport field	Central Governance values	Comment
Enable FIPS transfer mode	Operations > Server control > PeSIT Server > Enable FIPS Transfer Mode	Yes | No	If checked, it applies for all PeSIT servers enabled and started.

SFTP

Central Governance field	SecureTransport field	Central Governance values	Comment
Protocol	Operations > Server control > SSH Server > Enable Secure File Transfer Protocol (SFTP)	SFTP	SSH server corresponding to the Central Governance server communication profiles are enabled and started on SecureTransport.
Port	Operations > Server control > SSH Server > Enable Secure File Transfer Protocol (SFTP)	[1..65535]	
Client authentication	Access > Secure Socket Layer > Client Certificate Authentication > SSH CG: Public key - ST: Mandatory CG: Password - ST: Disabled CG: Password or Public key - ST: Optional	Public key (default) | Password | Password or Public key	
Server encryption	Operations > Server control > SSH Server > Enable Secure File Transfer Protocol > SSL Key Alias	Use existing private certificate | Upload new private certificate	
Enable FIPS transfer mode	Operations > Server control > SSH Server > Enable FIPS Transfer Mode	Yes | No	

FTP

Central Governance field	SecureTransport field	Central Governance values	Comment
Protocol	Operations > Server control > FTP Server > Enable PeSIT over <Secure/Plain>	FTP	FTP servers corresponding to the Central Governance server communication profiles are enabled and started on SecureTransport.
Port	Operations > Server control > FTP Server > Enable FTP over <Secure/Plain> Socket > Port	[1..65535]	
Passive port range	Setup > FTP Settings > FTP Passive Mode CG: From – ST: Base Port CG: To – ST: Port End ST: Number of Ports = (To - From + 1)	From / To: [1024..65535] From must be lower than To	
Enable SSL/TLS	Operations > Server control > PeSIT Server > Enable FTP over secured socket	Yes | No	
Client authentication required	Access > Secure Socket Layer > Client Certificate Authentication > FTPS CG: Yes - ST: Mandatory CG: No - ST: Disabled CG: Optional - ST: Optional	Yes | No | Optional	
Private certificate	Operations > Server control > PeSIT Server > Enable PeSIT over <TCP/pTCP> Secure Socket > SSL Key Alias	Use existing private certificate | Upload new private certificate	
Enable FIPS transfer mode	Operations > Server control > PeSIT Server > Enable FIPS Transfer Mode	Yes | No	If checked, it applies for all FTP servers enabled and started.

SecureTransport SFTP, FTP, HTTP flow definition

The following tables describe the fields and parameters available in flow definition in SecureTransport when it is used as a relay via SFTP, FTP or HTTP in Central Governance.

SecureTransport step definition mapping

The following tables describe fields in Central Governance and SecureTransport for received and send properties.

Receive properties: sender pushes files

Central Governance section	Central Governance field	SecureTransport field	Comment
File properties	Directory	Subscription > Subscription folder	The directory represents the path where the sender is able to push files on SecureTransport. If in the same account there is another subscription (created outside Central Governance) with the same folder, the flow deployment will fail. You must review the Directory value.
Post-reception actions	On failure	Subscription > Post Transmission Settings > On failure	Actions are applied to files that failed to arrive to the directory set before.

Receive properties: SecureTransport pulls files from the sender

Central Governance section	Central Governance field	SecureTransport field	Comment
File properties	Remote directory	Transfer Site > Download folder	Represents the folder on the sender from where SecureTransport pulls files.
File properties	File filterOptions: File Globbing/ Regular Expression	Download patternFile Globbing/ Regular Expression	Represents the filter on files SecureTransport pulls from the remote directory.

Central Governance section	Central Governance field	SecureTransport field	Comment
File properties	Directory	Subscription > Subscription folder	The directory represents where SecureTransport saves files pulled from the sender.
File properties	Scheduler	Subscription > For Files Received from this Account or its Partners > Schedule	Used to set a schedule for automatic retrieval of files from the sender. If the start date of the scheduler is earlier than the SecureTransport system data at deployment time, the flow deployment on SecureTransport fails.
Post-reception actions	On failure	Subscription > Post Transmission Settings > On failure	Actions are applied to files that failed to arrive to the directory set before.

Send properties: SecureTransport pushes files to receiver

Central Governance section	Central Governance field	SecureTransport field	Comment
File properties	Remote directory	Route Package > Route for receiver > Transfer settings > Step: Send to partner > Overwrite upload folder	Represents the folder on the receiver where SecureTransport pushes files.
File properties	File name sent	Route Package > Route for receiver > Step: Send to partner > Transfer settings > Route file as	Rename files to be routed to the receiver.
File properties	File filter:Options: File Globbing/Regular Expression	Route Package > Route for receiver > Step: Send to partner > File filter	If set, represents the filter on files SecureTransport pushes to the remote directory.
Post-sending actions	On failure	Subscription > Post Processing Settings > On failure	Actions are applied to files that failed to be sent to the receiver.
Post-sending actions	On success	Subscription > Post Processing Settings > On success	Actions are applied to files that were sent successfully to the receiver.

Send properties: receiver pulls files

Central Governance section	Central Governance field	SecureTransport field	Comment
Transfer properties	File exists	Route Package > Route for receiver > Transfer settings > Step: Publish to account > Target settings: Collision settings Cancel(Default) => Fail operation Overwrite => Replace existing file Rename existing file => Rename existing file Rename transferred file => Use a different file name to publish the file Append => Append to existing file	This field is used in detecting duplicate transfers on the remote directory.
File properties	Remote directory	Route Package > Route for receiver > Transfer settings > Step: Publish to account > Target settings > Folder	The directory represents the path where SecureTransport publishes files to the receiver.
File properties	Publish file as	Route Package > Route for receiver > Transfer settings > Step: Publish to account > Target settings > Publish file as	Represents the name of the published file.
File properties	File filter:Options: File Globbing/Regular Expression	Route Package > Route for receiver > Step: Publish to partner > File filter	If set, represents the filter on files SecureTransport publishes to partner.
Post-sending actions	On failure	Subscription > Post Processing Settings > On failure	Actions are applied to files that failed to be pulled by the receiver.
Post-sending actions	On success	Subscription > Post Processing Settings > On success	Actions are applied to files that were pulled successfully by the receiver.
Post-download actions	On success	Subscription > Post Client Download Actions > On success	Actions are applied to each file downloaded from the directory where SecureTransport received files from the sender.

Central Governance updates to SecureTransport objects

The following tables describe whether objects deployed on SecureTransport have updates available from Central Governance for transfers.

Account definition

The account represents a part of the flow that communicates directly with the SecureTransport relay: the sender that sends files to SecureTransport or the receiver that pulls files from SecureTransport.

The account is created on SecureTransport when:

- The sender pushes files to SecureTransport over SFTP, FTP or HTTP or SecureTransport pulls files from sender.
- The receiver pulls files from SecureTransport.

Field	Central Governance deployed value	Update from Central Governance
Name	If the account represents a part of the flow that acts as a client (push files to SecureTransport or pulls files from SecureTransport): client login. The value is taken from the protocol definition between SecureTransport and client. When SecureTransport pulls files from the sender, the account name is the name of the sender.	Yes
Email contact	Taken from the contact information of the part of the flow.	Yes
Phone contact	Taken from the contact information of the part of the flow.	Yes
Account Type:	Unspecified.	No
Business unit:	CentralGovernanceBusinessUnit.	Yes
Routing Mode	Ignore.	Yes
Encrypt Mode	Unspecified.	No
UID	Taken from SecureTransport settings: Setup > Central Governance > UID.	No
GID	Taken from SecureTransport settings: Setup > Central Governance > GID.	No
Home Folder	Taken from SecureTransport settings: Setup > Central Governance > Account Home Folder Prefix.	No

Field	Central Governance deployed value	Update from Central Governance
Home Folder Access Level	Business Unit.	No
Notes	N/A	No
Adhoc Settings: Delivery Method	Disabled.	No
Allow this account to login to SecureTransport Server	Is enabled only if the sender or receiver is an SFTP client for SecureTransport.	Yes
Login Name	The name of the account.	
Allow this account to login by email	No.	No
Allow this account to submit transfers using the Transfers RESTful API	No.	No
Password is stored locally (not in external directory)	Yes.	No
Password	If SecureTransport acts as server and the client authenticates via login and password: taken from the client communication profile from the protocol definition.	No*
Require user to change password on next login	Taken from SecureTransport settings: Setup > Central Governance > Expire password on account creation.	No

Field	Central Governance deployed value	Update from Central Governance
Require user to change password every X days	No.	No
Lock account after y failed login attempts	Taken from SecureTransport settings: Setup > Central Governance > Failed login attempts before account is locked.	No

* The certificate is updated if needed.

Transfer site definitions

See:

- *SFTP transfer site definition* on page 499
- *FTP transfer site definition* on page 501
- *HTTP transfer site definition* on page 504

Subscription

The subscription is always created in the sender account. It is created starting from the CentralGovernanceApplication.

Field	Central Governance deployed value	Update from Central Governance
Subscription folder	SecureTransport Step, Receive properties, Directory.	Yes
Automatically Retrieve Files From	Yes when SecureTransport pulls files from the sender. Otherwise No.	Yes
Automatically Retrieve Files From Transfer Site	The transfer site created for the sender only when SecureTransport pulls files from the sender.	Yes
Schedule	SecureTransport Step, Receive properties, Scheduler.	Yes
Transfer profile	<empty>	Yes
Post Transmission Settings		

Field	Central Governance deployed value	Update from Central Governance
On Temporary Failure	Delete.	No
On Failure	SecureTransport Step, Receive properties, Post-reception actions, On failure.	Yes
Routing Options Trigger Settings		
Trigger processing of files based on condition	No.	No
Submit for processing	All files in the subscription folder.	No
Post Processing Settings		
On Failure	SecureTransport Step, Send properties, Post-sending actions, On failure.	Yes
On Success	SecureTransport Step, Send properties, Post-sending actions, On failure.	Yes
Post Client Download Actions	SecureTransport Step, Send properties, Post-download actions, On failure.	Yes

Route package

The route package is always created in the sender account. It is created starting from the CentralGovernanceRouteTemplate. It contains information about how SecureTransport routes files received from senders to receivers.

Field	Central Governance deployed value	Update from Central Governance
Name	<flow name>	Yes
Description	Managed by Central Governance. Changing it can corrupt already deployed Central Governance flows.	Yes
Subscriptions	Link the subscriptions generated for each receiver.	Yes
Inherited Settings	Not set.	No
Specific Settings		

Field	Central Governance deployed value	Update from Central Governance
Execution Rule	All Matching Routes.	No
Routes	See *Route* on page 496.	Yes
Notifications		
Notify following e-mails on route failure	Disabled.	Yes
Notify following e-mails on route success	Disabled.	Yes

Route

A route in the route package corresponding to the flow is managed for each receiver.

Field	Central Governance deployed value	Update from Central Governance
Name	<sender name>-<receiver name>	Yes
Description	Managed by Central Governance. Changing it can corrupt already deployed Central Governance flows.	Yes
Condition	SecureTransport Step, File processing, Condition. If the user does not set a value in Central Governance, the value deployed on SecureTransport is **Always**.	No
Notifications		
Notify following e-mails on route failure	Disabled.	Yes
Notify following e-mails on route success	Disabled.	Yes
Step: Send to Partner	Managed only when the direction between SecureTransport and the receiver is sender pushes files.	Yes

Field	Central Governance deployed value	Update from Central Governance
File filter	SecureTransport Step, Send properties, File filter.	Yes
Proceed with route execution on step failure	N/A (SecureTransport default value: Yes)	No
Transfer Settings		
Select an account	If SecureTransport pushes files to the receiver over SFTP, FTP or HTTP, option is *Use current account*. Else (SecureTransport pushes files over PeSIT), option is Specify an account name; Account is <Receiver account>.	Yes
Account Transfer Site	The transfer site generated for the receiver.	Yes
Transfer Profile	Available only if the protocol between SecureTransport and receiver is PeSIT. The value is the transfer profile generated for the receiver	Yes
Configure advanced PeSIT settings	Enabled only if the protocol between SecureTransport and receiver is PeSIT.	Yes
Overwrite upload folder	SecureTransport Step, Send properties, File properties, Remote folder.	Yes
Route file as	SecureTransport Step, Send properties, File properties, Sent file as.	Yes
Send trigger file	N/A (SecureTransport default value: No).	No
Max number of parrallel transfers:	N/A (SecureTransport default value: 4).	No
Retry Settings	N/A (SecureTransport default value: 5).	No
Max number of retries:	N/A (SecureTransport default value: 3000).	No
Sleep between retries(in ms):	N/A (SecureTransport default value: 2000).	No

Field	Central Governance deployed value	Update from Central Governance
Sleep increment between retries(in ms)	N/A	No
Post Routing Action	N/A (SecureTransport default value: No)	No
Step: Publish to partner	Managed only when the direction between SecureTransport and the receiver is receiver pulls files.	Yes
File filter	SecureTransport Step, Send properties, File filter.	Yes
Proceed with route execution on step failure	N/A (SecureTransport default value: Yes).	No
Target Settings		
Account	Receiver account.	generated
Folder	SecureTransport Step, Send properties, File properties, Remote folder.	Yes
Publish File as	SecureTransport Step, Send properties, File properties, Publish File as.	Yes
Collision settings	SecureTransport Step, Send properties, Transfer properties, File exists.	Yes

SSH keys

When SecureTransport relay acts as a client, pushing files to the receiver or pulling files from the sender, and it must authenticate via login and SSH key , you define in the flow definition the SSH key SecureTransport uses for authentication.

- When SecureTransport pushes files to a receiver, the SSH key is imported in the private certificates list of the sender account and selected in the transfer site managed for defining the connection between SecureTransport and the receiver. Transfer site with name is <flowname><receiver name>.
- When SecureTransport pulls files from a sender, the SSH key is imported in the private certificates list of the sender account and selected in the transfer site managed for defining the connection between SecureTransport and the sender. Transfer site with name is <flowname><sender name>.

When SecureTransport relay acts as a server — sender pushes files to SecureTransport or receiver pulls files from SecureTransport — Central Governance does not deploy the SSH key. It is defined in the server communication profile, and SecureTransport already manages it. SecureTransport must have the client public key, which is imported in the in Login Certificates on the account created for the client:

- When sender pushes files to SecureTransport, the public key of the sender is imported in the sender account.
- When receiver pulls files from SecureTransport, the public key of the receiver is imported in the receiver account.

SFTP transfer site definition

The SFTP transfer site is created in SecureTransport when it acts as an SFTP client in the flow. When SecureTransport pulls files from the sender or pushes files to the receiver, the transfer site is created in the sender account.

Field	Central Governance deployed value	Update from Central Governance
Name	When SecureTransport pulls files from sender: <flow name><sender name>. When SecureTransport pushes files to receiver: <flow name><receiver name>.	Yes
Site type	Not set (default value : Unspecified).	No
Access Level:	Business Unit.	No
Transfer Protocol	SFTP.	No
Site settings		
Server	When SecureTransport pulls files from the sender, the server host is taken from the protocol definition between the sender and SecureTransport. When SecureTransport pushes files to the receiver, the server host is taken from the protocol definition between SecureTransport and the receiver.	Yes
Port	When SecureTransport pulls files from sender, the server port is taken from the protocol definition between the sender and SecureTransport. When SecureTransport pushes files to receiver, the server port is taken from the protocol definition between SecureTransport and the receiver.	Yes
Network Zone	Taken from the protocol definition in the communication profile of SecureTransport.	Yes
Download folder	When SecureTransport pushes files to receiver, taken from SecureTransport step, Send properties: Remote folder. Otherwise not set.	No
Download Pattern Type	When SecureTransport pulls files from sender, taken from SecureTransport step, Send properties: File filter. Otherwise not set.	Yes

Field	Central Governance deployed value	Update from Central Governance
Download pattern	When SecureTransport pulls files from sender, taken from SecureTransport step, Send properties: File filter. Otherwise not set.	Yes
Upload folder	When SecureTransport pushes files to receiver, taken from SecureTransport step, Receive properties: Remote folder. Otherwise not set.	Yes
Allow overwrite	Yes.	Yes
Transfer Settings		
Transfer Mode	When SecureTransport pulls files from sender, transfer mode taken from the protocol definition between the sender and SecureTransport. When ST pushes files to receiver, transfer mode taken from the protocol definition between SecureTransport and the receiver.	Yes
Verify Fingerprint for this Site	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Fingerprint	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Enable FIPS Transfer Mode	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Site Login Credentials		
User Name	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Password	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Use Password	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server: Yes if SecureTransport must use a password as SFTP client, otherwise No.	Yes

Field	Central Governance deployed value	Update from Central Governance
SSH Keys	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server: If SecureTransport must use an SSH key to connect to the SFTP server, the keys is imported in private certificates in the account and is selected in this field. Otherwise the field is empty.	Yes
Network Settings		
Connection Read/Write timeout:*	N/A	No
Connection Read/Write Buffer Size:*	N/A	No
Post Transmission Settings Send Options		
Send file as	N/A	No
On Temporary Failure	N/A	No
On Failure	N/A	No
On success	N/A	No
Post Transmission Settings Receive Options		
Receive file as	N/A	No
On Failure	N/A	No
On success	SecureTransport Step, Receive properties, Post-reception actions, On success	Yes

FTP transfer site definition

The FTP transfer site is created in SecureTransport when it acts as an FTP client in the flow. When SecureTransport pulls files from the sender or pushes files to the receiver, the transfer site is created in the sender account

Field	Central Governance deployed value	Update from Central Governance
Name	When SecureTransport pulls files from sender: <flow name><sender name>. When SecureTransport pushes files to receiver: <flow name><receiver name>.	Yes
Site type	Not set (default value : Unspecified).	No
Access Level	Business Unit.	No
Transfer Protocol	FTP(S)	No
Site settings		
Server	When SecureTransport pulls files from the sender, the server host is taken from the protocol definition between the sender and SecureTransport. When SecureTransport pushes files to the receiver, the server host is taken from the protocol definition between SecureTransport and the receiver.	Yes
Port	When SecureTransport pulls files from sender, the server port is taken from the protocol definition between the sender and SecureTransport. When SecureTransport pushes files to receiver, the server port is taken from the protocol definition between SecureTransport and the receiver.	Yes
Enable Active connection mode	Taken from sender's server communication profile, connection mode.	Yes
Network Zone	Taken from the protocol definition in the communication profile of SecureTransport.	Yes
Download folder	When SecureTransport pushes files to receiver, taken from SecureTransport step, Send properties: Remote folder. Otherwise not set.	No
Download Pattern Type	When SecureTransport pulls files from sender, taken from SecureTransport step, Send properties: File filter. Otherwise not set.	Yes
Download pattern	When SecureTransport pulls files from sender, taken from SecureTransport step, Send properties: File filter. Otherwise not set.	Yes

Field	Central Governance deployed value	Update from Central Governance
Upload folder	When SecureTransport pushes files to receiver, taken from SecureTransport step, Receive properties: Remote folder. Otherwise not set.	Yes
Allow overwrite	Yes.	Yes
Transfer settings		
Transfer Mode	When SecureTransport pulls files from sender, transfer mode taken from the protocol definition between the sender and SecureTransport. When SecureTransport pushes files to receiver, transfer mode taken from the protocol definition between SecureTransport and the receiver.	Yes
Transcode any line terminators in ASCII mode	Not set from Central Governance.	No
Use FTPS	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server. If between SecureTransport and sender: SSL=None => disabled, otherwise=Yes.	Yes
Verify certificate for this Site	Taken from protocol definition between SecureTransport and partner: is SSL is required: enabled, else disabled. If in protocol between SecureTransport and sender: SSL = Mutual Authentication => enabled else , taken from SecureTransport client communication profile with the sender.	Yes
Clear Command Channel	Not set.	No
Enable FIPS Transfer Mode	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Site command	Not set.	No
Site login credentials		

Field	Central Governance deployed value	Update from Central Governance
User Name	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Password	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Use Password	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server: Yes if SecureTransport must use a password as client, otherwise No.	Yes
Certificate	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server: If SecureTransport must use a certificate to connect to the server, the certificate is imported in private certificates in the account and is selected in this field. It is also added in the Trusted certificates of SecureTransport. Otherwise the field is empty.	Yes
Send options		
Send file as	N/A	No
On Temporary Failure	N/A	No
On Failure	N/A	No
On success	N/A	No
Receive options		
Receive file as	N/A	No
On Failure	N/A	No
On success	SecureTransport Step, Receive properties, Post-reception actions, On success	Yes

HTTP transfer site definition

The HTTP transfer site is created in SecureTransport when it acts as an HTTP client in the flow. When SecureTransport pulls files from the sender or pushes files to the receiver, the transfer site is created in the sender account

Field	Central Governance deployed value	Update from Central Governance
Name	When SecureTransport pulls files from sender: <flow name><sender name>. When SecureTransport pushes files to receiver: <flow name><receiver name>.	Yes
Site type	Not set (default value : Unspecified).	No
Access Level:	Business Unit.	No
Transfer Protocol	HTTP(S)	No
Site settings		
Specify partner using host name (IP address) and port number	Taken from the sender's server communication profile: If in the sender's server communication profile: host, port => Enabled, else Disabled.	
Server	When SecureTransport pulls files from the sender, the server host is taken from the protocol definition between the sender and SecureTransport. When SecureTransport pushes files to the receiver, the server host is taken from the protocol definition between SecureTransport and the receiver.	Yes
Port	When SecureTransport pulls files from sender, the server port is taken from the protocol definition between the sender and SecureTransport. When SecureTransport pushes files to receiver, the server port is taken from the protocol definition between SecureTransport and the receiver.	Yes
Specify partner using URL	Taken from the sender's server communication profile: If in the sender's server communication profile: URL => Enabled, else Disabled.	
URL	When SecureTransport pulls files from the sender, the server URL is taken from the protocol definition between the sender and SecureTransport. When SecureTransport pushes files to the receiver, the server URL is taken from the protocol definition between SecureTransport and the receiver.	
Network Zone	Taken from the protocol definition in the communication profile of SecureTransport.	Yes

Field	Central Governance deployed value	Update from Central Governance
Download folder	When SecureTransport pushes files to receiver, taken from SecureTransport step, Send properties: Remote folder. Otherwise not set.	No
Download Pattern Type	When SecureTransport pulls files from sender, taken from SecureTransport step, Send properties: File filter. Otherwise not set.	Yes
Download pattern	When SecureTransport pulls files from sender, taken from SecureTransport step, Send properties: File filter. Otherwise not set.	Yes
Upload folder	When SecureTransport pushes files to receiver, taken from SecureTransport step, Receive properties: Remote folder. Otherwise not set.	Yes
Allow overwrite	Yes.	Yes
Transfer settings		
Transfer Mode	When SecureTransport pulls files from sender, transfer mode taken from the protocol definition between the sender and SecureTransport. When SecureTransport pushes files to receiver, transfer mode taken from the protocol definition between SecureTransport and the receiver.	Yes
Use HTTPS	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server. If between SecureTransport and sender: SSL=None => disabled, otherwise=Yes.	Yes
Verify certificate for this Site	Taken from protocol definition between SecureTransport and partner: is SSL is required: enabled, else disabled. If in protocol between SecureTransport and sender SSL = Mutual Authentication => enabled else , taken from SecureTransport client communication profile with the sender.	Yes
Enable FIPS Transfer Mode	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Site command	Not set.	No
Site login credentials		

Field	Central Governance deployed value	Update from Central Governance
User Name	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Password	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server.	Yes
Use Password	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server: Yes if SecureTransport must use a password as a client, otherwise No.	Yes
Certificate	Taken from the SecureTransport client communication profile protocol definition between SecureTransport and the server: If SecureTransport must use a certificate to connect to the server, the certificate is imported in private certificates in the account and is selected in this field. It is also added in the Trusted certificates of SecureTransport. Otherwise the field is empty.	Yes
Send options		
Send file as	N/A	No
On Temporary Failure	N/A	No
On Failure	N/A	No
Receive options		
Receive file as	N/A	No
On failure	N/A	No
On success	SecureTransport Step, Receive properties, Post-reception actions, On success	Yes

SecureTransport PeSIT flow definition

The following tables describe the fields and parameters available in flow definition in SecureTransport when it is used as relay via PeSIT in Central Governance.

SecureTransport step definition mapping

The following tables describe fields in Central Governance and SecureTransport for receive and send properties.

Receive properties: PESIT, sender pushes files

Central Governance section	Central Governance field	SecureTransport field	Comment
File properties	Receive file as	Transfer profile > Receive File As	The field must be used to name the files received when files are transferred to SecureTransport.
File properties	File type	Transfer profile > Transfer Mode	Specifies the file type.
File properties	Record type	Transfer profile > Record Format	Indicates whether the records in the file are fixed or variable length.
File properties	Max record length	Transfer profile > Record Length	Specify the maximum length of the records in bytes.
Post-reception actions	On failure	Subscription > Post Transmission Settings > On failure	Actions are applied to files that failed to arrive to the directory set previously.

Receive properties: PESIT, SecureTransport pulls files from the sender

Central Governance section	Central Governance field	SecureTransport field	Comment
File properties	Receive file as	Transfer profile > Receive File As	The field must be used to name the files received when files are transferred to SecureTransport.

Central Governance section	Central Governance field	SecureTransport field	Comment
File properties	Files to receive	Transfer profile > All files	Specify if SecureTransport must pull all files available or only the first one.
File properties	File type	Transfer profile > Transfer Mode	Specifies the file type.
File properties	Record type	Transfer profile > Record Format	Indicates whether the records in the file are fixed or variable in length.
File properties	Max record length	Transfer profile > Record Length	Specify the maximum length of the records in bytes.
File properties	Scheduler	Subscription > For Files Received from this Account or its Partners > Schedule	Used to set a schedule for automatic retrieval of files from the sender. If the scheduled start date is later than the Secure Transport system date, the flow fails to deploy on SecureTransport.
Post-reception actions	On failure	Subscription > Post Transmission Settings > On failure	Actions are applied to files that failed to arrive to the directory set previously.

Send properties: PESIT, sender pushes files

Central Governance section	Central Governance field	SecureTransport field	Comment
File properties	File name sent	Route Package > Route for receiver > Step: Send to partner > Configure advanced PeSIT settings: Transfer profile > File label	The File label field is used to override the file label (PI37) predefined in the transfer profile.
File properties	Condition	Route Package > Route for receiver > Condition	Specify a condition to route files to receiver.
File properties	File type	Route Package > Route for receiver > Step: Send to partner > Configure advanced PeSIT settings: Transfer profile > Transfer mode	Specifies the file type.

Central Governance section	Central Governance field	SecureTransport field	Comment
File properties	Record type	Route Package > Route for receiver > Step: Send to partner > Configure advanced PeSIT settings: Transfer profile > Record format	Indicates whether the records in the file are fixed or variable length.
File properties	Max record length	Route Package > Route for receiver > Step: Send to partner > Configure advanced PeSIT settings: Transfer profile > Record length	Specify the maximum length of the records in bytes. Default value = 2048
Post-sending actions	On failure	Subscription > Post Processing Settings > On failure	Actions are applied to files that failed to arrive to the directory where SecureTransport received the files from the sender after they were routed to a receiver.
Post-sending actions	On success	Subscription > Post Processing Settings > On success	Actions are applied to files that properly arrived to the directory where SecureTransport received the files from the sender after they were routed to a receiver.

Send properties: PESIT, receiver pulls files

Central Governance section	Central Governance field	SecureTransport field	Comment
Transfer properties	File exists	Route Package > Route for receiver > Transfer settings > Step: Publish to account > Target settings : Collision settings Cancel(Default) => Fail operation Overwrite => Replace existing file Rename existing file => Rename existing file Rename transferred file => Use a different file name to publish the file Append => Append to existing file	This field is used in detecting duplicate transfers on the remote directory.

Central Governance section	Central Governance field	SecureTransport field	Comment
Transfer properties	User message	Route Package > Route for receiver > Step: Send to partner > Configure advanced PeSIT settings: User message	The User message field is used to override the predefined user message (PI99) in the PeSIT transfer site.
File properties	File name sent	Route Package > Route for receiver > Step: Send to partner > Configure advanced PeSIT settings: File label	The File label field is used to override the file label (PI37) predefined in the transfer profile.
File properties	Condition	Route Package > Route for receiver > Condition	Specify a condition to route files to receiver.
File properties	File type	Route Package > Route for receiver > Step: Send to partner > Configure advanced PeSIT settings: Transfer mode	Specifies the file type.
File properties	Record type	Route Package > Route for receiver > Step: Send to partner > Configure advanced PeSIT settings: Record format	Indicates whether the records in the file are fixed or variable length.
File properties	Max record length	Route Package > Route for receiver > Step: Send to partner-> Configure advanced PeSIT settings: Record length	Specify the maximum length of the records in bytes. Defualt value = 2048
Post-sending actions	On failure	Subscription > Post Processing Settings > On failure	Actions are applied to files that failed to arrive to the directory where SecureTransport received the files from the sender after they were routed to a receiver.
Post-sending actions	On success	Subscription > Post Processing Settings > On success	Actions are applied to files that properly arrived to the directory where SecureTransport received the files from the sender after they were routed to a receiver.
Post-download actions	On success	Subscription > Post Client Download Actions > On success	Actions are applied to each file downloaded from the directory where SecureTransport received files from sender.

Central Governance updates to SecureTransport objects

The following tables describe whether objects deployed on SecureTransport have updates available from Central Governance for PeSIT transfers.

Account definition

The account represents a part of the flow that communicates directly with the SecureTransport relay from:

- The sender that sends files to SecureTransport.
- The receiver that pulls files from SecureTransport.

Generally , the account is managed on SecureTransport when:

- The sender pushes files to SecureTransport or SecureTransport pulls files from sender.
- The receiver pulls files from SecureTransport.

In addition, for PeSIT the account is also managed when SecureTransport pushes files to receiver over PeSIT, or the sender pushes files to SecureTransport over PeSIT.

Field	Deployed value from Central Governance	Update from Central Governance
Name	The SecureTransport identifier (PeSIT login).	Yes
Email contact	Taken from the contact information in the flow.	Yes
Phone contact	Taken from the contact information in the flow.	Yes
Account type:	Unspecified	No
Business unit:	**CentralGovernanceBusinessUnit**	Yes
Routing Mode	Ignore	Yes
Encrypt Mode	Unspecified	No
UID	Taken from SecureTransport settings: Setup > Central governance > UID	No
GID	Taken from SecureTransport settings: Setup > Central governance > GID	No
Home Folder	Taken from SecureTransport settings: Setup > Central governance > Account Home Folder Prefix	No
Home Folder Access Level	Business Unit	No

Field	Deployed value from Central Governance	Update from Central Governance
Notes	N/A	No
Adhoc Settings: Delivery Method	Disabled	No
Allow this account to log in to SecureTransport Server	Is enabled only if the sender or receiver is a client for SecureTransport.	Yes
Login Name	The name of the account	
Allow this account to login by email	No	No
Allow this account to submit transfers using the Transfers RESTful API	No	No
Password is stored locally (not in external directory)	Yes	No
Password	If SecureTransport acts as a server and the client authenticates via the login and password, the value is taken from the client communication profile from the protocol definition. If SecureTransport acts as a server and the client authenticates via the login and public key, the password is randomly generated strong enough. In this case the client key is imported in the account login certificates.	No. The certificate is updated if needed.
Require user to change password on next login	Taken from SecureTransport settings: Setup > Central governance > Expire password on account creation	No
Require user to change password every X days	No	No
Lock account after Y failed login attempts	Taken from SecureTransport settings: Setup > Central governance > Failed login attempts before account is locked.	No

PESIT transfer site definition

The PESIT transfer site is created on SecureTransport for all senders and receivers that send or receive data to or from SecureTransport via PeSIT.

Field	Deployed value from Central Governance	Update from Central Governance
Name	The sender or receiver PeSIT identifier (login)	Yes
Site type	Not set (default value : Unspecified)	No
Access Level:	Business Unit	No
Transfer Protocol	PeSIT	No
Remote Partner Settings		
Server	Set only when SecureTransport partner acts as a client: Server port . Taken from the communication profile of a partner (sender or receiver), from the protocol definition.	Yes
Port	Set only when SecureTransport partner acts as a client: Server port Taken from the communication profile of a partner (sender or receiver), from the protocol definition.	Yes
Network Zone	Taken from the communication profile of SecureTransport from the protocol definition.	Yes
Transfer Settings		
Compression	Taken from the communication profile of a partner (sender or receiver), from the protocol definition.	Yes
Network Settings		
Parallel TCP connections:	If in Central Governance PeSIT protocol definition, the network protocol is `pTCP, 10` Otherwise it is `10`	Yes
USE TLS	Taken from the communication profile of partner (sender or receiver), from the protocol definition.	Yes

Field	Deployed value from Central Governance	Update from Central Governance
Verify certificate for this Site	If Mutual Authentication is Yes, Otherwise it is taken from the communication profile of a partner (sender or receiver), from the protocol definition.	
Enable FIPS Transfer Mode	Taken from the communication profile of partner (sender or receiver) from the protocol definition	
Enable Legacy Transfer CFT compatible SSL Mode	No	No

PESIT transfer profile definition

The PESIT transfer profile is created on SecureTransport for each PeSIT identifier configured between SecureTransport and partners over PeSIT (sender or receiver).

Sender pushes files to SecureTransport over PeSIT

Field	Deployed value from Central Governance	Update from Central Governance
Name	The PeSIT identifier configured between the sender and SecureTransport	Yes
Files To Send	N/A	No
Receive File As	/<PeSIT identifier between sender and SecureTransport>/<the value from SecureTransport step, Receive properties, Receive file as>	Yes
Acknowledge transfer	Taken from the protocol definition between sender and SecureTransport	Yes
File Label	N/A	No
All files	SecureTransport step, Receive properties > Files to receive	Yes
Transfer Mode	SecureTransport step, Receive properties > File type	Yes
Record Format	SecureTransport step, Receive properties > Record type	Yes

Field	Deployed value from Central Governance	Update from Central Governance
Record Length	SecureTransport step, Receive properties > Maximum record length	Yes

SecureTransport pulls files from sender

Field	Deployed value from Central Governance	Update from Central Governance
Name	The PeSIT identifier configured between the sender and SecureTransport	Yes
Files To Send	N/A	No
Receive File As	/<PeSIT identifier between sender and SecureTransport>/ **{pesit.senderID}**/<the value from SecureTransport step, Receive properties, Receive file as>	Yes
Acknowledge transfer	N/A	Yes
File Label	N/A	No
All files	SecureTransport step, Receive properties > Files to receive	Yes
Transfer Mode	SecureTransport step, Receive properties > File type	Yes
Record Format	SecureTransport step, Receive properties > Record type	Yes
Record Length	SecureTransport step, Receive properties > Maximum record length	Yes

SecureTransport pushes files to receiver

Field	Deployed value from Central Governance	Update from Central Governance
Name	The PeSIT identifier configured between SecureTransport and receiver	Yes

Field	Deployed value from Central Governance	Update from Central Governance
Files To Send	/<PeSIT identifier between sender and SecureTransport>/ *	No
Receive File As	N/A	Yes
Acknowledge transfer	N/A	Yes
File Label	N/A *	No
All files	N/A	Yes
Transfer Mode	N/A *	Yes
Record Format	N/A *	Yes
Record Length	N/A *	Yes

The PeSIT properties values are set in the Send to partner step that is defined in the route between original sender and receiver.

Receiver pulls from SecureTransport

Field	Deployed value from Central Governance	Update from Central Governance
Name	The PeSIT identifier configured between SecureTransport and receiver	Yes
Files To Send	/<PeSIT identifier configured between SecureTransport>/ and receiver>/**${pesit.receiverID}**/*	No
Receive File As	N/A	Yes
Acknowledge transfer	N/A	Yes
File Label	SecureTransport step, Send properties > Files name sent	No
All files	SecureTransport step, Send properties > Files to send	Yes
Transfer Mode	SecureTransport step, Send properties > File type	Yes
Record Format	SecureTransport step, Send properties > Record type	Yes

Field	Deployed value from Central Governance	Update from Central Governance
Record Length	SecureTransport step, Send properties > Maximum record length	Yes

SecureTransport general definitions in flows

The following definitions apply in SecureTransport for all protocols.

Subscription

The subscription is always created in the sender account. It is created from the CentralGovernanceApplication.

Field	Deployed value from Central Governance	Update from Central Governance
Subscription folder	If the protocol is PeSIT: • If sender pushes files to SecureTransport: /<the PeSIT identifier between SecureTransport and sender>. • If SecureTransport pulls files from sender: /<the PeSIT identifier between SecureTransport and sender>/<the sender PeSIT login>. Otherwise, the value is from SecureTransport step, receive properties, directory.	Yes
Automatically Retrieve Files From	Yes when SecureTransport pulls files from sender, otherwise No	Yes
Automatically Retrieve Files From Transfer Site	Only when SecureTransport pulls files from sender: The transfer site created for the sender.	Yes
Schedule	SecureTransport Step, Receive properties, Scheduler	Yes
Transfer profile	If the protocol is PeSIT, the value is the transfer profile created for the files received from the sender. Otherwise the value is <empty>	Yes
Post Transmission Settings		
On Temporary Failure	Delete	No

Field	Deployed value from Central Governance	Update from Central Governance
On Failure	SecureTransport Step, Receive properties, Post-reception actions, On failure	Yes
Routing options trigger settings		
Trigger processing of files based on condition	No	No
Submit for processing	All files in the subscription folder	No
Post-processing settings		
On Failure	SecureTransport Step, Send properties, Post-sending actions, On failure	Yes
On Success	SecureTransport Step, Send properties, Post-sending actions, On failure	Yes

Route package

The route package is always created in the sender account. It is created from the CentralGovernanceRouteTemplate. It contains information about how SecureTransport routes files received from senders to receivers. You create a route package for each receiver.

Field	Deployed value from Central Governance	Update from Central Governance
Name	SFTP, FTP,HTTP: <flow name> PeSIT: <PeSIT identifier set between sender and SecureTransport>	Yes
Description	Managed by Central Governance. Changing it can corrupt already deployed Central Governance flows.	Yes
Subscriptions	Link the subscriptions generated for each receiver	Yes
Inherited Settings	Not set	No
Specific settings		

Field	Deployed value from Central Governance	Update from Central Governance
Execution Rule	All matching routes	No
Routes	See *Route* on page 520 table	Yes
Notifications		
Notify following e-mails on route failure	Disabled	Yes
Notify following e-mails on route success	Disabled	Yes

Route

A route in the route package corresponding to the flow is managed for each receiver.

Field	Deployed value from Central Governance	Update from Central Governance
Name	<sender name>-<receiver name> When SecureTransport is the destination of a store-and-forward path, the sender name is original sender name, the initiator of the path. When SecureTransport is the initiator of a store-and-forward path, the receiver name is the final destination name.	Yes
Description	Managed by Central Governance. Changing it can corrupt already deployed Central Governance flows.	Yes
Condition	SecureTransport step, File properties, Condition. If PeSIT, the following is automatically added to the value set in flow definition: When SecureTransport is the target of a store-and-forward path: ${(pesit.pi.originalsenderID.toUpperCase() eq '<original sender PeSIT login>')} Else: ${(pesit.pi.senderID.toUpperCase() eq '<sender PeSIT login>')}	Yes
Notifications		
Notify following e-mails on route failure	Disabled	Yes

Field	Deployed value from Central Governance	Update from Central Governance
Notify following e-mails on route success	Disabled	Yes
Step: send to partner	Managed only when the direction between SecureTransport and the receiver is sender pushes files.	Yes
File filter	SecureTransport Step, Send properties, File filter	Yes
Proceed with route execution on step failure	N/A (SecureTransport default value: Yes)	No
Transfer settings		
Select an account	If SecureTransport pushes files to the existing receiver over SFTP, FTP or HTTP, the option is *Use current account* Otherwise, (SecureTransport pushes files over PeSIT and the receiver account is different to the sender account), the option is *Specify an account name* where Account is the *<Receiver account>*.	Yes
Account Transfer Site	The transfer site generated for the receiver.	Yes
Transfer Profile	Available only if the protocol between SecureTransport and receiver is PeSIT. The value is the transfer profile generated for the receiver.	Yes
Configure advanced PeSIT settings	Enabled only if the protocol between SecureTransport and receiver is PeSIT.	Yes
Store and Forward mode	Start New	Yes
Virtual File Name	N/A	No

Field	Deployed value from Central Governance	Update from Central Governance
Data Encoding	SecureTransport Step, SEND Properties, File type	Yes
Record Format	SecureTransport Step, SEND Properties, Record type	Yes
Record Length	SecureTransport Step, SEND Properties, Max record length	Yes
File Label	SecureTransport Step, SEND Properties, File name sent	Yes
Final Destination	When SecureTransport is the initiator of the store-and-forward path, the value is equal to the PeSIT login of the store-and-forward destination.	Yes
User Message	SecureTransport Step, SEND Properties, User message	Yes
Overwrite upload folder	If the protocol is PeSIT between SecureTransport and the receiver, then the value is: /<flow id>/<receiver name>/*. Otherwise, the value is: SecureTransport Step, Send properties, File properties, Remote folder.	Yes
Route file as	SecureTransport Step, Send properties, File properties, Sent file as	Yes
Send trigger file	N/A (SecureTransport default value: No)	No
Max number of parallel transfers:	N/A (SecureTransport default value: 4)	No
Retry Settings	N/A (SecureTransport default value: 5)	No
Max number of retries:	N/A (SecureTransport default value: 3000)	No
Sleep between retries(in ms):	N/A (SecureTransport default value: 2000)	No

Field	Deployed value from Central Governance	Update from Central Governance
Sleep increment between retries(in ms)	N/A	No
Post Routing Action	N/A (SecureTransport default value: No)	No
Step: Publish to partner	Managed only when the direction between SecureTransport and receiver is receiver pulls files.	Yes
File filter	SecureTransport Step, Send properties, File filter	Yes
Proceed with route execution on step failure	N/A (SecureTransport default value: Yes)	No
Target settings		
Account	Receiver account	generated
Folder	If PeSIT: /<The PeSIT identifier configured between the ST and receiver>/**<receiver PeSIT login>**/* Otherwise: SecureTransport Step, Send properties, File properties, Remote folder	Yes
Publish File as	SecureTransport Step, Send properties, File properties, Publish File as	Yes
Collision settings	SecureTransport Step, Send properties, Transfer properties, File exists	Yes

Certificates used for authentication

When SecureTransport relay acts as a client, pushing files to a receiver or pulling files from the sender, and when it requires authentication via a certificate, you must define in the flow definition the certificate SecureTransport uses for authentication as the client.

When SecureTransport pushes files to a receiver, the certificate is imported in the private certificates list of the sender account and selected in the transfer site managed for defining the connection between SecureTransport and the receiver (transfer site with name is <flow name><receiver name>).

When SecureTransport pulls files from sender, the certificate is also imported in the private certificates list of the sender account and selected in the transfer site managed for defining the connection between SecureTransport and sender (transfer site with name is <flow name><sender name>).

The root certificate is imported in the SecureTransport trusted certificates.

When SecureTransport relay acts as a server (sender pushes files to SecureTransport or receiver pulls files from SecureTransport), Central Governance does not deploy the certificate. It is defined in the server communication profile, and SecureTransport already manages it. SecureTransport must have the client certificate of the client and it is imported in Login Certificates on the account created for the client:

- When sender pushes files to SecureTransport, the certificate of the sender is imported in the sender account.
- When receiver pulls files from SecureTransport, the certificate of the receiver is imported in the receiver account.

Certificates used with FTP, HTTP, PeSIT

When SecureTransport relay acts as a FTP, HTTP or PeSIT client, pushing files to the receiver or pulling files from the sender, and it must authenticate via a certificate, you define in the flow definition the certificate SecureTransport uses for authentication.

When SecureTransport pushes files to a receiver, the certificate is imported to the private certificates list of the sender account and selected in the transfer site managed for defining the connection between SecureTransport and the receiver.

When SecureTransport pulls files from a sender, the certificate is imported to the private certificates list of the sender account and selected in the transfer site managed for defining the connection between SecureTransport and the sender.

The SecureTransport certificate and the partner's certificate are added to the list in SecureTransport of trusted certificates, if they are new certificates.

When SecureTransport relay acts as a server — sender pushes files to SecureTransport or receiver pulls files from SecureTransport — Central Governance does not deploy the certificate. The certificate is defined in the server communication profile, and SecureTransport already manages it. SecureTransport must have the client public certificate, which is imported to the Login Certificates on the account created for the client:

- When the sender pushes files to SecureTransport, the public certificate of the sender is imported to the sender account.
- When the receiver pulls files from SecureTransport, the public certificate of the receiver is imported to the receiver account.
- The certificate, from the partner's client communication profiles, is added to the list in SecureTransport of trusted certificates, if new certificates.
- If the incoming flow is over edges, the certificate also is imported to the list in the edge of trusted certificates.

Folder monitoring when SecureTransport is source in flow

SecureTransport can be the source of the flow when it is linked with an application that is selected as source of the flow.

In this case, an account is managed on SecureTransport that has the name of the application used in the flow and it does not have permissions to log in to SecureTransport server (Allow this account to login to SecureTransport Server property is not enabled). A transfer site of type Folder Monitor is managed in this account with the definition of the download settings. A subscription on this account will automatically retrieves files from the directories configured in the Transfer Site of Folder Monitor type.

The Folder Monitor transfer site is created in SecureTransport when it acts as source in the flow. It is created in the account that represents the source application.

Field	Central Governance deployed value	Update from Central Governance
Name	<flow name>	Yes
Site type	Not set (default value : Unspecified).	No
Access Level	Business Unit.	No
Transfer Protocol	Folder Monitor.	No
Download settings		
Download Folder	Taken from SecureTransport step, Receive properties, Folder Monitoring: Directory to scan	Yes
Download File Filter	Taken from SecureTransport step, Receive properties, Folder Monitoring: File Filter	Yes
Download pattern	Taken from SecureTransport step, Receive properties, Folder Monitoring: File Filter	Yes
Subfolder Monitoring, Do Not Monitor Subfolders	Taken from SecureTransport step, Receive properties, Folder Monitoring: Scan sub-directories = No	Yes
Subfolder Monitoring, Monitor All Subfolders	Taken from SecureTransport step, Receive properties, Folder Monitoring: Scan sub-directories = Yes, Directory depth=Unlimited	Yes

Field	Central Governance deployed value	Update from Central Governance
Subfolder Monitoring, Subfolder Monitoring, Monitor All Subfolders	Taken from SecureTransport step, Receive properties, Folder Monitoring: Scan sub-directories = Yes, Directory depth=Limited, Directories	Yes
Subfolder Monitoring, Download Subfolder Pattern Type	Taken from SecureTransport step, Receive properties, Folder Monitoring, Scan sub-directories = Yes, Sub-directory filter	Yes
Subfolder Monitoring, Download Subfolder Pattern	Taken from SecureTransport step, Receive properties, Folder Monitoring, Scan sub-directories = Yes, Sub-directory filter	Yes
Post Transmission Settings, Receive file as	Taken from SecureTransport step, Receive properties, File properties, Receive file as	Yes
Upload settings	Not set when ST is source	

Folder monitoring when SecureTransport is target in flow

SecureTransport can be the target of the flow when it is associated with an application selected as the target in the flow. The transfer site of type Folder Monitor is managed in the account for the sender of the file. A route between the sender and the application target configures the processing of the files and the routing to the destination directory.

The folder monitor transfer site is created in SecureTransport when it acts as target in the flow. It is created in the sender account.

Field	Central Governance deployed value	Update from Central Governance
Name	<flow name>	Yes
Site type	Not set (default value : Unspecified).	No
Access Level	Business Unit.	No
Transfer Protocol	Folder Monitor.	No
Download settings	Set when SecureTransport is target	
Upload settings upload folder		

Field	Central Governance deployed value	Update from Central Governance
Folder	Taken from SecureTransport step, Send properties, Upload directory	Yes
Expression Language support for upload folder	Taken from SecureTransport step, Send properties, Use expression lamguage	Yes
Automatically create upload folder if it doesn't exist	Taken from SecureTransport step, Receive properties, Create directory if not existent	Yes
Allow overwrite	Yes	No
Post-transmission settings		
Send file as	Taken from SecureTransport step, Send properties, Upload file as	Yes
On Failure	Taken from SecureTransport step, Send properties, On failure	Yes
On Success	Taken from SecureTransport step, Send properties, On success	Yes

Appendix D: Flows deployed on SecureTransport

When SecureTransport is used as a relay in flows, the deployment of the flows on SecureTransport depends on:

- The protocol definition between SecureTransport and the sender or receiver.
- The flow properties defined on the SecureTransport relay step.

Objects created in SecureTransport

The following are the objects created in SecureTransport for sender and receiver when Central Governance deploys flows to it.

Account for the sender

If the sender must connect to SecureTransport, acting as a client pushing files to SecureTransport, the option *Allow this account to login to SecureTransport Server* is enabled.

The name of the sender account is generated according to the protocol definition between the sender and SecureTransport:

- If the protocol between source and SecureTransport is PeSIT, the name of the account is the PeSIT identifier that SecureTransport uses for the sender.
- If the protocol between source and SecureTransport is SFTP or FTP and the sender presents a login to authenticate, the name of the account is the login. Otherwise, it is the name of the sender.

Account for the receiver

For protocols SFTP, FTP and HTTP protocols, the account is created only if the receiver acts as a client and must connect to SecureTransport.

For the PeSIT protocol, the account always is created.

The name of the receiver account follows the same rules as for the sender account name. It is generated according to the protocol definition between SecureTransport and the receiver:

- If the protocol between source and SecureTransport is PeSIT, the name of the account is the PeSIT identifier that SecureTransport uses for the receiver.
- If the protocol between source and SecureTransport is SFTP or FTP and the receiver presents a login to authenticate, the name of the account is the login. Otherwise, it is the name of the receiver.

Objects deployed in sender account

The following objects are deployed in the sender account.

Transfer sites

- For all cases when SecureTransport must connect to the sender and pull files from the sender over SFTP, FTP and HTTP.
- For all the cases when SecureTransport must connect to the receiver and push files to the receiver over SFTP, FTP and HTTP.
- Always when PeSIT is the protocol between the sender and SecureTransport.

The name of the transfer site depends on the protocol definition between SecureTransport and the part of the flow for which the transfer site is created:

- For PeSIT, it represents the PeSIT identifier that the partner presents to SecureTransport.
- For SFTP, FTP and HTTP, it represents the <flow name value><partner name>.

For one flow there can be multiple transfer sites generated to the sender account. For example, sender pushes to SecureTransport via PeSIT and SecureTransport pushes to receiver via SFTP. In this case the sender account has one transfer site for the sender and one for the receiver.

Transfer profile

- Only when PeSIT is the protocol between sender and SecureTransport.
- The name of the transfer profile is the PeSIT identifier configured in the protocol between sender and SecureTransport.

Subscription

- Always created in the sender account. It allows the trigger of the transfers routing from sender to receiver.
- For SFTP, FTP and HTTP, the subscription folder is taken from the SecureTransport step flow definition between each sender and SecureTransport send properties.
- For PeSIT, the subscription folder is generated: `/<PeSIT identifier of the flow>`.

Route package

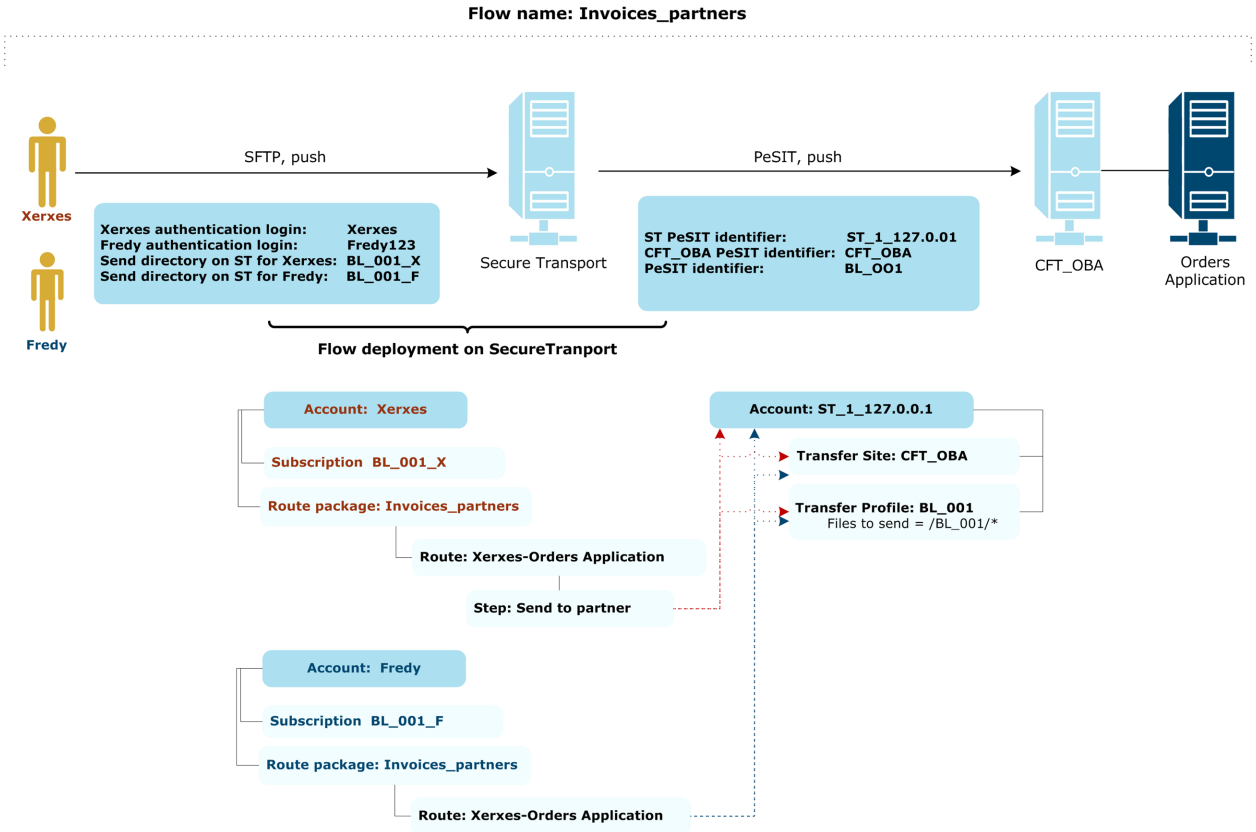
- Always created in the sender account. It allows routing files received in the subscription folder from sender to receiver. A route package is created for each receiver defined in the flow.
- The name of the route package. For SFTP, FTP, HTTP: <flow name>. For PeSIT: <PeSIT identifier between sender and SecureTransport>.
- The subscription is linked to the route package.

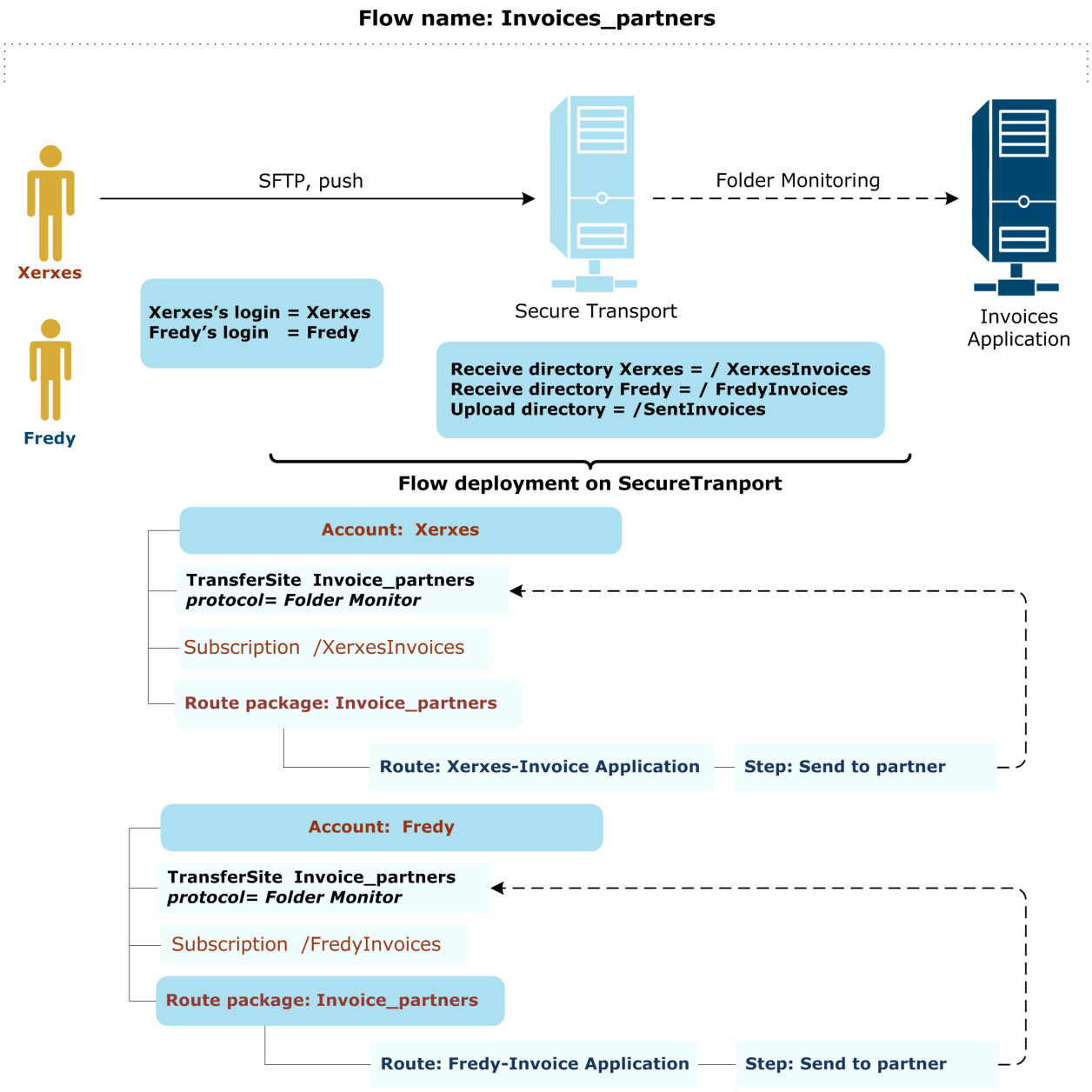
- A route is created for the receiver for each route package created:
  - The name of the route is <sender name><receiver name>.
  - It contains steps according to the file processing defined in the SecureTransport step flow definition between SecureTransport and the receiver. The last step is Send to Partner, if SecureTransport pushes files to the receiver, or Publish to Account, if the receiver pulls files from SecureTransport. (Send to Partner and Publish to Account are types of steps defined in SecureTransport.)

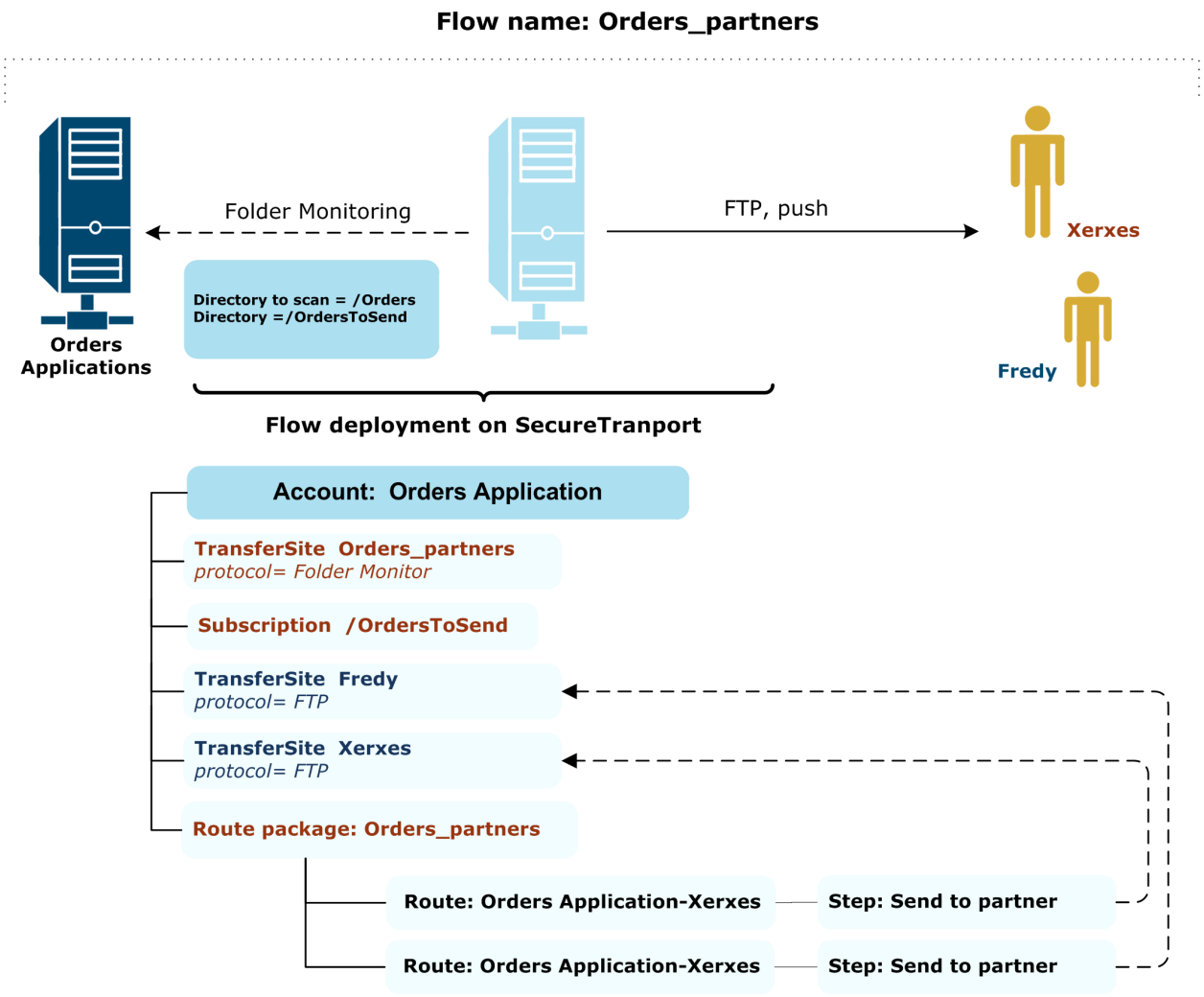
Objects deployed in receiver account: Subscription

- Central Governance creates a subscription in the receiver account when the receiver pulls files from SecureTransport.
- The subscription allows for triggering the post-download action on files successfully pulled by the receiver.
- For SFTP, FTP and HTTP, the subscription folder is taken from the SecureTransport step flow definition between each SecureTransport and receiver, send properties.
- For PeSIT, this subscription folder is generated: `/<PeSIT identifier of the flow between ST and receiver>/<the receiver PeSIT login>`.

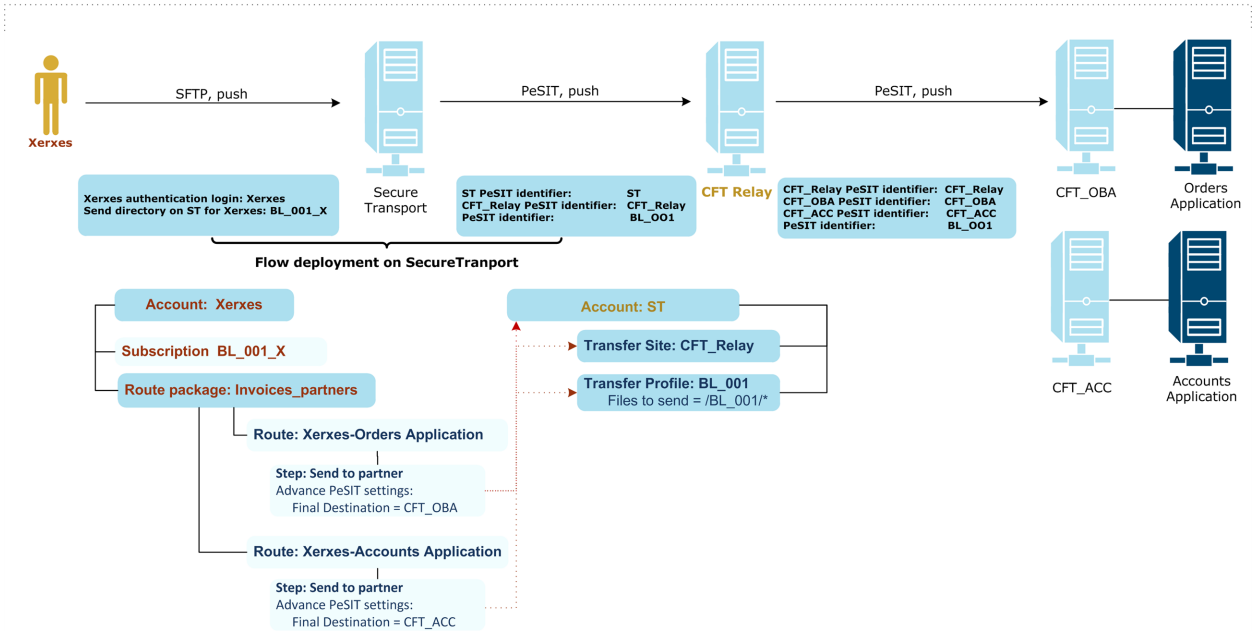
Deployed flow examples



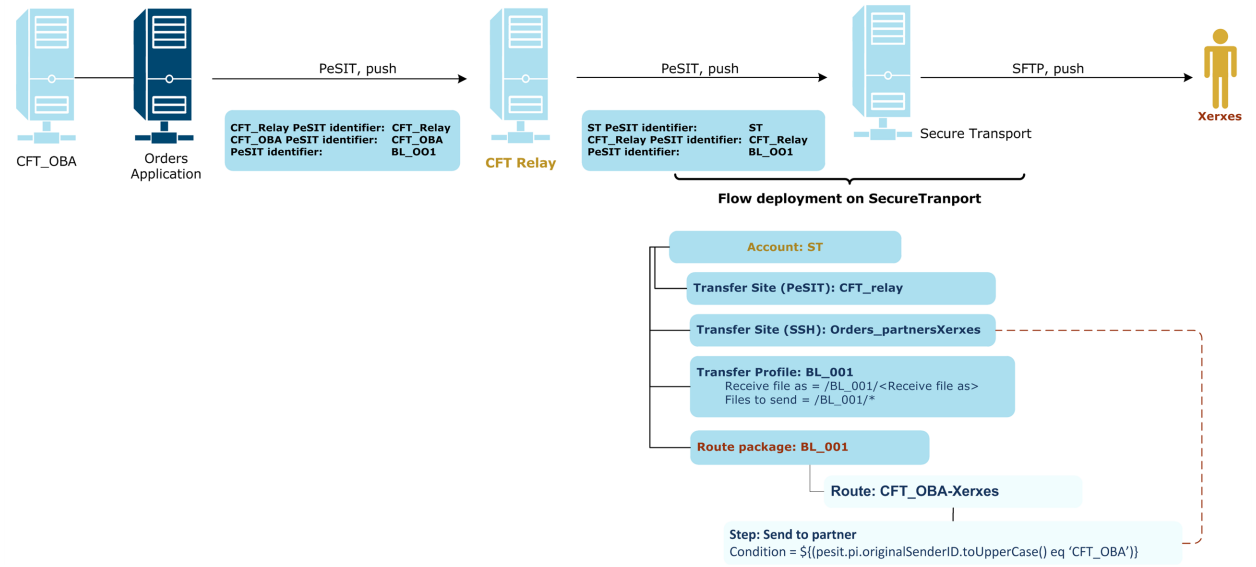

Flow name: Invoices_partners
Xerxes
Fredy
SFTP, push
Secure Transport
Folder Monitoring
Invoices Application
Xerxes's login = Xerxes
Fredy's login = Fredy
Receive directory Xerxes = / XerxesInvoices
Receive directory Fredy = / FredyInvoices
Upload directory = /SentInvoices
Flow deployment on SecureTranport
Account: Xerxes
TransferSite Invoice_partners
protocol= Folder Monitor
Subscription /XerxesInvoices
Route package: Invoice_partners
Route: Xerxes-Invoice Application
Step: Send to partner
Account: Fredy
TransferSite Invoice_partners
protocol= Folder Monitor
Subscription /FredyInvoices
Route package: Invoice_partners
Route: Fredy-Invoice Application
Step: Send to partner


Flow name: Orders_partners
Folder Monitoring
FTP, push
Xerxes
Fredy
Orders Applications
Directory to scan = /Orders
Directory =/OrdersToSend
Flow deployment on SecureTranport
Account: Orders Application
TransferSite Orders_partners
protocol= Folder Monitor
Subscription /OrdersToSend
TransferSite Fredy
protocol= FTP
TransferSite Xerxes
protocol= FTP
Route package: Orders_partners
Route: Orders Application-Xerxes
Step: Send to partner
Route: Orders Application-Xerxes
Step: Send to partner

SecureTransport as source in store-and-forward path
Flow name: Invoices_partners



SecureTransport as target in store-and-forward path
Flow name: Orders_partners



Glossary

active-passive

In an active-passive cluster failover configuration, one or more passive or standby nodes can take over for failed nodes. Normally, only the primary node is used for processing. When it fails, the standby node takes over the resources and the identity of the failed node. The services provided by the failed node are started on the standby node. After the switch, clients can access the services unaware that the services are being provided by a different node.

active mode

In FTP active mode, the client establishes the command channel, from client port X to server port 21, but the server establishes the data channel, from server port 20 to client port Y, where the client supplies Y. Also see passive mode.

advanced routing

In SecureTransport, advanced routing is an intelligent routing engine enabling flexible provisioning of new data flows, creating diverse patterns for moving data among parties.

AES

The Advanced Encryption Standard (AES) is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001.

alert

An event that occurs in the system, such as a product registration failure. An alert may require user intervention.

alert notification

Message sent to recipients defined in an alert rule.

alert rule

A definition containing parameters and notification information. An alert rule is triggered when an alert occurs in the system.

API

An application programming interface (API) is a protocol intended for use as an interface by software components to communicate with each other.

application

The logical definition of a business application that is the real endpoint of a file exchange.

asynchronous

Asynchronous communication is when data are sent intermittently rather than in a steady stream.

base DN

The top level of the LDAP directory tree is the base DN. DN stands for distinguished name.

BSON

BSON, short for Binary JSON, is a binary-encoded serialization of JSON-like documents. Like JSON, BSON supports the embedding of documents and arrays within other documents and arrays.

CA

A certificate authority (CA) is a trusted third party that issues digital certificates for use by other parties.

certificate

A digital certificate contains keys used for encrypting and signing messages, and also for decrypting and verifying signatures. A certificate can contain a public-private key pair or a public key only. See key.

CFTPARM

An object or command for general Transfer CFT environment parameters.

CFTPART

In Transfer CFT, the CFTPART command describes each partner relative to the network/protocol environment by defining the Transfer CFT protocols, network identification, and intermediate partner identification.

CLI

Command line interface (CLI) is a tool for performing actions on products and services.

client communication profiles

Client communication profiles contain details for the sender or receiver to connect via a protocol to the server. The sender acts as client when it pushes files to the receiver. The receiver acts as a client when it pulls files from the sender.

cnxin

Maximum number of sessions for incoming connections in Transfer CFT.

cnxinout

Maximum number of communication sessions in Transfer CFT.

cnxout

Maximum number of sessions for outgoing connections in Transfer CFT.

command line interface

Command line interface (CLI) is a tool for performing actions on products and services.

communication profile

A communication profile contains the technical details for making connections between clients and servers to transfer data. There are two types of communication profiles: client and server. The two types of profiles are based on the roles of senders and receivers in file transfers.

component security descriptor file

Component security descriptor (CSD) files are XML files that define product resources, user privileges and user roles for each product that integrates with Central Governance for identity and access management.

core services

Core services support the Central Governance graphical user interface, identity management and management of all functions related to product configuration and flow definition.

credential

Information to verify a user's identity. For example: passwords, X.509 certificates.

CRL

A certificate revocation list (CRL) is a list of X.509 certificates that have been revoked or are no longer valid and should not be relied upon.

CRONJOB

In Transfer CFT, a scheduled job defined within a script that executes a specified task at set dates and times.

Cronjobs

See CRONJOB

CRONTABs

In Transfer CFT, a CRONTAB is a parameter that represents a file or list of files containing the scheduled jobs.

CSD

Component security descriptor (CSD) files are XML files that define product resources, user privileges and user roles for each product that integrates with Central Governance for identity and access management.

CSR

In public key infrastructure, a certificate signing request (CSR) is a message sent from an applicant to a certificate authority to apply for a digital identity certificate.

DBA

database administrator

Dependency-Check

Dependency-Check is a utility that identifies project dependencies and checks if there are any known, publicly disclosed, vulnerabilities. Dependency-check can currently be used to scan Java applications and their dependent libraries to identify any known vulnerable components.

distribution list

In Transfer CFT, a distribution list manages the list of partners for distribution and collection operations. Use of a distribution list enables, with a single command, sending or receiving a file to all targets or sources in the list.

DMZ

A demilitarized zone (DMZ) or perimeter network is a physical or logical subnetwork that contains and exposes an organization's external-facing services to a larger and untrusted network, usually the Internet. A DMZ adds an additional layer of security to an organization's local area network (LAN). An external network node only has direct access to equipment in the DMZ, rather than any other part of the network.

DNS

Domain Name System (DNS) is an Internet service that translates domain names into IP addresses. As the Internet is based on IP addresses, a DNS service must translate the name into the corresponding IP address.

DTD

A document type definition (DTD) is a set of markup declarations that define a document type for an SGML-family markup language (SGML, XML, HTML).

EBCDIC

Extended Binary Coded Decimal Interchange Code (EBCDIC) is an 8-bit character encoding used mainly on IBM mainframe and IBM midrange computer operating systems.

entity

A password-protected repository of certificates and keys.

expression language

SecureTransport uses an expression language (EL) based on the Sun JSP Expression Language. See the SecureTransport documentation for details. The following SecureTransport features can use EL: transfer site post-transmission actions, subscription post-transmission actions, PGP, account templates.

FGAC

Fine-grained access control (FGAC) is a way to manage users' access to objects or capacity to perform actions. For example, you could enable some users to view specific objects in the user interface, but prohibit other users from viewing the same objects.

file-by-file mode

When a group of files are transferred, the files are transmitted individually

FIPS

Federal Information Processing Standards (FIPS) have been developed by the U.S. government. FIPS describes document processing, cryptographic algorithms and other information technology standards for use within non-military government agencies and by government contractors and vendors who work with the agencies. Your user license for this software supports FIPS-compliant implementations of certain cryptographic algorithms or IAIK implementations of those algorithms. Also see IAIK in the glossary. For more information about FIPS, see http://www.itl.nist.gov/fipspubs/.

flow

A flow specifies the technical details and communications protocols for exchanging business data between business applications or partners.

flows

A flow specifies the technical details and communications protocols for exchanging business data between business applications or partners.

Fortify

Fortify is Hewlett-Packard software for users to assess, assure and protect enterprise software and applications from security vulnerabilities.

FQDN

fully qualified domain name

FTP

File Transfer Protocol (FTP) is a standard network protocol for transfering files from one host to another host over a TCP-based network, such as the Internet. FTP is built on a client-server architecture and uses separate control and data connections between the client and the server.

GFS

Global File System (GFS) is a cluster file system that enables a cluster of computers to simultaneously use a block device that is shared between them. GFS reads and writes to the block device like a local file system, but also allows the computers to coordinate their I/O to maintain file system consistency. With GFS any changes that are made to the file system on one computer will immediately be seen on all other computers in that cluster.

GID

group ID

grouped transfer mode

When a group of files is transferred, the files are transmitted as a group when possible.

groups

Groups enable you to organize and manage related products.

GUI

graphical user interface

Hazelcast

Hazelcast is an in-memory open-source software data grid based on Java. By having multiple nodes form a cluster, data are distributed evenly among the nodes. This enables horizontal scaling for storage space and processing power. Backups are distributed similarly to other nodes, protecting against single-node failure.

HSM

Hardware security module

HSTS

HTTP Strict Transport Security (HSTS) ensures browsers connect securely to the user interface. If a user includes http:// in the URL to connect, HSTS converts it to https://.

HTTP

Hypertext Transfer Protocol (HTTP) is an application protocol for distributed, collaborative, hypermedia information systems. HTTP is the foundation of data communication for the World Wide Web. HTTP defines how messages are formatted and transmitted, and what actions web servers and browsers should take in response to commands.

HTTPS

HTTPS is a protocol for secure communication over a computer network widely used on the Internet. HTTPS consists of communication over Hypertext Transfer Protocol (HTTP) within a

connection encrypted by Transport Layer Security or its predecessor, Secure Sockets Layer. HTTPS is also called HTTP over TLS, HTTP over and HTTP Secure.

IAIK

The Institute for Applied Information Processing and Communications (IAIK) researches information and computer security. This includes design and implementation of cryptographic algorithms and protocols in hardware and software, network security and trusted computing. Your user license for this software supports FIPS-compliant implementations of certain cryptographic algorithms or IAIK implementations of those algorithms. Also see FIPS in the glossary. For more information about IAIK see http://www.iaik.tugraz.at/.

IAM

Identity and access management (IAM) is a role-based solution for securing enterprise resources and managing user access to protected network components through a continuous and interactive authorization process.

IBM i

IBM i is an EBCDIC-based operating system that runs on IBM Power Systems and on IBM PureSystems. The name, introduced in 2008, is the current evolution of the operating system. IBM i, formerly named i5/OS, originally was named OS/400 when introduced with the AS/400 computer system in 1988.

identity and access management

Identity and access management (IAM) is a role-based solution for securing enterprise resources and managing user access to protected network components through a continuous and interactive authorization process.

identity store

A central repository for managing user identity information, such as roles, privileges and groups. There are two types: internal and external.

IDF

An IDF is a model file identifier, or file identifier, in Transfer CFT.

IPART

In Transfer CFT, the IPART is the local identifier of an intermediate partner. The identifier must correspond to the ID parameter of a CFTPART object. This parameter is involved in the file store and forward or backup mechanism.

IPv4

Internet Protocol version 4 (IPv4) is the fourth version of the Internet Protocol (IP). It is one of the core protocols of standards-based internetworking methods in the Internet, and was the

first version deployed for production in the ARPANET in 1983. It still routes most Internet traffic today, despite the ongoing deployment of a successor protocol, IPv6.

IPv6

Internet Protocol version 6 (IPv6) is a set of specifications from the Internet Engineering Task Force (IETF) that's essentially an upgrade of IP version 4 (IPv4).

ISO

The International Organization for Standardization sets standards in many businesses and technologies, including computing and communications.

JDBC

Java database connectivity (JDBC) is an API for the Java programming language that defines how a client can access a database. It provides methods for querying and updating data in a database. JDBC is oriented towards relational databases.

JSON

JavaScript Object Notation (JSON) is a text-based open standard designed for human-readable data interchange. It is derived from the JavaScript scripting language for representing simple data structures and associative arrays, called objects. Despite its relationship to JavaScript, it is language-independent, with parsers available for many languages.

key

Keys are contained in digital certificates. There are two kinds of keys: Private and public. A private key is your secret key for decrypting messages or signing messages. A public key is also your key, but it can be used by a partner to encrypt messages that only you can decipher with your private key.

LDAP

Lightweight Directory Access Protocol (LDAP) is an open, vendor-neutral, industry-standard application protocol for accessing and maintaining distributed directory information services over an Internet Protocol (IP) network.

log4j

Log4j, an open source project, allows the developer to control which log statements are output with arbitrary granularity. It is fully configurable at runtime using external configuration files.

MFT

Managed file transfer (MFT) refers to software or a service that manages the secure transfer of data from one computer to another through a network or over the Internet.

Nessus

Nessus is an open-source network vulnerability scanner that uses the Common Vulnerabilities and Exposures architecture for easy cross-linking between compliant security tools. Nessus employs the Nessus Attack Scripting Language (NASL), a simple language that describes individual threats and potential attacks.

NFNAME

In Transfer CFT, the name of the physical file at the receiver partner site.

NFS

Network File System (NFS) is a distributed file system protocol originally developed by Sun Microsystems in 1984. It allows a user on a client computer to access files over a network in a manner similar to how local storage is accessed.

notification

See alert notification.

NTOSpider

NTOSpider, a dynamic application security scanner for testing web and mobile applications, identifies application vulnerabilities and site exposure risk.

OMAXTIME

In Transfer CFT, the IMINTIME, IMAXTIME, OMINTIME and OMAXTIME parameters of the CFTPART command define the time slot for communicating with a partner for incoming and outgoing calls.

OMINTIME

In Transfer CFT, the IMINTIME, IMAXTIME, OMINTIME and OMAXTIME parameters of the CFTPART command define the time slot for communicating with a partner for incoming and outgoing calls.

OS/400

See IBM i

PARM

A user parameter or parameter file in Transfer CFT.

partner

In Transfer CFT a partner is a logical entity such as a bank, a government agency or trading partner, that can be the sender or receiver of data. A partner corresponds to a remote file-transfer controller.

partners

Partners represent entities such as companies that send or receive business data in file transfers governed by Central Governance flows.

passive mode

In FTP passive mode, the client establishes the command channel and the data channel. The server tells the client which port to use for the data channel. Also see active mode.

password policy

Rules and conditions for valid passwords, such as character length, case requirements and validity periods.

pentest

A penetration test, or pentest, is an attack on a computer system with the intention of finding security weaknesses, potentially gaining access to it, its functionality and data.

PeSIT

PeSIT is an open file transfer protocol often associated with Axway. PeSIT stands for Protocol d'Echanges pour un Systeme Interbancaire de Telecompensation. It was designed as a specialized replacement for FTP to support European interbank transactions in the mid-1980s.

PGP

Pretty Good Privacy (PGP) is a data encryption and decryption program for cryptographic privacy and authentication in data communication. PGP often is used for signing, encrypting, and decrypting texts, emails, files, directories and whole disk partitions, and to increase the security of email communications.

PI37

The file label predefined in the transfer profile in SecureTransport.

PI99

The predefined user message in the PeSIT transfer site in SecureTransport.

PKI

Public key infrastructure (PKI) enables users of a basically unsecure public network such as the Internet to securely and privately exchange data through use of a public and a private cryptographic key pair that is obtained and shared through a trusted authority.

privilege

A user right to perform an action on a resource.

processing script

In Transfer CFT, a set of rules in a file for actions to execute on files transferred between Transfer CFTs. A processing script can be executed before a transfer (pre-processing script) or after a transfer (post-processing script).

product

Axway business application software. For example, Transfer CFT.

products

Axway business application software. For example, Transfer CFT.

pTCP

The parallel Transmission Control Protocol (pTCP) is an end-to-end transport layer protocol that supports striped connections.

RAC

Oracle Real Application Clusters (RAC) provides software for clustering and high availability in Oracle database environments.

RAID

Redundant array of independent disks (RAID) is a storage technology that combines multiple disk drive components into a logical unit for the purposes of data redundancy and performance improvement. Data are distributed across the drives in one of several ways, referred to as RAID levels, depending on the specific level of redundancy and performance required.

RBAC

In computer systems security, role-based access control (RBAC) is an approach to restricting system access to authorized users. RBAC is sometimes referred to as role-based security.

relay

A relay is an intermediate product between the source and target, or true sender and true receiver, in a flow. A relay can be an Axway product or an unmanaged product.

resource

A class of object in a product whose use can be authorized only through privileges associated with user roles.

REST

Representational State Transfer (REST) is a software architecture style for building scalable web services. REST consists of a coordinated set of constraints applied to the design of components in a distributed hypermedia system that can lead to a more performant and maintainable architecture.

restart Transfer CFTs

When Central Governance restarts Transfer CFTs to update their configurations, only the Transfer CFT servers are restarted. Central Governance does not start or stop Transfer CFT Copilots. Central Governance communicates with Transfer CFT servers via their Copilots. Central Governance can perform actions on Transfer CFT servers only when their Copilots are running.

RESTful

RESTful systems typically, but not always, communicate over the Hypertext Transfer Protocol with the same HTTP verbs (GET, POST, PUT, DELETE) web browsers use to retrieve web pages and to send data to remote servers. Representational State Transfer (REST) interfaces usually involve collections of resources with identifiers. For example, /people/paul, which can be operated upon using standard verbs, such as DELETE /people/paul.

RMI

Remote method invocation (RMI) is a distributed object technology for the Java programming language. It is available as part of the core Java application programming interface (API) where the object interfaces are defined as Java interfaces and use object serialization.

role

A collection of privileges. Roles are assigned to users and govern the products they can access and the actions they can perform.

salt

In cryptography, a salt is random data that are used as an additional input to a one-way function that hashes a password or passphrase. The primary function of salts is to defend against dictionary attacks and pre-computed rainbow table attacks. A new salt is randomly generated for each password. In a typical setting, the salt and the password are concatenated and processed with a cryptographic hash function, and the resulting output (but not the original password) is stored with the salt in a database. Hashing allows for later authentication while defending against compromise of the plaintext password in the event that the database is somehow compromised.

SDL

The secure development lifecycle (SDL) is process for enhancing product security during development of the product.

self-signed certificates

In cryptography and computer security, a self-signed certificate is an identity certificate that is signed by the same entity whose identity it certifies. This term has nothing to do with the identity of the person or organization that actually performed the signing procedure. In technical terms a self-signed certificate is one signed with its own private key.

server communication profiles

Server communication profiles contain details for a client to transfer data via a protocol to the sender or receiver that acts as a server. The sender acts as a server when it publishes files for the receiver. The receiver acts as a server when it receives files pushed by the sender.

service class

A service class groups product services that facilitate data flows.

SFTP

SSH File Transfer Protocol (SFTP) is a network protocol for file transfer and manipulation functionality over any reliable data stream.

SHA

The Secure Hash Algorithm (SHA) is a family of cryptographic hash functions published by the National Institute of Standards and Technology (NIST) as a U.S. Federal Information Processing Standard (FIPS).

SID

In Oracle, a SID dentifies the database instance (database name + instance number). For example, if the database name is somedb and the instance number is 3, the SID is somedb3.

single sign-on

Single sign-on (SSO) enables a user to log on once and gain access to all components managed by the SSO system without being prompted to log on again for each component.

SMTP

Simple Mail Transfer Protocol (SMTP) is an Internet standard for electronic mail (email) transmission.

SOAP

Simple Object Access Protocol (SOAP) is a protocol specification for exchanging structured information in the implementation of Web services in computer networks. It relies on Extensible Markup Language (XML) for its message format. SOAP usually relies on other application layer protocols, most notably Hypertext Transfer Protocol (HTTP) or Simple Mail Transfer Protocol (SMTP), for message negotiation and transmission.

source

The source is the owner of the data being transferred.

source initiator mode

Transfer mode in which the Transfer CFT where the files are located initiates the transfer

SSH

Secure Shell (SSH), sometimes known as Secure Socket Shell, is a UNIX-based command interface and protocol for securely getting access to a remote computer.

SSL

Secure Sockets Layer (SSL), which is the predecessor of Transport Layer Security (TLS), is an encryption protocol that ensures communication security over the Internet. See TLS for more information.

SSO

Single sign-on (SSO) enables a user to log on once and gain access to all products managed by the SSO system without being prompted to log on again for each product.

STARTTLS

STARTTLS is an extension to plain text communication protocols, which offers a way to upgrade a plain text connection to an encrypted TLS or SSL connection instead of using a separate port for encrypted communication.

store-and-forward

Store-and-forward occurs when files are routed through one or more intermediary sites called store-and-forward sites. The feature only is available from a requester/sender (write-mode transfer).

synchronous

Synchronous communication is when data are sent in a continuous stream at a constant rate.

tags

Keywords for classifying objects like applications, products, flows, partners.

target

The target is the receiver of the data exchange.

target initiator mode

Transfer mode in which the Transfer CFT that will receive the files sends a request to another Transfer CFT to send the files

TCP

The Transmission Control Protocol (TCP), one of the core protocols of the Internet protocol suite (IP), is often called TCP/IP. TCP provides reliable, ordered and error-checked delivery of a stream of octets between programs running on computers connected to a local area network, intranet or the public Internet. It resides at the transport layer.

TLS

Transport Layer Security (TLS) is an encryption protocol that ensures communication security over the Internet. TLS encrypts the network connection above the transport layer. TLS uses asymmetric cryptography for key exchange, symmetric encryption for privacy and message authentication codes for message integrity. Secure Sockets Layer (SSL) is the predecessor of TLS.

transfer

in Transfer CFT, the data transport and exchange of the actions to be taken on the data (read, write, create, delete) from one computer (partner) to another via a network. One of the partners is the sender and the other the receiver.

UCONF

Stands for unified configuration in Transfer CFT.

UDT

The UDP-based Data Transfer Protocol (UDT) is a high performance data transfer protocol designed for transferring large volumetric data sets over high-speed wide-area networks.

UFM

see Unified Flow Management

UI

user interface (same as graphical user interface or GUI)

UID

A user ID (UID) is a unique positive integer assigned by a Unix-like operating system to each user. Each user is identified to the system by its UID, and user names are generally used only as an interface for humans.

Unified Flow Management

Unified Flow Management (UFM) is a set of products in the Axway 5 Suite that enable you to manage the flow of data within and outside your enterprise.

unmanaged product

Unmanaged products are systems that are not registered in Central Governance, but that are integrated in flows for transferring files. Unmanaged products can be Axway products that cannot register in Central Governance or third-party products.

unmanaged products

Unmanaged products are systems that are not registered in Central Governance, but that are integrated in flows for transferring files. Unmanaged products can be Axway products that

cannot register in Central Governance or third-party products.

update package

A file containing a service pack, patch or version upgrade for a product. Typically, users download update packages from the Axway Sphere support website.

URI

A uniform resource identifier (URI) is a string of characters for identifying the name of a resource. This enables interaction with representations of the resource over a network, typically the World Wide Web, using specific protocols.

user exit

A subroutine invoked by a software package for a predefined event in the execution of the package. Clients of the package can substitute their own subroutines in place of the default ones provided by the package vendor to provide customized functionality. A typical use is replacing the user exits provided by a sort-merge package, where the user program provides its own subroutines for comparing records. The procedures provided by the user take the place of the default routines provided by the package vendor. Procedures provided as user exits are typically compiled into a static library and linked directly with the package to produce an executable program. Another approach employs dynamic libraries to accomplish the same thing.

UUID

A universally unique identifier (UUID) is an identifier standard used in software construction. A UUID is simply a 128-bit value. The meaning of each bit is defined by any of several variants.

VPN

A virtual private network (VPN) is constructed by using public wires — usually the Internet — to connect to a private network, such as a company's internal network.

Web service

A software system designed to support interoperable machine-to-machine interaction over a network.

workflow

The sequence of tasks through which a process advances from initiation to completion.

write-mode transfer

In the PeSIT protocol, a write-mode transfer occurs when a file is sent from the requester to the server.

z/OS

z/OS is a 64-bit operating system for mainframe computers, produced by IBM. It derives from and is the successor to OS/390.

zip bomb

A zip bomb, also known as a zip of death or decompression bomb, is a malicious archive file designed to crash or render useless the program or system reading it. It is often employed to disable antivirus software and create an opening for more traditional viruses.