



TIBCO® Managed File Transfer Command Center

Quick Start Guide

Version 8.7.1 | June 2026

Contents

Contents	2
Getting Internet Server Transfers Working	10
Creating a Transfer User	10
Creating a Server Definition	11
Retrieving Keys From Target Servers	12
Creating a Transfer Definition	12
Performing a File Transfer	13
Viewing File Transfer History	14
Adding Server Definitions	15
Server Types Supported by MFT	15
Viewing Audit Records	40
Configuring and Starting MFT Transfer Services	41
Configuring and Starting the SSH Service	42
Configuring and Starting the FTP Service	44
Configuring and Starting the Platform Server Service	48
Configuring PGP	52
How PGP Works	52
When to Use PGP?	53
Creating a PGP System Key	54
Configuring the Public Key: Client Sends or Receives PGP-Encrypted Data	56
Configuring the Public Key: Client Sends or Receives PGP-Encrypted Data With Target Server	57
Client Sends PGP-Encrypted Data to MFT Internet Server	58
Client Receives PGP-Encrypted Data from MFT Internet Server	59

MFT Internet Server Sends PGP-Encrypted Data to a Target Server	60
MFT Internet Server Receives PGP-encrypted Data from a Target Server	61
Configuring MFT for Key or Certificate Authentication	63
Incoming HTTPS Client Requests Certificate Authentication	64
Configuring the MFT Server for HTTPS Certificate Authentication	64
Configuring the MFT Server to Allow or Require HTTPS Certificate Authentication	66
Incoming FTPS Client Requests Certificate Authentication	68
Configuring the MFT Server to Allow or Require FTPS Certificate Authentication	69
Incoming SSH or SFTP Client Requests Key or Certificate Authentication	72
Configuring the MFT Server to Allow or Require SFTP Key Authentication	73
Incoming Platform Server Client Request Certificate	77
Saving the MFT Platform Server Public Certificate in the Trusted Authority File	77
Creating a Platform Server Keypair	78
Configuring the MFT Server to Allow or Require Platform Server Authentication	79
Authenticating Outgoing Certificate to Target HTTPS Server	83
Authenticating Outgoing Certificate to Target FTPS Server	84
Authenticating Outgoing Key or Certificate to Target SFTP Server	85
Authenticating Outgoing Certificate to Target Platform Server	87
Creating and Using Alerts	90
Login Alerts	92
Transfer Non-Event Alerts	92
Adding a Transfer Non-Event Alert	92
Create a Scheduler Job Type: Non-Event Transfer Alert	93
Transfer Event Alerts	95
Adding a Transfer Event Alert	95
Collecting Platform Server Transfer Audit Records	96
Configuring Platform Server	96
Configuring Command Center	97
Configuring Server Definition to Turn Collection On	97

Configuring Command Center Collection Service	98
Start or Restart Command Center Collection Service	99
Viewing Platform Server Audit Records	100
Polling Platform Servers for Audit Record Inquiry	101
Diagnosing Problems	101
Managing Platform Servers	103
Configuring Command Center to Support Platform Server Configuration	104
Configuring Platform Server Definitions	105
Performing Platform Server Node Configuration	106
Managing and Updating Platform Server Nodes	107
Executing Platform Server Transfers	109
Configuring Command Center and Platform Server	110
Configuring Platform Server Definitions	111
Adding and Executing Platform Server Transfers	112
Credentials and Security Properties	113
Updating and Executing Platform Transfer	115
Executing Platform Transfers	116
FileShare, Mailbox, and Four Eyes	120
Creating PGP System Key	121
Creating Repository Server Definition	122
Associating PGP Public Key with Server Definition	126
Updating the Global Configuration	127
Updating FileShare Mailbox Four Eyes Configuration	129
Updating User Definition	132
Configuring Four Eyes Policy	134
Test the Mailbox Browser	134
Testing the FileShare Browser	136
Testing the Four Eyes Browser	138
Testing Automated Four Eyes Requests	139

Configure and Start the AS2 Transfers	141
Configuring MFT AS2 Transfer Server	142
Creating AS2 System Key to Decrypt Data	143
Creating User Definition for Incoming AS2 Requests	144
Server Definition for Incoming and Outgoing AS2 Requests	145
Creating Server Definition for Incoming and Outgoing AS2 Requests	146
Creating Transfer Definitions for Incoming and Outgoing AS2 Requests	148
Information for AS2 Transfers	151
Executing AS2 Transfers	152
 Using the MFT Scheduler	 154
Configuring and Starting Command Center Scheduler	155
Calendars	156
Creating Exclusion Calendar	156
Creating Inclusion Calendar	157
Scheduler Jobs	158
Creating Scheduler Jobs	159
Schedule Triggers	163
Debugging Scheduler Jobs	164
Manage Scheduler	164
 Delegated Administration	 166
Creating Departments	167
Adding Users to Departments	168
Managing Other Resources	170
Viewing Resources	171
 Configuring and Managing DNI Daemons	 172
Installing DNI	173
Starting the DNI Daemon	174
Configuring Command Center to Support DNI Management	175
Managing DNI Daemons	176

Generating Templates	178
Using DNI to Send and Receive files to Internet Server Partners	181
File Transfer Initiated by pDNI to Target SSH Server	182
Configure the Internet Server Platform Server Service and Start the Platform Server Service	183
Configuring a Server Definition to Define the Connectivity and Credentials for Target SSH Server	183
Creating a User Definition for Incoming Client Request	184
Configuring Platform Server Definitions to Connect to Internet Server	185
Create Transfer Definition for Defined User and Server Definitions	186
Creating and Configuring the pDNI Template	188
Starting pDNI Template	190
Receiving Files from Target SSH Servers	192
Configure the Internet Server Platform Server Service and Start the Platform Server Service	193
Configuring a Server Definition to Define the Connectivity and Credentials for Target SSH Server.	194
Creating a User Definition for Incoming Client Request	195
Configuring Platform Server to Connect to Internet Server	195
Create Transfer Definition for Defined User and Server Definitions	196
Creating and Configuring the pDNI Template	198
Starting pDNI Template	200
Configuring MFT for LDAP Authentication	202
Defining LDAP Authenticators	203
Testing the LDAP Authenticator	208
Syncing LDAP to Database	209
Debug LDAP Problems	210
Sync Users at Login	210
LDAP and SSO (Single Sign-On)	211
Configuring Single SignOn support using OIDC (OpenID Connect)	212

Constructing MFT OIDC Redirect URL	213
Configuring the OIDC Authentication Server	214
Adding the OIDC Provider	214
Configuring the OIDC Information for Internet Server and Command Center Instances	215
Testing the OIDC Log In	217
Configuring OFTP2 Transfers	218
Creating OFTP2 System Keys	221
Configuring MFT OFTP2 Transfer Server	223
Create a User Definition for incoming OFTP2 Requests	224
Create a Server Definition for Incoming and Outgoing OFTP2 Requests	225
Create Transfer definitions for Incoming and Outgoing OFTP2 Requests	231
Starting OFTP2 Service	234
Send Information about the MFT OFTP2 Environment to OFTP2 Transfer Partner	234
Executing OFTP2 Transfers	235
Configuring MFT for SAML SSO	237
Creating SAML Private Keys	238
Importing SAML Identity Provider Metadata	239
Configuring SAML Service Provider Metadata	240
Generating SAML Service Provider Metadata	241
Sending SAML Service Provider Metadata to the Identity Provider	242
Restarting the MFT Server	243
Updating MFT Shortcuts	243
Configuring MFT for Multi Factor Authentication	245
Common MFA Configuration	245
Manage MFT MFA Instances	246
Diagnosing and Debugging Problems	249
Dashboard	254
Active Transfers	256

MFT Tracing	258
Login Tracing	258
SSH Tracing	259
JMS Tracing	260
MFT Tracing	260
Other Types of Tracing	262
Common Debugging Scenarios	262
Key or Certificate Log ins are Failing	263
Using REST calls to configure Internet Server and Command Center	274
Supported Admin and Command Center REST Calls	274
Documentation on REST Calls	277
Format of the REST Call	278
Configuring and Using JMS	281
JMS Support Overview	281
Command Center	281
Supported JMS Software	282
Installing TIBCO EMS JAR Files	283
Configuring JMS On Command Center	283
Testing JMS Connections	285
Starting the JMS Service	286
Using JMS on Command Center and Internet Server	287
Configuring Alerts to Send JMS Messages	287
Configuring JMS Servers and Transfers	288
Initiating Transfers to a JMS Queue	291
Configuring Hawk Microagent	292
Configuring Password Vault	296
Processing ZIP Files	300

TIBCO Documentation and Support Services	301
Legal and Third-Party Notices	303

Getting Internet Server Transfers Working

This section describes the process of installing TIBCO MFT for the first time, and running the transfers.

Prerequisites

Before a transfer can run, create the following resources:

- *User*: Defines the user that can run the transfer.
- *Server*: Defines the destination server where files can be uploaded to or downloaded from.
- *Transfer*: Defines the user that can run the transfer, the server where the files are located, and the directories where the files are read from or written to.

See the following topics to get started with TIBCO MFT:

1. [Creating a Transfer User](#)
2. [Creating a Server Definition](#)
3. [Creating a Transfer Definition](#)

Creating a Transfer User

To create a transfer user, perform the following procedure:

Procedure

1. Go to **Partners > Users > Add User**.
2. Enter the required user information (such as **User Id**, **Full Name**, **Password**, **Confirm Password**).
3. Assign **TransferRight** to the user in the **Rights and Groups** section.



Note: **TransferRight** is required to perform a file transfer.

4. On the **Optional User Properties** tab, clear the **Change Password at Next Login** checkbox.

i **Note:** The **Change Password at Next Login** checkbox is selected by default. Clearing this checkbox ensures that you do not need to change the password for this user the next time the user logs in.

5. Click **Add**.

Creating a Server Definition

During the installation, a server called "*LOCAL" is created. This server definition allows you to read and write to disks on the server where the MFT Internet Server is running. You can also use this server definition (or another pre-existing server) and skip to the [Creating a Transfer definition](#) step.

✓ **Tip:** If you are setting up transfers for the first time, using a local server is recommended.

To create a server definition, complete the following steps:

Procedure

1. Go to **Partners > Servers > Add Server**.
2. Enter the required server information (Server Name, Server Type, IP Address, IP Port, and Server Platform).
3. Enter the server credentials (User ID and password) on the **Server Credentials** tab. Credentials are not needed for Server Type=LOCAL.
4. Click **Add**.

i **Note:** If you are connecting to an HTTPS, FTPS, SSH (SFTP), or Platform Server SSL, or Platform Server tunnel, you must follow this procedure to retrieve the key from the target server and save it in the MFT database.

Retrieving Keys From Target Servers

You need to retrieve keys from target servers when you add an HTTPS, FTPS, SSH, or Platform Server SSL or tunnel server type. Transfers with these servers fail until you have retrieved the key or manually added the certificate or key and associated it with the server. To retrieve the certificate or SSH key from the target server, perform the following steps:

1. Go to **Partners > Servers > Manage Servers**.
2. Find the server that you created and click the server name link.
3. Click the **Retrieve xxx public key through this server** link to retrieve the partner's Public key. If you cannot retrieve the key from the default server, you might need to select an Internet Server to retrieve the key. This connects to the target server.
4. The key or certificate is extracted from the target server and saved in the MFT database.

i Note: After adding a server that requires you to retrieve a Public Key, an information message is displayed. This message includes a link to the **Update Server** page, where you can retrieve the partner public key.

Creating a Transfer Definition

The transfer definition connects the user to the server definition and allows you to define where the files to be transferred are located.

To create a transfer, complete the following steps:

Procedure

1. Go to **Transfers > Internet Server Transfers > Add Transfer**.
2. Enter the following user information:

Field	Description
Client File Name	Define any value. This field is required but not used.
Server File Name	Define the location where files are read from or written to.
Directory Transfer	Set to Yes.
Description	Create a unique description.
Authorized userid	Set this to the user ID created in the prior step.
Server Name	Set this to the server created in the prior step.
Transfer Direction	Set to Upload to Server , Download to Client , or Both .
Virtual Alias	Set this to the name that is displayed as the directory name when performing file transfers. Do not embed spaces or slashes in this field.

3. On the **Additional Transfer Properties** tab, update the write mode as per your requirement.
4. On the **Client Permissions** tab update the permissions fields to permit the client to perform the delete, rename, create directory operations.
5. Click **Add**.

Performing a File Transfer

To perform the file transfer, complete the following steps:

Procedure

1. Log in to the Internet Server Browser Client using the following URL

```
https://your.mftis.server:7443/browser
```

i Note: You need to set the DNS name and possibly, the port if the Internet Server is not using port 7443.

2. When you enter the credentials for the user defined in the previous step, you are directed to a screen that displays the **Virtual Alias** defined in the transfer definition.
3. Select the name that matches the **Virtual Alias** defined in the transfer definition. You must see a list of files located in the directory.
4. Select a directory to go to the next level.
5. Select the file that you want to upload by using one of the following methods:
 - a. Drag a file from Windows File Explorer to the current page.
 - b. Click **Upload** to select a file to upload.

Viewing File Transfer History

After you have performed a file transfer, you can view the history of file transfers performed by the logged in user.

Procedure

Click **History** in the upper-left corner of the browser transfer client.

Result

A list of the last 100 file transfers for the logged in user is displayed.

Adding Server Definitions

You can use server definitions to define the configuration parameters necessary to send files to and receive files from destination servers. The [Getting Internet Server transfers](#) topic discusses the process of running transfers on a *LOCAL server definition. We also briefly mentioned how to get transfers working to other server types. This section discusses in more detail, the target Servers that MFT supports and how to get transfers working to these target servers.

For more information about server definitions, see the following sections:

- [Server Types Supported by MFT](#)
- [Creating a New Transfer Definition](#)

Server Types Supported by MFT

The MFT server supports the following server types:

Server Type	Function
LOCAL	Send files to or receive files from any disk that the Command Center has access to.
Platform Server	Send files to or receive files from Platform Servers.
FTP	Send files to or receive files from FTP servers.
SSH	Send files to or receive files from SSH/SFTP servers.
JMS	Send files to or receive files from JMS servers.
AS2	Send files to AS2 servers.
HTTP	Send files to or receive files from HTTP servers.

Server Type	Function
Microsoft Azure	Send files to or receive files from Azure File Storage, Blob storage, or ADLS Gen2 storage.
Amazon S3	Send files to or receive files from Amazon S3 storage.
Google Cloud	Send files to or receive files from Google Cloud Storage or BigQuery.
HDFS	Send files to or receive files from HDFS/Hadoop servers.
FileShare	Send files to or receive files from MFT FileShare folders.
Email	Send a file to a user as an email attachment.
Mailbox	Send a file to a user as an MFT mailbox attachment.
Custom Server	Write customized code to support protocols not supported by MFT.
SharePoint	Send file to or receive file from Microsoft SharePoint servers.
OFTP2	Send files to a target OFTP2 Server.
Four Eyes	Send files to a user as a Four Eyes attachment.

One of the advantages of the way that MFT is designed is that MFT virtualizes access to target servers. So, clients do not know where the target files are located. This also makes it easier to give a user access to upload and download files to multiple target servers.



Note: Not all functionality is supported on all of the server types.

The following subtopics list each server type, and its function.

LOCAL

You can use the local storage to save files on any disk accessible to the MFT instance. This can be an NFS share on UNIX, a UNC drive, or mapped drive on Windows. LOCAL servers are sometimes used for testing or debugging clients.

Point	Keep in mind
Capabilities	• Uploads and downloads are supported. • File or directory lists are supported. • Checkpoint Restart is supported. • Files can be renamed or deleted; directories can be created.
Authentication	No credentials required.
Considerations	• When using multiple Internet Server instances, each instance should have access to the defined Server File Name directories. Therefore, you should use an NFS share or a UNC drive when multiple Internet Servers need to process transfers for LOCAL storage. • It is a good practice not to use LOCAL storage on servers running in the DMZ unless the files are stored in an encrypted mode. The exception to this is when running on a cloud server and the data is stored on secured cloud storage. • If you want to disable LOCAL storage, you can configure the AllowLocalServerDefinition web.xml parameter. By setting this parameter to false, you cannot define LOCAL server definitions. If the transfer users attempt to use an existing LOCAL server, the transfer fails with an error. • LOCAL storage can be used as a repository for FileShare and Mailbox services. Platform Server can also be used as a repository.

Platform Server

Platform Servers enable you to upload files to and download files from target Platform Servers.

Point	Keep in mind
Capabilities	• Uploads and downloads are supported. • File or directory lists are supported.

Point	Keep in mind
	• Checkpoint Restart is supported. • Files can be renamed or deleted; directories can be created.
Authentication	• User ID/password credentials are required when connected to target Platform Servers. • Certificate authentication when TLS or Tunnel mode is used for Platform Server for UNIX or Platform Server for z/OS.
Considerations	• When using multiple Internet Server instances, each instance can connect to the same Platform Server. Hence, all files are accessible to all Internet Server instances. • When configuring Platform Server for "Implicit SSL" or "TLS Tunnel", you must associate the target Platform Server SSL key with the server definition. To associate the target Platform Server SSL key with the server definition, perform the following steps: 1. Go to Partners > Servers > Manage Servers. 2. Select the Platform Server. 3. Click the Retrieve Platform Server public key. The public key is retrieved from the Platform Server associated with the server definition, and is stored in the database. • When configuring Platform Server for "Implicit SSL" or "TLS Tunnel", the certificate associated with the MFT private key must be added to the Platform Server "Trusted Authority File" on UNIX/Windows, or the RACF keyring on z/OS. • Platform Server storage can be used as a repository for FileShare and Mailbox services. LOCAL can also be used as a repository.

FTP

FTP servers enable you to upload files to and download files from target FTP, or FTPS servers.

Point	Keep in mind						
Capabilities	• Uploads and downloads are supported. • File or directory lists are supported. • Checkpoint Restart is supported. • Files can be renamed or deleted; directories can be created.						
Authentication	• User ID/password credentials are required when you are connected to target FTP servers. • Certificate authentication when using FTPS explicit or implicit SSL mode.						
Considerations	• FTP/FTPS servers do not work well in the cloud due to the requirement for data and control connections. We recommend using SFTP when Internet Server is running in the cloud. • When using multiple Internet Server instances, each instance can connect to the same FTP Server. Hence, all files are accessible to all Internet Server instances. • FTP Transfer requires the following TCP connections: <table data-bbox="548 1136 1414 1455"> <tr> <th>Connection</th><th>Description</th></tr> <tr> <td>Control</td><td>Used to authenticate, change directories, and initiate transfers.</td></tr> <tr> <td>Data</td><td>Used to return list command responses and to perform uploads and downloads.</td></tr> </table> Note: Load balancers must be configured to send data connections to the same FTP server as the control connections. • Local TCP ports numbers are typically required when going through firewalls. To define local TCP ports, complete the following steps:	Connection	Description	Control	Used to authenticate, change directories, and initiate transfers.	Data	Used to return list command responses and to perform uploads and downloads.
Connection	Description						
Control	Used to authenticate, change directories, and initiate transfers.						
Data	Used to return list command responses and to perform uploads and downloads.						

Point	Keep in mind								
	1. Go to Configuration > System Configuration > Global FTP Settings. 2. Configure the following parameters: <table border="1"> <thead> <tr> <th>Parameter</th><th>Instruction</th></tr> </thead> <tbody> <tr> <td>Limit Local Ports</td><td>Set to Yes.</td></tr> <tr> <td>Starting Port</td><td>Set to the desired port or use the default.</td></tr> <tr> <td>Number of Ports to Use</td><td>Set the number of ports that can be used.</td></tr> </tbody> </table> • PORT and PASV mode are both supported. We suggest using PASV mode when communicating with target FTP/FTPS servers. • When configuring FTPS for "Implicit SSL" or "Explicit SSL", you must associate the target FTPS SSL key with the server definition. To associate the target FTPS SSL key with the server definition, perform the following steps: 1. Go to Partners > Servers > Manage Servers. 2. Select the desired FTPS server. 3. Click Retrieve FTP public key. The public key is retrieved from the FTPS server, associated with the server definition and stored in the database.	Parameter	Instruction	Limit Local Ports	Set to Yes.	Starting Port	Set to the desired port or use the default.	Number of Ports to Use	Set the number of ports that can be used.
Parameter	Instruction								
Limit Local Ports	Set to Yes.								
Starting Port	Set to the desired port or use the default.								
Number of Ports to Use	Set the number of ports that can be used.								

SSH

SSH servers enable you to upload files to and download files from target SSH servers. When used by MFT, SSH means SFTP. SFTP means transfers over SSH. This is different from FTPS, which means FTP transfers over SSL.

Point	Keep in mind
Capabilities	• Uploads and downloads are supported. • File or directory lists are supported. • Checkpoint Restart is supported. • Files can be renamed or deleted; directories can be created.
Authentication	• User ID/password credentials are required when connected to target SSH servers. • Certificate authentication is required.
Considerations	• When using multiple Internet Server instances, each instance can connect to the same SSH server. Hence, all files are accessible to all Internet Server instances. • When configuring an SSH server, you must associate the target SSH server key with the server definition. To associate the target SSH server key with the server definition, perform the following steps: 1. Go to Partners > Servers > Manage Servers. 2. Select the desired SSH server. 3. Click Retrieve SSH public key. The public key is retrieved from the SSH server, associated with the Server definition and stored in the database.

JMS

JMS Servers enable you to upload files to and download files from target JMS servers. Currently, files are transferred using JMS queues, data written to JMS queues is written as JMS messages and not as files.

Point	Keep in mind
Capabilities	• Uploads and downloads are supported.

Point	Keep in mind
	• File or directory lists are not supported. • Checkpoint Restart is not supported. • Files cannot be renamed or deleted; directories cannot be created.
Authentication	• User ID/password credentials are optional when connected to target JMS servers.
Considerations	• JMS configuration can only be configured by the MFT Command Center. • When using multiple Internet Server instances, each instance can connect to the same JMS server. Hence, all JMS queues are accessible to all Internet Server instances. • The JMS Server URL is configured globally for all Internet Server and Command Center instances. It can be overridden by selecting the "Override JMS Service Configuration" checkbox and defining the JMS Server URL in the "IP Address or fully qualified IP Name" text box. The transfer definition defines the following JMS information: • Queue used for file transfers • Input Selectors • Output JMS Type and property Types • Max Message Size

AS2

AS2 servers enable you to upload files to target AS2 servers. Downloads are not supported from AS2 servers.

Point	Keep in mind
Capabilities	• Only uploads are supported.

Point	Keep in mind
	• File or directory lists are not supported. • Checkpoint Restart is not supported. • Files cannot be renamed or deleted; directories cannot be created.
Authentication	• Credentials are optional when connected to target AS2 servers. • User ID/password credentials are not used when communicating to the target AS2 servers. • AS2 servers typically use public or private keys when communicating to AS2 servers. • Certificate authentication can typically be configured by setting the server definition to AS2 Options > HTTPS System Key parameter.
Considerations	• For more information about AS2 Configuration, see the Configuring AS2 Transfers section. • When using multiple Internet Server instances, each instance can connect to the same AS2 Server. Hence, all AS2 servers are accessible to all Internet Server instances. • Since AS2 is encrypted using public or private keys, AS2 servers typically use the HTTP protocol. However, AS2 can be configured to use HTTPS communication. • By default, the AS2 protocol does not allow you to define the file name of the incoming data. However, MFT does support the AS2 file name extension. This allows MFT to perform the following tasks: ◦ Extract the file name on incoming transfers. ◦ Define the file name on outgoing transfers.

HTTP

HTTP servers enable you to upload files to and download files from target HTTP servers.

Point	Keep in mind
Capabilities	• Uploads and downloads are supported. • File or directory lists are not supported. • Checkpoint Restart is not supported. • Files cannot be renamed or deleted; directories cannot be created.
Authentication	• Credentials are optional when connected to target HTTPS servers. • User ID/password is supported. • Certificate authentication is supported.
Considerations	• When using multiple Internet Server instances, each instance can connect to the same HTTP server. Hence, all HTTP servers are accessible to all Internet Server instances. • When configuring HTTPS, you must associate the target HTTPS server SSL key with the server definition. To associate the target HTTPS server SSL key with the server definition, perform the following steps: 1. Go to Partners > Servers > Manage Servers. 2. Select the desired HTTP server. 3. Click Retrieve HTTP public key. The public key is retrieved from the HTTPS server, associated with the server definition and stored in the database. • Uploads and downloads use standard HTTP transfer modes. Transfer definitions define whether to use Stream mode or Form/Post mode for uploads and downloads.

Microsoft Azure

Microsoft Azure Servers enable you to upload files to, and download files from target Azure Storage. The following Azure storage is supported:

- Block Blob Storage

- File Storage
- Data Lake Gen2 (ADLS Gen2)

Point	Keep in mind
Capabilities	• Uploads and downloads are supported. • File or directory lists are supported. • Azure File, Blob, ADLS Gen2 support FTP/SFTP client Restart • Platform Server client Checkpoint Restart is supported for Azure File; it is not supported for Blob or ADLS Gen2 • Files can be renamed or deleted; directories can be created.
Authentication	• Credentials are required when connected to target Azure servers. Azure Access Key • The Storage Account name is stored in the Account Name field. • The Access Key is stored in the Access Key field. Azure Active Directory • The client ID is stored in the Client ID field. • The client secret is stored in the Client Secret field. • The tenant ID should be stored in the Tenant Id field. • The account name should be stored in the Account Name field. Azure Managed Identities MFT extracts authorization tokens from Azure and uses these tokens when accessing Azure Blob or ADLS Gen2. This type of authentication only works when MFT is running on an Azure virtual machine. Use the Server Credentials Server definition to define the Account name, Object Id, and Access Key. User Delegation SAS User Delegation SAS is supported for Azure Blob and ADLS Gen2. Authentication can be performed in the following two ways:

Point	Keep in mind
	• Generate Token: Generates a User Shared Access Signature (SAS) token using Active Directory credentials. • Existing Token: Uses an existing user-delegated token to grant access to an Azure storage container. Use this to provide access that is limited for a specific time range without sharing the storage account key. Note: When you change the authentication type, the required server credential parameters also change.
Considerations	• When using multiple Internet Server instances, each instance can connect to the same Azure Server. Hence, all Azure Servers are accessible to all Internet Server instances. • By default, Microsoft Azure does not return the file "last modified date/time" for Blob or File storage. Individual calls must be made to get the file "last modified date/time", but this can delay directory lists. By default, the current date/time is displayed. If you want to get the actual "last modified date/time", set the "Microsoft Azure Options: Retrieve Last Modify" to "Yes". But this could slow down processing of directories with many files in it. There are server parameters to speed up transfers to target Azure storage: • Upload Chunk Size • Number of Upload Threads • Number of Upload Buffers Setting the "Upload Chunk Size" or the "Number of Upload Buffers" to high values can cause timeouts when the Client FTP or SFTP connection is faster than the connection to the Azure server.

Amazon S3

Amazon S3 servers enable you to upload files to and download files from target Amazon S3 storage.

Point	Keep in mind
Capabilities	• Uploads and downloads are supported. • File or directory lists are supported. • Checkpoint Restart is not supported. • Files can be renamed or deleted; directories can be created.
Authentication	MFT supports the following authentication methods when transferring files with Amazon S3 storage: • Secret Key: Access Key ID is stored in the Server Credential Access Key ID, Secret Access Key is stored in the Server Credential Secret Access Key. • EC2 MetaData: MFT extracts authorization tokens from the Amazon EC2 image and uses these tokens when accessing S3 storage. This only works when MFT is running on an Amazon EC2 image and the EC2 image has been configured with rights to Amazon S3 storage. • SAML IDP Form: Extracts authorization assertions by simulating a log in to Amazon.
Considerations	• When using multiple Internet Server instances, each instance can connect to the same Amazon S3 Server. Hence, all Amazon S3 Servers are accessible to all Internet Server instances. There are server parameters to speed up transfers to target Azure storage: • Upload Chunk Size • Number of Upload Threads • Number of Upload Buffers Setting the "Upload Chunk Size" or the "Number of Upload Buffers" to high values can cause timeouts when the Client FTP or SFTP connection is faster than the connection to the Amazon S3 server.

Google Cloud

Google Cloud Servers enable you to upload files to, and download files from defined buckets or datasets. The following Google Cloud products are supported:

- Cloud Storage
- BigQuery

Common Points

Point	Keep in mind
Authentication	MFT supports the following authentication methods when transferring files with Google Cloud servers. • Workload Identity Federation (WIF): WIF provides on-premises or multicloud workloads with access to Google Cloud resources by using federated identities. TIBCO® Managed File Transfer Command Center supports only On-premises Active Directory Federation Services (AD FS) workload identity pool provider. Create a workload identity pool in Google console. Add the provider and grant access to service accounts in the workload identity pool. In MFT, the following data has to be specified for WIF: ◦ Google ProjectID: Defines the Google Project ID. ◦ Identity Provider URL: Defines the AD FS URL required to get the token. ◦ Client ID: Defines the AD FS client ID required to get the token. ◦ Client Secret: Defines the AD FS client secret required to get the token. ◦ Scope: Defines the AD FS scope required to get the token. ◦ JSON File Content: When the authentication type is Workload Identity Federation, it defines the pool identities that enable access to Google Cloud resources. • Service Account: An authentication key associated with a service account to access Google Cloud resources. ◦ JSON File Content: When the authentication type is Service Account, it defines the JSON Service Account Key associated with the Service Account. This JSON data is used as credentials to access the Google Cloud products.

Note: The **Server Credentials** tab is not available for Google Cloud server type.

Considerations	When you use multiple Internet Server instances, each instance can connect to the same Google Cloud Server. As a result, all Google Cloud services are available to every Internet Server instance. To optimize transfer speeds to Google Cloud servers, adjust the following server parameters: • Upload Chunk Size • Number of Upload Buffers Setting either the Upload Chunk Size or the Number of Upload Buffers to high values can cause timeouts if the client FTP or SFTP connection is faster than the connection to the Google Cloud server.
----------------	--

Google Cloud Storage

Point	Keep in mind
Capabilities	• Uploads and downloads are supported. • File or directory lists are supported. • Checkpoint Restart is not supported. • Files can be renamed or deleted; directories can be created. • Renaming directories are not supported.

BigQuery

Point	Keep in mind
Capabilities	• Uploads are supported. AVRO, CSV, JSON, ORC and Parquet file types are supported. • Downloads are not supported. • File or directory lists are supported.

Point	Keep in mind
	• Checkpoint Restart is not supported. • Files can be deleted. • Renaming files or directories are not supported. • Creating or deleting directories is not supported.

HDFS

HDFS Servers enable you to upload files to and download files from target HDFS or Hadoop Servers. HDFS support is limited and supports only basic upload and downloads.

Point	Keep in mind						
Capabilities	• Uploads and downloads are supported. • File or directory lists are supported. • Checkpoint Restart is not supported. • Files can be renamed or deleted; directories can be created.						
Authentication	MFT supports the following authentication methods when transferring files with HDFS Servers: <table> <tr> <th>Authentication Method</th><th>Description</th></tr> <tr> <td>Simple</td><td>Only the user ID is validated. Passwords are not validated. This server is typically used for testing. If this is used, then some other mechanism should be defined for validation, such as limiting transfers to specific IP Addresses.</td></tr> <tr> <td>Kerberos</td><td>A keytab is defined and is used to authenticate to a Kerberos server.</td></tr> </table>	Authentication Method	Description	Simple	Only the user ID is validated. Passwords are not validated. This server is typically used for testing. If this is used, then some other mechanism should be defined for validation, such as limiting transfers to specific IP Addresses.	Kerberos	A keytab is defined and is used to authenticate to a Kerberos server.
Authentication Method	Description						
Simple	Only the user ID is validated. Passwords are not validated. This server is typically used for testing. If this is used, then some other mechanism should be defined for validation, such as limiting transfers to specific IP Addresses.						
Kerberos	A keytab is defined and is used to authenticate to a Kerberos server.						

Point	Keep in mind
	The Server Credentials tab is ignored for HDFS Servers.
Considerations	• All HDFS Servers must use either Simple authentication or Kerberos Authentication. You cannot define some HDFS servers as Simple and some as Kerberos. • When using multiple Internet Server instances, each instance can connect to the same HDFS Server. Hence, all HDFS Services are accessible to all Internet Server instances.

FileShare

FileShare enables MFT clients (FTP, SFTP, HTTP, Platform Server) to send files to or receive files from the File Share component of MFT.

Point	Keep in mind
Capabilities	• Uploads and downloads are supported. • File or directory lists are supported. • Checkpoint Restart is not supported. • Files can be renamed or deleted; directories can be created based on the FileShare rights the user has for the defined folder.
Authentication	MFT does not require authentication to the FileShare server. The user that initiates the transfer must be a FileShare user with the following folder rights: • Edit or Admin rights to allow uploads, renames, and deletes. • View, Edit or Admin rights to allow downloads. The Server Credentials tab is ignored for FileShare servers.
Considerations	• Transfer definitions that use FileShare servers must configure the FileShare user folder to point to the position in the FileShare where files can be transferred.

Point	Keep in mind
	- When using multiple Internet Server instances, each instance can connect to the FileShare Server. Hence, all FileShare Services are accessible to all Internet Server instances. - You can use this capability to automate sending files to or receiving files from FileShare folders.

Email Server

MFT Clients (FTP, SFTP, HTTP, Platform Server) can use Email Server to send files to target recipients as email attachments.

i Note: When customers want to restrict files from being sent as email attachments, you can use the `AllowEmailServerDefinition` web.xml parameter to disable the ability to define email servers or initiate transfers to an email server. The default value `true` allows Email transfers; `false` does not allow Email transfers.

Point	Keep in mind
Capabilities	- Downloads are not supported. - File or directory lists are not supported. - Checkpoint Restart is not supported. - PGP is supported.
Authentication	Depending on the target SMTP Server, authentication might be required. Use the Server definition "Server Credentials" to define the user ID/password for the defined SMTP Server.
Considerations	- You can set limits for the attachment file size. We recommend using this capability for small files only. - Any server that can access the defined SMTP Server can send files as email attachments.

Point	Keep in mind
	• Only one attachment can be sent to a recipient as an email attachment. • PGP encryption can be used to encrypt attachments so that only the defined recipients can decrypt the attachment. • PGP is supported only for defined MFT users that have a PGP associated with the recipient. • Emails can be sent to defined MFT users or to any email address. The server definition "Email Options: Send only to defined users" parameter defined whether email attachments can be set to any email address or only to defined MFT users. • Tokens can be used to override email parameters on Transfer definition email Options: Recipients, Subject, and Message Text. • You can use this capability to automate sending files to target email users.

Mailbox

MFT Clients (FTP, SFTP, HTTP, Platform Server) can use Mailbox to send files to target recipients as mailbox attachments.

i Note: With the `AllowMailboxServerDefinition web.xml` parameter, you can disable the functionality to define email servers or initiate transfers to an email server. This is required when customers want to restrict files from being sent as email attachments. The default value `true` allows Email transfers; `false` does not allow Email transfers.

Point	Keep in mind
Capabilities	• Downloads are not supported. • File or directory lists are not supported. • Checkpoint Restart is not supported.

Point	Keep in mind
	• PGP is not supported.
Authentication	Authentication is not required. Client users must be defined as FileShare or mailbox users.
Considerations	• You can set limits for the attachment file size. We suggest using this capability for small files only. • Any Internet Server instance files as Mailbox attachments. • Only one attachment can be sent to a Mailbox attachment. • Mailbox attachments can be sent to defined MFT users or to any email address, depending on the Client user type. • Power users can send Mailbox attachments to Full, Power, or Guest users and can create Full and Guest users. • Full users can send Mailbox attachments to Full, Power, or Guest users and can create Guest users. • Guest users can send Mailbox attachments to Full and Power users. • Tokens can be used to override email parameters on Transfer definition email Options: Recipients, Subject, Message Text. • You can use this capability to automate sending files to target mailbox users.

Four Eyes

MFT Clients (for example, FTP, SFTP, HTTP, and Platform Server) can use Four Eyes to send files to target recipients as Four Eyes attachments.

Point	Keep in mind
Capabilities	• Downloads are not supported. • File or directory lists are not supported. • Checkpoint Restart is not supported.

Point	Keep in mind
	• PGP is supported.
Authentication	Authentication is not required.
Considerations	• You can set limits for the attachment file size. Use this capability for small files only. • A Four Eyes Approver must approve the request before the recipient can view and download the attachment. • You can automate sending files to target Four Eyes recipients along with approvers.

Custom Server

MFT customers can use custom servers to write java code to support target server protocols not supported by MFT Internet Server.

Point	Keep in mind
Capabilities	• Uploads and downloads are supported. • File or directory lists are supported. • Checkpoint Restart is supported. • PGP is supported. Support for the above capabilities depends on the implementation created by the MFT customer.
Authentication	Authentication is up to the implementation. You can pass the user ID, password, and domain tokens to the implementation through the Server definition "Custom Server Options: Configuration data" parameter.
Considerations	• MFT supplies sample code for the Customer Server Framework in this folder: <code><MFT-Install>/server/webapps/cfcc/example/customTransfers</code>

Point	Keep in mind
	This directory includes the following steps: • Java Doc • Build procedures • A sample implementation that transfers files to or from a local directory. • It is up to the customer to write Java code for all of the required features. • All configuration information is defined through the Server definition "Custom Server Options: Configuration data" parameter.

SharePoint Server

SharePoint servers enable you to upload files to and download files from target SharePoint servers.

Point	Keep in mind
Capabilities	• Uploads and downloads are supported. • Directory navigation is supported within a document library. You cannot navigate through sites, sub-sites, and document libraries. • File or directory lists are supported. • Checkpoint Restart is not supported. • PGP is supported.
Authentication	Azure Active Directory The client ID is stored in the Client Id field. The client secret is stored in the Client Secret field. The tenant ID should be stored in the Tenant Id field. The account name should be stored in the Account Name field.

Point	Keep in mind
Considerations	- When using multiple Internet Server instances, each instance can connect to the same SharePoint server. Hence, all files are accessible to all Internet Server instances. - To access SharePoint documents, you must define the SharePoint server URL for the document library. The SharePoint server URL can be defined in the following two ways: - In the Server Definition > Required Server Information > SharePoint Server URL. - By appending the Transfer Definition > SharePoint Properties > SharePoint Document Library Url to the Server Definition > Required Server Information > SharePoint Server URL. - Using the second option allows a single server definition with a single set of credentials, to access multiple sites, sub-sites and Document libraries.

OFTP2 Server

OFTP2 servers enable you to upload files to target OFTP2 servers.

Point	Keep in mind
Capabilities	- Uploads are supported. Downloads are not supported. - Directory navigation is not supported. - File or directory lists are not supported. - Checkpoint Restart is not supported. - PGP is supported.
Authentication	Authentication of incoming requests is performed by matching the incoming request with the following fields:

Point	Keep in mind
	• Partner Odette ID • Partner Password (optional) • Client TLS Server Certificate (optional) • Session Authentication (optional)
Considerations	When an incoming request is received, MFT matches the Partner's Odette ID against the Partner Odette ID of defined server definitions. When a match is found, the user ID used for the transfer is picked from the User ID for incoming requests server definition field. Transfer definitions for this user are used in the file transfer request from the OFTP2 client.

For more information on configuring OFTP2, see the [Configuring OFTP2 Transfers](#) section.

Creating a New Transfer Definition

Once you define the server definition, complete the following instructions to give a user access to files on that server:

Procedure

1. Go to **Transfers > Internet Server Transfers > Add Transfer**.
2. Enter the required transfer information as described below:

Field Name	Description
Server File Name	Define the directory where you want to upload files to or download files from.
Authorized User Id	Select the client user that requires access. This can also be done through the Authorized Group Id .
Authorized Group Id	Select the Group that requires access. This can also be done through Authorized User Id .

Field Name	Description
Transfer Direction	Set to Upload , Download , or Both as needed.
Virtual Alias	Set to a Unique Virtual Alias for that user.

3. Click **Add**.

Viewing Audit Records

Audit records are displayed for every file transfer which include both successful and failed file transfers.

In order to view audit records, you must have one of the following rights:

- AdministratorRight
- ViewAuditRight

Audit records can only be written if authentication to the MFT Internet Server is successful and the file transfer has started. For example, if an SCP client attempts to perform a file transfer but the authentication fails, no MFT audit record is written. Similarly, if a Platform Server client attempts to perform a file transfer but authentication fails, no MFT audit record is written. However, the Local Platform Server client can write an audit record.

To view audit records, complete the following steps:

Procedure

1. On the landing page, go to **Reports > Audits > Search Audits**.

i Note: The most recent 100 audit records are displayed in the **Results** table.

2. Use **Search Criteria** to search for transfers matching the defined criteria.

i Note: You can enter the date ranges for your search at the bottom of the **Search Criteria**. Alternatively, you can set the **Number of Days** parameter to define the number of days that need to be searched.

3. After you have finished entering search criteria, click **Search**.
Transfers matching the selection criteria are displayed in a table.
4. Click the **Audit Id** link which is next to the transfer to view its details. Information is displayed about the user, transfer, client, server, and many other transfer properties.
5. Click **Back to Audit Search List** to return to the **Audit Search** page.

Configuring and Starting MFT Transfer Services

MFT Internet Server supports the following incoming file transfer protocols:

- HTTP/HTTPS
- FTP/FTPS
- SSH (SFTP)
- Platform Server
- AS2
- OFTP2

HTTPS transfers are handled by the Application Server HTTP or HTTPS connectors. HTTP or HTTPS transfers are enabled by default. Note that the HTTP protocol is disabled by default. Hence, clients must use HTTPS.

Incoming AS2 requests use the HTTP or HTTPS protocol but you must still configure the MFT AS2 Server for AS2 transfers to work. See [Configuring AS2 Transfers](#) for more information.

The other protocols must be configured and the service must be started. This document will explain in detail how to get the SSH (SFTP) service configured and started. Later in this document, we will explain the steps needed to configure and start FTP and Platform Server transfer services.

 **Note:** In this document, the term SSH refers to the term SFTP.

To configure and start MFT transfer services, complete the following steps:

Procedure

1. [Configure and Start the SSH Service.](#)
2. [Configure and Start the FTP Service.](#)

3. [Configure and Start the Platform Server service.](#)
4. [Configure and Start the AS2 Service.](#)
5. [Configure and Start the OFTP2 Service.](#)

Configuring and Starting the SSH Service

To configure and start the SSH service, you must first create an SSH system key for the SSH service.

Creating an SSH System Key

To create an SSH system key, perform the following steps:

Procedure

1. Go to **Management > Protocol Keys > System Keys > Create System Key.**
2. Enter the required information described in the table below:

Field	Instruction
System Key Type	Set to SSH system key.
Description	Set to a unique value for system keys.
Password	Set to a secure password.
Expiration Date	Set this based on your installation's security requirements.
Key Size	Set to 2048 bits or higher.
Signing Algorithm	Set to SHA-256 or SHA-512.
Set as Default Key	Select the checkbox if you want this key to be the default SSH key.
Common Name	Set to the common name of the server.

i Note: Common Name is not validated during SSH key exchange. It is used for information purposes only.

3. After entering the information, click the **Create Key** button.

Configuring the SSH Server

To configure the SSH server, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > SSH Server > Configure SSH Server**.
2. Select the Internet Server instance you want to configure.
3. Enter the required information described in the table below:

Field	Instruction
Enabled	Set to Yes.
IP Port	Set to the desired IP port.
SSH System Key	Select from 1 to 6 SSH system keys. RSA, Ed25519, Ed448, ECDSAP256, ECDSAP384 and ECDSAP521 are the currently supported key types. Only one key per algorithm can be selected.
Key or Certificate	Set to Key . Very few SFTP clients support SSH certificates.
Welcome Message	Set a generic welcome message. Note: Many SFTP clients do not display the 'Welcome' message.

i Note: On UNIX machines, only root users can start ports below 1025. For best results, use an SSH port (for example, 2022) to run the MFT Internet Server, instead of running it as a root user. SSH clients can connect to port 2022, or they can connect to a passthrough load balancer using port 22 (the standard SSH port), and the load balancer can redirect the request to port 2022. Administrators can also configure an iptables command to route incoming data on port 22 to port 2022.

Starting the SSH Server

To start the SSH server, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > SSH Server > SSH Server Status**.
2. Select the Internet Server instance where you want to start the SSH service.
3. Click the **Status** button to get the current status of the SSH server service.
4. Click the **Stop** button to stop the SSH server service.
5. Click the **Start** button to start the SSH server service.

i Note: If the SSH service does not start, the `catalina.out` file might have some information about why the service did not start.

A service might not start for the following reasons:

- You specified a port below 1025, but are not a root user.
- The SSH port defined in the **Configure SSH Server** page is already in use by another process.
- The SSH system key is expired or is less than 2048 bits.

Configuring and Starting the FTP Service

To configure and start the FTP service, you must first create an FTP system key for the FTP service.

i Note: If you do not define the FTPS port and will not support Explicit SSL, you do not need to create an FTPS system key.

Creating an FTP System Key

To create an FTP system key, perform the following steps:

Procedure

1. Go to **Management > Protocol Keys > System Keys > Create System Key**.
2. Enter the required information described in the table below:

Field	Instruction
System Key Type	Set to FTP system key.
Description	Set to a unique value for system keys.
Password	Set to a secure password.
Expiration Date	Set this based on your installations security requirements.
Key Size	Set to 2048 bits or higher.
Signing Algorithm	Set to SHA-256 or SHA-512.
Set as Default Key	Select the checkbox if you want this key to be the default SSH key.
Common Name	Set to the common name of the server.

i Note: Common Name is not validated during SSH key exchange. It is used for information purposes only.

3. After entering the information, click the **Create Key** button.

Configuring the FTP Server

To configure the FTP server, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > FTP Server > Configure FTP Server**.
2. Select the Internet Server instance you want to configure.

Note:

- FTP services are configured similar to SSH services. However, due to the fact that FTP uses two TCP connections (Control and Data) to perform file transfers, there are some differences.
- If you are using FTPS, you must create an FTP system key. See *Creating an FTPS system key*.

3. Enter the required information described in the table below:

Field	Instruction
Enabled	Set to Yes.
IP Port	Used for clear text and explicit SSL.
SSL Port	Used for implicit SSL.
FTP System Key	Select the FTPS system key or set to Use Default .
Welcome Message	Set a generic welcome message. Note: Many SFTP clients do not display the 'Welcome' message.
Use External IP Address	Enter the IP address of the Internet Server machine.
External IP Address	Enter the IP address of the Internet Server machine.

Updating Global FTP Parameters

To update the global FTP parameters, perform the following steps:

Procedure

1. Go to **Configuration > System Configuration > Global FTP Settings**.
2. Enter the required information described in the table below:

Field	Instruction
Limit Local Port	Set to Yes.
Starting Port	Select a port (after discussing with the network team).
Number of Ports	Select a number of ports to use.

Note: 100 is a good starting point.

3. After entering the information, click the **Update** button.

i Note: On UNIX machines, only root users can start ports below 1025. For best results, use an SSH port (for example, port 2021) to run the MFT Internet Server, instead of running it as a root user. FTP clients can connect to port 2021, or they can connect to a passthrough load balancer using port 21 (the standard FTP port), and the load balancer can redirect the request to port 2021. Administrators can also configure an iptables command to route incoming data on port 21 to port 2021.

Starting the FTP Server

To start the FTP server, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > FTP Server > FTP Server Status**.
2. Select the Internet Server instance where you want to start the FTP service.

3. Click the **Status** button to get the current status of the FTP server service.
4. Click the **Stop** button to stop the FTP server service.
5. Click the **Start** button to start the FTP server service.

i Note: If the FTP service does not start, the `catalina.out` file might have some information about why the service did not start.

The following points describe a few reasons why a service will not start.

- You specified a port below 1025 but are not a root user.
- The FTP port defined in the **Configure FTP Server** page is already in use by another process.
- The FTP system key is expired or is less than 2048 bits.

Configuring and Starting the Platform Server Service

To configure and start the Platform Server service, you must first create a Platform Server system key for the Platform Server service.

i Note: If you do not define the SSL or SSL tunnel port, you do not need to create an Platform Server system key.

Creating an Platform Server System Key

To create a Platform Server system key, perform the following steps:

Procedure

1. Go to **Management > Protocol Keys > System Keys > Create System Key**.
2. Enter the required information described in the table below:

Field	Instruction
System Key Type	Set to Platform Server system key.
Description	Set to a unique value for system keys.
Password	Set to a secure password.
Expiration Date	Set this based on your installations security requirements.
Key Size	Set to 2048 bits or higher.
Signing Algorithm	Set to SHA-256 or SHA-512.
Set as Default Key	Select the checkbox if you want this key to be the default Platform Server system key.
Common Name	Set to the common name (that is, the DNS) of the server.

3. After entering the information, click the **Create Key** button.

Configuring the Platform Server Service

To configure the Platform Server service, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > Platform Server > Configure Platform Server**.
2. Select the Internet Server instance you want to configure.

Note:

- Platform services are configured similar to SSH services. However, due to the fact that Platform Server uses three TCP listeners (normal, SSL, and TLS tunnel), there are some differences.
- If you are defining SSL or TLS tunnel ports, you must create an FTP system key. See *Creating an FTPS system key*.

3. Enter the required information described in the table below:

Field	Instruction
Enabled	Set to Yes.
IP Port	Non-SSL port. Set to 48484 unless that port is already in use.
SSL Port	Connections are validated using TLS and data is transmitted using AES 256 encryption. Set to 58585 unless that port is already used.
TLS Tunnel Port	Data is transferred over a secure TLS connection. Set to 59595 unless that port is already used.
Platform Server System Key	Select a system key or use the default system key. Note: If you define the SSL port or the TLS tunnel port, you must create a Platform Server system key.

4. After entering the information, click the **Update** button.

Starting the Platform Server Service

To start the Platform Server service, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > Platform Server > Platform Server Status**.
2. Select the Internet Server instance where you want to start the Platform Server

service.

3. Click the **Status** button to get the current status of the Platform Server service.
4. Click the **Stop** button to stop the Platform Server service.
5. Click the **Start** button to start the Platform Server service.



Note: If the Platform Server service does not start, the `catalina.out` file might have some information about why the service did not start.

The following points describe a few reasons why a service will not start.

- The Platform Server port defined in the **Configure Platform Server** page is already in use by another process.
- The Platform Server system key is expired or is less than 2048 bits.

Configuring PGP

By using PGP (Pretty Good Privacy), you can encrypt, compress, and sign data. MFT fully supports PGP on all of its transfer protocols (both incoming and outgoing).

For more information about PGP, see the following topics:

- [About PGP](#)
- [To Configure a PGP](#)

About PGP

This section describes the process of configuring PGP for MFT Internet Server transfers.

 **Note:** MFT Internet Server performs PGP encryption and decryption in a streamed mode, not in a store and forward mode. Interim data is never written to disk.

Before starting with the MFT instructions, let us look at the following example to understand the basic overview of PGP. Let us assume that user A wants to PGP encrypt and sign a file that will be sent to user B.

Before we start to encrypt and decrypt data:

- User A needs to send its PGP public key to user B.
- User B needs to send its PGP public key to user A.

How PGP Works

- User A uses PGP encryption to send data using the public key of user B.
- User A uses PGP sign to send the data with the private key of user A.
- User B decrypts the data using its system key. Only users with the system key and passphrase associated with the system key that encrypted the file can decrypt the

file. That is what makes PGP so secure.

- User B verifies the file signature using the public key associated with the private key of User A. Hence, any user with access to the public key of User A can verify the signature.
- The advantages of PGP file encryption and compression are simple. Signing files is not as simple. Signing provides non-repudiation of files. It verifies that a file comes from a trusted source.

When to Use PGP?

See the following guidelines on when to use PGP:

- A file contains confidential or secure data.
- A file contains financial transactions.
- To secure files sent through unsecure protocols like FTP.
- To save data on disk in a PGP-encrypted format.

i Note: Many customers use PGP encryption to double-encrypt data. You can use a secure protocol like SSH (SFTP). All data transmitted over SFTP is encrypted. PGP encrypts and signs the file to be sent. Using SSH with PGP double-encrypts the data and provides the most secure way to transfer critical data. PGP signatures verify that the file was encrypted and signed by a valid sender.

To Configure a PGP

MFT Internet Server supports PGP encryption, compression, and signing in a streamed mode. The MFT Internet Server supports PGP encryption in the following ways:

- When a client sends or receives PGP encrypted files to MFT Internet Server or from MFT Internet Server. MFT uses PGP to decrypt data received from the client and MFT uses PGP to encrypt data sent to the client. In this case, you must associate the user's PGP public key with the user performing the file transfer.
- When MFT Internet Server sends or receives PGP encrypted files to a target server or from a target server. MFT uses PGP to decrypt data received from the server and MFT uses PGP to encrypt data sent to the server. In this case, you must associate the

target server's PGP public key with the server configured in the transfer definition.

The methods in which MFT Internet Server supports PGP encryption are configured differently.

To configure PGP, perform the following steps:

Procedure

1. [Create a PGP system key.](#)
2. [Configure the public key: Client sends or receives PGP-encrypted data.](#)
3. [Configure the public key: Client sends or receives PGP-encrypted data with target server.](#)
4. [Client sends PGP-encrypted data to MFT Internet Server.](#)
5. [Client receives PGP-encrypted data from MFT Internet Server.](#)
6. [MFT Internet Server sends PGP-encrypted data to a target server.](#)
7. [MFT Internet Server receives PGP-encrypted data from a target server.](#)

Creating a PGP System Key

To create a PGP system key, perform the following steps:

Procedure

1. Go to **Management > PGP Keys > System Keys > Create PGP Key.**
2. Enter the required information described in the table below:

Field	Instruction
Description	Set to a unique description.
Pass Phrase	Enter a secure pass phrase.
Key Size	Select a key size of 2048 or greater.

Field	Instruction
Key Type	Supported values are DSA and ElGamal, RSA, Ed25519, Ed448, ECDSAP384, and ECDSAP521. RSA and ECDSA support FIPS 140.
Hashing Algorithm	Recommend using SHA-256 or SHA-384.
Set as Default Key	Select the checkbox to set this key as the default PGP.
Real Name	Defines a name to be associated with the key.
Email Address	Enter an email address.

3. Click the **Create Key** button to create a new PGP system key with the given details.

What to do next

Extract the PGP public key information and send it to your transfer partner.

Extracting the PGP Public Key

To extract the PGP public key, perform the following steps:

Procedure

1. Go to **Management > PGP Keys > System Keys > Manage PGP Keys**.
2. Click the **Description** of the key that you just created.
3. Click the **PGP Keys** tab.
4. Copy the public key from the public key text area.

PGP public keys usually look like the following:

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
. . . . . pgp public key data. . . . .
-----END PGP PUBLIC KEY BLOCK-----
```

5. Email this key to the transfer partner. Since this is a public key, it does not contain secure information.

i Note: This key does not contain secure information. However, you can use it to encrypt a file that only your PGP system key can decrypt. You must sign files and verify the required signatures, which ensures that a file is encrypted by the correct partner.

Configuring the Public Key: Client Sends or Receives PGP-Encrypted Data

Before we configure the transfers to support PGP, we must associate the client's PGP public key with the user performing the transfer. The user's PGP public key is used for two purposes:

- To verify the signature of the user that encrypted the file.
- To encrypt data sent to the user.

To configure the public key, perform the following steps:

Procedure

1. Go to **Management > PGP Keys > Public Keys > Add PGP Key**.
2. Enter the required information described in the table below:

Field	Instruction
Apply Key to	Select user.
Select User	Select the user that initiates the transfer.
Status	Set to Enabled.
Set as Default Key	Sets key that is used to encrypt data.
PGP Text Area	Enter the PGP public key in the text box.

3. Click **Continue** to configure the public key.

i Note: Follow the UI instructions until the key is added.

Configuring the Public Key: Client Sends or Receives PGP-Encrypted Data With Target Server

Before we configure the server to support PGP, we must associate the PGP public key of the target server with the server definition.

i Note: To apply a PGP public key to a server definition, ensure that **Server > PGP Information > PGP Enabled** checkbox is selected.

The server PGP public key is used for the following purposes:

- To verify the signature of the target server that encrypted the file
- To encrypt data sent to the target server

To configure the public key, perform the following steps:

Procedure

1. Go to **Management > PGP Keys > Public Keys > Add PGP Key**.
2. Enter the required information described in the table below:

Field	Instruction
Apply Key to	Select server.
Select Server	Select the target server that initiates the transfer.
Status	Set to Enabled.

Field	Instruction
Set as Default Key	Sets key that is used to encrypt data.
PGP Text Box	Enter the PGP public key in the text box.

3. Click **Continue** to configure the public key to the target server.



Note: Click **OK** or **Continue** as needed until the key is added.

Only PGP-related parameters are discussed in the following four use cases:

- [Client Sends PGP-Encrypted Data to MFT Internet Server](#)
- [Client Receives PGP-Encrypted Data from MFT Internet Server](#)
- [MFT Internet Server Sends PGP-Encrypted Data to a Target Server](#)
- [MFT Internet Server Receives PGP-encrypted Data from a Target Server](#)

Let's assume that the transfer and server definitions have already been created.

Client Sends PGP-Encrypted Data to MFT Internet Server

Make sure that a public key is associated with the client user. See [Client Sends or Receives PGP encrypted data](#) for more detail.

Configure the transfer definition to decrypt and optionally, verify the PGP signature. The transfer definition contains the PGP configuration information.

To send the PGP-encrypted data to the MFT Internet Server, perform the following steps:

Procedure

1. Go to **Transfers > Internet Transfers > Manage Transfers**.
2. On the **Manage Transfers** page, click **Transfer Id**.
3. Go to the **PGP Information** tab.

4. Configure the following fields described in the table below:

Field	Instruction
Decrypt	Select this checkbox to decrypt the data.
Verify Signature	Select this checkbox to verify whether the signature is valid for any user.
Verify User Signature	When checked, verifies if the signature is valid for the transfer user.

5. Click **Update** to update the transfer definition.



Note: When you perform the upload for this transfer definition, the data is decrypted using the MFT PGP private key. The signature is validated using the PGP public key of any user defined to MFT, or to the user that performed the file upload.

Client Receives PGP-Encrypted Data from MFT Internet Server

Make sure that a public key is associated with the client user. See [Client Sends or Receives PGP encrypted data](#) for more detail.

Configure the transfer definition to encrypt and optionally, sign the PGP signature. The transfer definition contains the PGP configuration information.

Procedure

1. Go to **Transfers > Internet Transfers > Manage Transfers**.
2. On the **Manage Transfers** page, click **Transfer Id**.
3. Go to the **PGP Information** tab.
4. Configure the following fields described in the table below:

Field	Instruction
Private Key	Defines the PGP system key used to sign the data.
Encrypt	Select this checkbox to encrypt the data.
Sign	Prompts MFT to sign the data.
ASCII Armor	Prompts MFT to encrypt the data using ASCII Armor format.

5. Click **Update** to update the transfer definition.

 **Note:** When you perform the download for this transfer definition, the data is encrypted using the **Transfer Users Public Key**. The file is signed using the Private Key defined on this page.

MFT Internet Server Sends PGP-Encrypted Data to a Target Server

Ensure that a public key is associated with the target server. For more information, see [Client Sends or Receives PGP encrypted data](#).

Configure the server definition to encrypt and optionally, sign the PGP signature. The server definition contains the PGP configuration information.

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Click the server name of the target server.
3. Go to the **PGP Information** tab.
4. Configure the following fields described in the following table:

Field	Instruction
PGP Enabled	Yes indicates that all uploads and downloads are PGP encrypted.
Private Key	Defines the PGP system key used to sign the data.
Encrypt	Select this checkbox to encrypt the data.
Sign	Prompts MFT to sign the encrypted data.
ASCII Armor	Prompts MFT to encrypt the data using ASCII Armor format.

5. Click **Update** to update the server definition.



Note: When an upload is performed to this server, the data is encrypted using the public key associated with this server definition. The file is signed using the private key defined on this page.

MFT Internet Server Receives PGP-encrypted Data from a Target Server

Ensure that a public key is associated with the target server. For more information, see [Client Sends or Receives PGP encrypted data](#).

Configure the server definition to decrypt and optionally, verify the PGP signature. The server definition contains the PGP configuration information.

To receive the PGP-encrypted data to the MFT Internet Server, perform the following steps:

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Click the server name of the target server.
3. Go to the **PGP Information** tab.
4. Configure the following fields described in the following table:

Field	Instruction
PGP Enabled	Yes indicates that all uploads and downloads are PGP encrypted.
Verify Signature	Prompts MFT to verify the signature using any defined PGP public key.
Verify Server Signature	Verifies this signature is valid for PGP keys associated with this server

5. Click **Update** to update the server definition.

i Note: When downloading from this server, the data is decrypted using the MFT PGP system key. The file signature is verified using any PGP public key, or optionally only with PGP keys associated with this server.

Configuring MFT for Key or Certificate Authentication

MFT supports key or certificate authentication for incoming and outgoing requests. An incoming request is when a client connects to MFT Internet Server or Command Center. Outgoing requests are when MFT Internet Server connects to a target server.

Incoming Request Key or Certificate Protocol Support

The following table lists the incoming request keys or certificate protocols that are supported.

Request Key or Certificate Protocol	Support
HTTPS	Certificate authentication.
FTPS	Certificate authentication.
SSH/SFTP	Key or certificate authentication.
Platform Server	For SSL and TLS tunnel requests, certification authentication.

Outgoing Request Key or Certificate Protocol Support

The following table lists the incoming request keys or certificate protocols that are supported.

Request Key or Certificate Protocol	Supports
HTTPS	Certificate authentication.

Request Key or Certificate Protocol	Supports
FTPS	Certificate authentication.
SSH/SFTP	Key or certificate authentication.
Platform Server	For SSL and TLS tunnel requests, certification authentication.

Incoming HTTPS Client Requests Certificate Authentication

HTTPS certificate authentication for incoming requests is more difficult to implement than other protocols because a private SSL key or TLS key must be configured and installed in the browser certificate manager. The functionality of each browser varies. Hence, creating and configuring keys in the browser is not discussed in this guide. This guide concentrates on how MFT supports HTTPS certificate authentication.

Configuring the MFT Server for HTTPS Certificate Authentication

The MFT Tomcat server HTTPS connector must be configured to request a certificate from the browser. This must be done directly from the computer where Internet Server or Command Center is installed.

To configure the HTTPS connector to request a certificate, perform the following steps:

Procedure

1. Log in to the server where Internet Server or Command Center is installed.
2. Run the `cd <MFT-Install>/server/conf` command.
3. Make a backup copy of the `server.xml` file.

4. Edit the `server.xml` file.
5. Configure the `ClientAuth` parameter described in the table below:

Field	Instruction
false	No certificate is requested.
want	Request a certificate. If no certificate is available, the request continues without a certificate. If allowed, the user can log in with the User ID and password credentials.
true	Requires a certificate. If no certificate is available, the request fails.

Example:

```
<Connector SSLEnabled="true" URIEncoding="UTF-8" acceptCount="128"
ciphers="TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,"
. . . . .
clientAuth="want"
. . . . .
server.tomcat.uri-encoding="utf-8" socket.txBufSize="131072"
sslEnabledProtocols="TLSv1.2" sslProtocol="TLS" tcpNoDelay="true"
trustManagerClassName="com.proginet.sift.tomcat.ssldap.TrustAllMg
r"/>
```

6. Save the `server.xml` file.
7. Restart the MFT Server by running the following commands:

UNIX:

```
cd <MFT-Install>/server/bin
./shutdown.sh
```

Windows:

```
cd <MFT-Install>/server/bin
shutdown
```

8. After the server shutdown is complete, start the MFT Server.

UNIX:

```
./startup.sh
```

Windows:

```
startup
```

Configuring the MFT Server to Allow or Require HTTPS Certificate Authentication

There are two ways to configure users for HTTPS certificate authentication:

1. Applying global HTTPS setting to all users.
2. Applying HTTP authentication only to individual users.

Applying Global HTTPS Setting to all Users on all MFT Servers

To apply global HTTPS setting to all users on all MFT servers, perform the following steps:

Procedure

1. Go to **Configuration > System Configuration > Global HTTPS Settings**.
2. Set the **HTTPS Client Authentication Method** parameter to **Certificate or Password**.

i Note: We do not suggest setting this parameter to **Certificate Only** or **Certificate and Password**, since all HTTPS log ins require a browser certificate. The means that even admin users require certificate authentication.

Applying HTTP Authentication only to Individual Users

To apply HTTP authentication only to individual users, perform the following steps:

Procedure

1. Go to **Partners > Users > Manage Users**.
2. Select the user that you want to configure.
3. On the **Authentication Options** tab, set the **HTTPS Client Authentication Method** parameter to **Certificate Only** or **Certificate and Password**.
4. Click **Update** to save the change.

Associating a Public Key with an MFT User

The user must extract the certificate component of the HTTP system key that they use to connect to the MFT server.

To associate a public key with an MFT user, perform the following steps:

Procedure

1. Go to **Management > Protocol Keys > Public Key > Add Key**.
2. Configure the following parameters described in the table below:

Parameter	Instruction
Public Key Type	HTTPS public key.
Apply Key to	User.
Select user	Select the user that requires HTTPS certificate authentication.
Status	Enabled.
Description	Enter a unique description.

Enter the X.509 Paste the X.509 certificate to this box.

The following is an example of an X.509 key:

```
-----BEGIN CERTIFICATE-----
. . . . base 64 encode data . . . .
-----END CERTIFICATE-----
```

3. Click **Continue**.

A confirmation page is displayed.

4. Click **Continue** on the confirmation page.

The key is added and associated with the selected user.

Configuring the Browser to Use Certificate Authentication

You must configure the browser to use the SSL system key. Each browser supports Client Certificate Authentication differently; this is not documented in this guide.

Logging In to MFT to Test the Certificate Authentication

To log in to MFT to test the certificate authentication, perform the following steps:

Procedure

1. Connect to the MFT server.
2. If the browser has been configured correctly, it prompts you to select the certificate that you use to connect to the MFT server.
3. If the MFT has been configured correctly and the certificate has been associated with the user, the user is logged into the MFT server without requiring authentication.

Incoming FTPS Client Requests Certificate Authentication

FTPS certificate authentication for incoming requests is relatively simple to implement. All the steps included in the implementation of FTPS certificate authentication are performed on the MFT Admin pages. FTPS certificate authentication is supported in the following FTP modes:

- Explicit SSL using Clear Port
- Implicit SSL using SSL/TLS Port

Each FTPS client configures certificate authentication differently. How FTP Client certificate authentication is configured is not discussed in this guide.

Configuring the MFT Server to Allow or Require FTPS Certificate Authentication

You can configure users for FTPS certificate authentication in one of the following ways:

1. Applying global FTPS setting to all users on all MFT servers.
2. Applying user definition only to individual users.

Applying Global FTPS Setting to all Users on all MFT Servers

To apply global FTPS setting to all users on all MFT servers, perform the following steps:

Procedure

1. Go to **Configuration > System Configuration > FTP Settings**.
2. The **FTP Client Authentication Method** parameter allows four authentication values as listed in the following table. Any value other than **Password Only** requests a certificate from the FTP client.

Value	Description
Password Only	MFT does not request a client certificate and client certificate authentication fails. Password authentication is performed for incoming FTP requests.
Certificate Only	Client certificate authentication is supported.
Certificate and Password	Both certificate and password authentication are required.
Certificate or Password	Authenticate FTP clients using a certificate or password.

Applying User Definition only to Individual Users

To apply user definition only to individual users, perform the following steps:

Procedure

1. Go to **Partners > Users > Manage Users**.
2. Select the user that you want to configure.
3. Click the **Authentication Options** tab.
4. Set the **FTP Client Authentication Method** parameter to **Certificate Only** or **Certificate and Password**.
5. Click **Update** to save the change.

Associating a Public Key with an MFT User

The user must extract the certificate component of the system key that they will be using to connect to the MFT server.

To associate a public key with an MFT user, perform the following steps:

Procedure

1. Go to **Management > Protocol Keys > Public Key > Add Key**.
2. Configure the following parameters described in the table below:

Parameter	Instruction
Public Key Type	FTP public key.
Apply Key to	User.
Select user	Select the user that requires FTP certificate authentication.
Status	Enabled.
Description	Enter a unique description.
Enter the X.509	Paste the X.509 certificate to this box. The following is an example of an X.509 key:

Parameter	Instruction
	<pre>-----BEGIN CERTIFICATE----- base 64 encode data -----END CERTIFICATE-----</pre>

3. Click **Continue**.

A confirmation page is displayed.

4. Click **Continue** on the confirmation page.

A key is added and associated with the selected user.

Configuring the FTP Service to Use TLS

To configure the FTP service to use TLS, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > FTP Server > Configure FTP Server**.
2. Select the Internet Server instance of the FTP server that you want to configure.
3. Ensure the **IP Port** and the **TLS IP Port** are defined.



Note:

- **IP Port** points to the clear text port that is used by Explicit SSL
- **TLS IP Port** points to the Implicit SSL port.

4. Ensure the **FTP System Key** points to the correct FTP private key.

Restarting the FTP Server Service

When you change the FTP server authentication method, you must restart the FTP server service.

To restart the FTP server service, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > FTP Server > FTP Server Status**.
2. Select the Internet Server instance where you want to restart the FTP server service.
3. To get the current status of the FTP server service, press the **Status** button.
4. To stop the FTP server service, press the **Stop** button.
5. To start the FTP server service, press the **Start** button.

Using the FTP Client to Connect to MFT Internet Server

To use the FTP client to connect to the MFT Internet Server, perform the following steps:

Procedure

1. Ensure the FTP client is configured to use the private key associated with the public key configured for the incoming user.
2. Ensure the FTP client SSL mode is configured to the correct port.
 - Explicit SSL goes to the Clear Text Port (typically 21).
 - Implicit SSL goes to the SSL Port (typically 990).
3. Connect to the Internet Server.

i Note: The connection is made without requesting password authentication.

Incoming SSH or SFTP Client Requests Key or Certificate Authentication

SSH key, SFTP key, SSH certificate, and SFTP certificate authentication for incoming requests is relatively simple to implement. All the steps included in the implementation of SFTP key or certificate authentication are performed in the MFT administrator pages.

i Note: Most SFTP clients and servers support key authentication and not certificate authentication. An SSH key is a subset of an X.509 certificate. MFT supports both SSH key and certificate authentication. The example below explains how key authentication works. From an MFT perspective, SSH certificate authentication follows the same procedures as SSH key authentication.

Since the SFTP protocol is so frequently used, this guide explains how to create an SFTP keypair in a Linux environment. Perform the following procedures to create an SFTP keypair.

- Execute the `ssh-keygen` utility to create a keypair.
- Follow the instructions in the `ssh-keygen` utility to create the public or private keypair.
- The system key and public key are created in the directory that you defined. Let us assume that the following keys were created:

```
/home/user1/.ssh/sshkeyauth private key
```

```
/home/user1/.ssh/sshkeyauth.pub public key
```

The private key is used in the SCP or SFTP command; it defines the system key that is used when connecting to MFT.

The public key is sent to the MFT administrator. The MFT administrator associates the public key with an MFT user ID.

Configuring the MFT Server to Allow or Require SFTP Key Authentication

There are two ways to configure users for SFTP key authentication:

1. Applying global SSH or SFTP setting to all users on all MFT servers.
2. Applying user definition only to individual users.

Applying Global SSH or SFTP Setting to all Users on all MFT Servers

To apply global SSH or SFTP setting to all users on all MFT servers, perform the following steps:

Procedure

1. Go to **Configuration > System Configuration > Global SSH Settings**.
2. The **SSH Client Authentication Method** parameter allows four authentication values as listed in the following table. Any value other than **Password Only** requests a certificate from the SSH client.

Value	Description
Password Only	MFT does not request a client certificate and client certificate authentication fails. Password authentication is performed for incoming SSH/SFTP requests.
Certificate Only	Client certificate authentication is supported.
Certificate and Password	Both certificate and password authentication are required.
Certificate or Password	Authenticate SSH clients using a certificate or password.

Applying User Definition only to Individual Users

To apply user definition only to individual users, perform the following steps:

Procedure

1. Go to **Partners > Users > Manage Users**.
2. Select the user that you want to configure.
3. Click the **Authentication Options** tab.
4. Set the **SSH Client Authentication Method** parameter to **Key/Certificate Only** or **Key/Certificate and Password**.
5. Click **Update** to save the change.

Associating a Public Key with an MFT User

The SFTP client user must send the public key created by the `ssh-keygen` utility to the MFT administrator. This key must be added to the MFT system and associated with the MFT user.

To associate a public key with an MFT user, perform the following steps:

Procedure

1. Go to **Management > Protocol Keys > Public Key > Add Key**.
2. Configure the following parameters described in the table below:

Parameter	Instruction
Public Key Type	SSH public key.
Apply Key to	User.
Select user	Select the user that requires FTP certificate authentication.
Status	Enabled.
Description	Enter a unique description.
Enter the X.509	Paste the SSH key certificate to this box. There are two formats of SSH keys. MFT supports both formats. The following is an example of an SSH2 format: <pre>ssh-rsa Base64 encoded SSH Public Key user@computer.com</pre> The following is an example of an OpenSSH format: <pre>----- BEGIN SSH2 PUBLIC KEY ----- Comment: "ssh-rsa pubkey for 1.2.3.4"</pre>

Parameter	Instruction
	<pre> Base64 encoded SSH Public Key ----- END SSH2 PUBLIC KEY ----- </pre>

3. Click **Continue**.

A confirmation page is displayed.

4. Click **Continue** on the confirmation page.

The key is added and associated with the selected user.

Restarting the SSH or SFTP Server Service

When you change the SSH server authentication method, you must restart the SSH or SFTP server service.

To restart the SSH or SFTP server service, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > SSH Server > SSH Server Status**.
2. Select the Internet Server instance where you want to restart the SSH server service.
3. To get the current status of the SSH server service, press the **Status** button.
4. To stop the SSH server service, press the **Stop** button.
5. To start the SSH server service, press the **Start** button.

Logging In to MFT to Test the Certificate Authentication

To log in to MFT to test the certificate authentication, perform the following steps:

Procedure

1. Use the Linux SFTP client to connect to the MFT server

```
sftp -o port=2022 -o  
identityfile=/home/sshkeyauth/.ssh/sshkeyauth  
sshkeyauth@your.mft.server.com
```



Note: The **identityfile** parameter points to the private key created by the ssh-keygen utility.

2. If the SFTP client is configured correctly, the SFTP client passes the ssh key to the MFT SSH or SFTP server.
3. If MFT is configured correctly and the key is associated with the user, the user logs into MFT without requiring password authentication.

Incoming Platform Server Client Request Certificate

Platform Server certificate authentication for incoming requests is relatively simple to implement. All of the changes to implement Platform Server certificate authentication are performed in the MFT admin pages. Platform Server Certificate authentication is supported in the following Platform Server modes:

- Platform Server SSL
- Platform Server TLS Tunnel

Each Platform Server platform client configures certificate authentication differently. Since Platform Server is so frequently used, this topic includes general instructions on how to configure Platform Server for Unix to use certificate authentication.

Saving the MFT Platform Server Public Certificate in the Trusted Authority File

Before configuring Platform Server for Unix to use certificate authentication, save the MFT Platform Server public certificate in a Unix file called the Trusted Authority file by performing the following procedure.

Procedure

1. Go to **Administration > Protocol Keys > System Keys > Manage Keys**.
2. Select the Platform Server system key. The **Update System Key** page is displayed.
3. Click the **Public Key** box.
4. Copy the contents of the public key.
5. Use the Unix text editor, like vi, to create the TrustedAuthority file and paste the contents of the public key into this file.
6. Save the file.

This file is configured in the config.txt file.

Creating a Platform Server Keypair

To create a Platform Server keypair, perform the following steps:

Procedure

1. Run the `$CFR00T/util/sslutility.exe` utility to create a keypair.
2. In the **SSL Utilities** menu, enter 1 to generate a certificate request.
3. Enter the appropriate number for each prompt.
The private key file and certificate request file are created.
4. Send the **Certificate Request** file to the Certificate Authority.
The certificate file is configured in the `$CFR00T/config/config.txt` file. The private key file is configured in the `$CFR00T/config/config.txt` file.
5. Run the `$CFR00T/util/createPwd.exe` file to create the private key password file.
6. Enter the appropriate number for each prompt.
This creates the private key password file that is configured in the `$CFR00T/config/config.txt` file.
7. Edit the `$CFR00T/config/config.txt` file with the following changes in the server configuration entries for the Platform Server daemon and the client configuration entries for the Platform Server client:

Config Parameter	Instruction
CertificateFileName	Set to the certificate returned by the Certificate Authority
PrivateKeyFileName	Set to the Private Key file created by sslutility.exe.
PrivateKeyPwdFileName	Set to the Private Key password created by createPwd.exe.
TrustedAuthorityFile	Set to the Trusted Authority file that contains the MFT Platform Server public certificate.

8. Restart the Platform Server Responder by running the following commands:

```
cfstop -ssl  
cfstop -tunnel  
cfstart -ssl  
cfstart -tunnel
```

Configuring the MFT Server to Allow or Require Platform Server Authentication

There are two ways to configure users for Platform Server certificate authentication:

1. By applying global Platform Server setting to all users on all MFT servers.
2. By applying user definition only to individual users.

Applying Global SSH or SFTP Setting to all Users on all MFT Servers

To apply global SSH or SFTP setting to all users on all MFT servers, perform the following steps:

Procedure

1. Go to **Configuration > Configuration > Global Platform Server Settings**.
2. The **Platform Server Client Authentication Method** parameter allows five

authentication values as listed in the following table.

Value	Description
Any value other than Password Only	Requests MFT for a client certificate.
Password Only	MFT does not request a client certificate and client certificate authentication fails.
Certificate Only	Client certificate authentication is supported.
Certificate and Password	Both certificate and password authentication are required.
Certificate or Password	To authenticate Platform Server clients using a certificate or password.

Applying User Definition only to Individual Users

To apply user definition only to individual users, perform the following steps:

Procedure

1. Go to **Partners > Users > Manage Users**.
2. Select the user that you want to configure.
3. Click the **Authentication Options** tab.
4. Set the **Platform Server Client Authentication Method** parameter to **Certificate Only** or **Certificate and Password**.
5. Click **Update** to save the change.

Associating a Public Key with an MFT User

Import the certificate to the MFT system is associated with the MFT user.

To associate a public key with an MFT user, perform the following steps:

Procedure

1. Go to **Management > Protocol Keys > Public Key > Add Key**.
2. Configure the following parameters described in the table below.

Parameter	Instruction
Public Key Type	Platform Server public key.
Apply Key to	User.
Select user	Select the user that requires Platform Server certificate authentication.
Status	Enabled.
Description	Enter a unique description.
Enter the X.509	Paste the x.509 certificate to this box. The following is an example of an x.509 key: <pre>-----BEGIN CERTIFICATE----- base 64 encode data -----END CERTIFICATE-----</pre>

3. Click **Continue**.
A confirmation page is displayed.
4. Click **Continue** on the confirmation page.
A key is added and associated with the selected user.

Configuring the Platform Server Service to Use SSL

To configure the Platform Server service to use SSL, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > Platform Server > Configure Platform**

Server.

2. Select the Internet Server instance where you want to configure the Platform Server.
3. Ensure the **SSL IP Port** and **TLS Tunnel Port** are defined.
4. Ensure the **SSL System Key** points to the correct Platform Server private key.

Restarting the Platform Server Service

When you change the Platform Server authentication method, you must restart the Platform Server service.

To restart the Platform Server service, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > Platform Server > Platform Server Status**.
2. Select the Internet Server instance where you want to restart the Platform Server service.
3. To get the current status of the Platform Server service, press the **Status** button.
4. To stop the Platform Server service, press the **Stop** button.
5. To start the Platform Server service, press the **Start** button.

Configuring the Platform Server Client to Use Certificate Authentication

The Platform Server client is already configured to use SSL per the previous instructions. At this point, you might want to create a Platform Server node definition for the target MFT Platform Server.

To configure the Platform Server client to use certificate authentication, perform the following steps:

Procedure

1. Run the following program on a Unix Platform Server

```
cfnode
```

2. Configure the following parameters to define an SSL or TLS tunnel server.

Parameter	Description
IP address	Defines the DNS name or IP address of the MFT server.
Port	Defines the Port that MFT listens to for SSL and TLS tunnel requests.
TLS or SSL should be used	Defines whether the MFT port is for the SSL or TLS tunnel listener.

3. Save the node definition.

Executing a Platform Server Transfer to MFT to Test Certificate Authentication

The following command is an example of the Platform Server for Unix command to send a file:

```
cfsend n:mfttunnel lf:/tmp/file.txt rf:VirtualAlias/file.txt
```

Note: mfttunnel is the node definition created in the prior step.

Authenticating Outgoing Certificate to Target HTTPS Server

Outgoing HTTPS certificate authentication is performed when MFT acts as a client and connects to a target HTTPS server. It is the responsibility of the target HTTPS server to request a certificate from the MFT Internet Server. It is the responsibility of the Internet Server to provide the certificate to the HTTPS server.

There are two things that must be considered when MFT connects to a target server using **Certificate Authentication**. Both of these two considerations can be configured on the **Update Server** page.

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Select the server that you want to configure.
3. Ensure that the IP address or the fully-qualified IP name starts with `https://` and the IP port on the URL points to the HTTPS port.
4. Click **Retrieve HTTP public key** to associate the target server's public key with this server and add it to the database.
5. Click the **HTTP Options** tab and set the **HTTPS System Key** to the desired key.
6. Click **Update** to save the server definition.

When any client logs in and selects a transfer definition that points to this server definition, MFT connects to the target HTTPS server. If the server requests an HTTPS key, MFT authenticates using the configured HTTPS private key.

Authenticating Outgoing Certificate to Target FTPS Server

Outgoing FTPS certificate authentication is performed when MFT acts as a client and connects to a target FTPS server. It is the responsibility of the target FTPS server to request a certificate from MFT Internet Server. It is the responsibility of the Internet Server to provide the certificate to the FTPS server.

There are two things that must be considered when MFT connects to a target server using **Certificate Authentication**. Both of these two considerations can be configured on the **Update Server** page.

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Select the server that you want to configure.
3. Ensure that the IP port on the URL points to the FTPS port for the proper FTP mode.

IP port	FTPS port
Explicit SSL	clear text port, typically 21.
Implicit SSL	SSL, typically 990.

4. Ensure the **FTP Options** parameters are configured correctly:

Parameter	Description
Connection Security Type	Should be configured for Explicit SSL or Implicit SSL and must match the configured port SSL mode.
FTP System Key	Set to the desired key.

5. Click **Retrieve FTP public key** to associate the public key of the target server with this server and add it to the database.
6. Click **Update** to save the server definition.

When any client logs in and selects a transfer definition that points to this server definition, MFT connects to the target FTPS server. If the server requests an FTPS key, MFT authenticates using the configured FTPS private key.

Authenticating Outgoing Key or Certificate to Target SFTP Server

Outgoing SFTP key or certificate authentication is performed when MFT acts as a client and connects to a target SFTP server. It is the responsibility of the target SFTP server to request a key or certificate from MFT Internet Server. It is the responsibility of the Internet Server to provide the key or certificate to the SFTP server.

There are a few things that must be considered when MFT connects to a target server using the key or certificate authentication. These considerations can be configured on the **Update Server** page.

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Select the server that you want to configure.
3. Ensure the **SSH Options** parameters are configured correctly:

Parameter	Description
Key or Certificate	Set to Key unless the target SFTP server requires certificate authentication.
SSH System Key	Set to the desired key.

4. Click **Retrieve SSH public key** to associate the public key of the target server with this server and add it to the database.
5. Click **Update** to save the server definition.

Note: Alternatively, the transfer definition can override the SSH system key.

Converting SSH2 Key to an OpenSSH Key

Now, you must configure the target SSH server to associate an SSH public key with a user. This is done through the `authorized_keys` file in the `".ssh"` directory for the target user.

The following is an example of an SSH2 key:

```
----- BEGIN SSH2 PUBLIC KEY -----
Comment: "email@acme.com"
. . . . .
----- END SSH2 PUBLIC KEY -----
```

The following is an example of an OpenSSH key:

```
ssh-rsa . . . . .
. . . . .
. . . . . ==
```

To configure the target SSH server to associate an SSH public key with a user, complete the following steps in a Linux machine.

Procedure

1. Go to **Management > Protocol Keys > System Keys > Manage Keys**.
2. Select the system key that is used for key authentication. MFT supports both SSH2 and OpenSSH keys but generates the keys in the SSH2 key format.
3. Copy the key displayed in the **Public Key** tab.
4. Use the SSH command to communicate to the Linux machine.
5. Save the copied key to a file, namely, ssh2key.txt.
6. Convert the SSH2 key to an openssh key using the following command:

```
ssh-keygen -i -f ./sshkey.txt
```

The ssh-keygen command will display the key in the openssh format.

7. Save the openssh output of the previous command and update the authorized_keys file. Use the following command to perform the task:

```
cd /home/targetuser/.ssh
```

8. Use a vi text editor and add the OpenSSH key to the end of the authorized_keys file.

When any client logs in and selects a transfer definition that points to this server definition, MFT connects to the target SSH or SFTP server. If the server requests an SSH key, MFT authenticates using the configured SSH private key.

Authenticating Outgoing Certificate to Target Platform Server

Outgoing Platform Server certificate authentication is performed when MFT acts as a client and connects to a target Platform Server. It is the responsibility of the target Platform Server to request a certificate from MFT Internet Server. The Internet Server provides the certificate to the Platform Server.

i Note: The following Platform Servers support Certificate Authentication for incoming requests:

- Platform Server for z/OS
- MFT Internet Server Platform Server service

There are a few things that must be considered when MFT connects to a target server using **Certificate Authentication**. These considerations can be configured on the **Update Server** page.

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Select the server that you want to configure.
3. Ensure that the IP port on the URL points to the Platform Server port for the proper Platform Server mode.

IP port	Platform Server port
Implicit SSL	Points to the SSL port.
TLS Tunnel Port	Points to the TLS tunnel port.

4. Ensure the **Platform Server Options** parameters are configured correctly:

IP port	Instruction
Connection Security Type	matches the IP port defined. Possible values are Implicit SSL and TLS Tunnel.
Platform Server System Key	Set to the desired key.

5. Click **Retrieve Platform Server public key** to associate the public key of the target server with this server, and add it to the database.
6. Click **Update** to save the server definition.

When a client logs in and selects a transfer definition that points to this server definition, MFT connects to the target Platform Server. If the server requests a Platform Server key, MFT authenticates using the configured Platform Server private key.

Creating and Using Alerts

Alerts allow you to perform an action based on predefined trigger criteria. Only MFT Command Center allows you to configure alerts. You can execute pre-configured alerts on MFT Internet Server but you cannot configure alerts. So, Command Center is required to use alerts. The following types of alerts are supported.

Alert	Description
Login alerts	Performs an action when a user logs in to a system.
Transfer event alerts	Perform an action based on the completion of a transfer.
Transfer non-event alerts	Perform an action when a transfer does not complete.

The following table lists the five actions that can be configured for each type of alert.

Alert	Description
Send an email	Send an email to one or more recipients.
SNMP Trap	Send a trap to an SMTP server.
Execute Command	Execute a command. The command can be a command executed locally or a command to execute on a target Platform Server.
Execute Java Class	Execute a Java class. The customer must create and compile the Java class. Sample Java classes are provided.
Write a JMS Message	Write a message to a JMS queue or topic.
Webhooks	Send an alert notification to one or multiple Webhooks.

Execution of Alerts

The following table lists where alerts are executed.

Alert	Executed Location
Login alerts	On the Internet Server or Command Center where the user logs in.
Transfer Non-Event Alerts	On the Command Center where the scheduler is executing. If the MFT scheduler is configured to run with multiple Command Centers in Active/Active mode, the Alert can execute on either of the Command Center instances.
Transfer Event Alerts	There are two type Transfer Alerts : • Internet Transfer Alerts execute on the Internet Server instance where the Internet Transfer is executed. • Platform Transfer Alerts execute on the Command Center instance where the Platform Server Transfer is collected.

All the alerts can be configured with the following information:

- Required parameters such as description and whether the alert is enabled or disabled.
- Trigger criteria that define the conditions when an alert executes (See the trigger criteria below for each alert type).
- Alert actions define the five alert actions that can execute (see above).

Checking if an Alert has Executed

To check the alerts that are executed, perform the following steps:

Procedure

1. Go to **Reports > Alert History > Search Alerts**.

You can set the **Selection Criteria** to return specific alerts.

Alert history records that match the selection criteria are displayed in the **Results** table.

2. Click the **Alert Audit Id** tab for more information about the alert that is executed.

Login Alerts

Login alerts execute on the Internet Server or Command Center Server where the user logs in.

Trigger conditions can be set on the following criteria:

- User ID or group name
- Client protocols
- MFT instances (or systems)

If all of the criteria match, then one or more of the alert actions are executed.

Transfer Non-Event Alerts

Transfer non-event alerts are executed when one or more transfers do not execute within a pre-defined time frame. Transfer non-event alerts are a little more complicated because you need to make definitions in the following two different places:

- Adding the alert through the **Transfers > Alert** pages
- Adding a job through the Scheduler (**Management → Scheduler**) to execute at a pre-defined time to check if the transfers have completed.

Adding a Transfer Non-Event Alert

Adding a transfer non-event alert is only half the work needed. The alert definition defines the alert trigger criteria and the actions to be taken but it does not define when the alert trigger criteria is checked. You must create a scheduler job to execute. The scheduler job is executed at a pre-defined time and checks if any transfers match the alert trigger criteria.

Trigger conditions can be set on the following criteria

- Platform Server or Internet Server transfer
- Send or Receive

- Target Server Name
- Client and Server file names
- Process Name and Transfer Description

Create a Scheduler Job Type: Non-Event Transfer Alert

To create a non-event transfer alert scheduler job type, perform the following steps:

Procedure

1. Go to **Management > Scheduler > Jobs > Add Job**.
2. Enter the required information in the **Job Name**, **Group Name**, and **Description** fields.
3. Select the **Non-Event Transfer alert** job type.
4. Click the **Non-Event Transfer Alert** tab.

The following table lists the fields that need to be defined.

Field	Definition
Alert Description	Used to define the transfer non-event alert definition that you want to use.
Check for successful completion for a defined number of hours, days, or weeks	This defines how many hours, days, or weeks before the current time, where we check if transfers match the trigger criteria. For the sake of this example, enter "1 hour".

5. Select the action that you want to execute. As an example, configure the **Send Email** action. This assumes that the SMTP server has been configured properly in the system configuration.
6. Click **Add** to add the transfer non-event alert definition.
7. Click the **Scheduling Information** tab.

8. Define when the alert should be scheduled.

Defining Alerts

There are many fields that define when the alert is scheduled. The Help page has more details about scheduling parameters. For this example, let us assume that you expect a transfer to run every weekday between 1 PM and 2 PM. Hence, you would want to verify that the transfer has run from Monday to Friday between 1 PM and 2 PM. To do so, perform the following steps:

Procedure

1. In the **Scheduling Information** tab, click **By Day**.
2. Select the days from Monday through Friday. Do not select **Saturday** and **Sunday**.
3. Set **Execute Job at** to 14:00.
4. Enter value for the **Range of Recurrence**. This defines when checking starts and checking ends.
5. Click **Add** to the scheduler job.

Result

At 2 PM, from Monday to Friday, a scheduler job is executed. It performs the following tasks:

- When checking for Internet Server transfers, it must scan the audits database table for transfers that match the trigger criteria that have completed successfully in the last hour.
- When checking for Platform Server transfers, it must execute an audit inquiry request to the defined Platform Server for transfers that match the trigger criteria that have completed successfully in the last hour.
- If a transfer match is found, the alert is not executed.
- If a transfer match is not found, the defined alert actions are executed.

Transfer Event Alerts

Transfer event alerts are executed when one or more transfers completes, either successfully or unsuccessfully.

Adding a Transfer Event Alert

Trigger conditions can be set on the following criteria:

- Platform Server or Internet Server transfer
- Platform Server Node Name
- Send (Upload) or Receive (Download)
- Transfer Status: Success or Failure
- Target Server Name
- Client and Server file names
- Process Name and Transfer Description
- Post-Processing Action
- Success or Failure
- Tags

Transfer Alert checking depends on whether the alert is for a Platform Server Transfer or an Internet Server transfer.

Internet Server Transfer

When an Internet Server transfer completes, Internet Server checks the **Alert Trigger** criteria when it is about to write the audit record. If the defined **Trigger Criteria** matches the completed transfer, the defined alert actions are executed.

Platform Server Transfer

Platform Server Alerts only execute when the Command Center Collector retrieves Platform Server transfers. As each Platform Server transfer is collected, Command Center checks the **Alert Trigger** criteria when it is about to write the audit record. If the defined **Trigger Criteria** matches the completed transfer, the defined alert actions are executed.

Collecting Platform Server Transfer Audit Records

Platform Server audit records are saved on each individual Platform Server. By using the Command Center, you can search the Platform Servers for completed transfers. When the Command Center collects audit records from Platform Servers, the following additional capabilities are supported:

- Collected Platform Server records are written to the AuditFTS database table.
- You can use the **Audits > Search Audits** options to search and report on collected Platform Server transfers.
- You can configure alerts on the completed Platform Server transfers.
- You can execute database reports on the Platform Server transfers.
- You can view dashboards on Platform Server transfers.

Before you configure the Command Center Collection capabilities, you must configure both the Platform Server and the Command Center.

Configuring Platform Server

Perform the following Platform Server configurations to ensure that Command Centers collect audit records:

- Creating node definitions for incoming Command Center requests
- Adding the user ID to the Unix cfadmin or cfbrowse group

Creating Node Definitions for Incoming Command Center Requests

To create a node definition for each Command Center instance, use the cfnode utility. You must enter the IP name or IP address of the Command Center instance. You must enter ALL in **Command Center Support**. This allows you to collect Command Center audit records. It allows Command Center users to perform administrative functions such as defining nodes and profiles. It also allows Command Center users to execute Platform Server transfers.

Adding the User ID to the Unix `cfadmin` and `cfbrowse` group

When Command Center connects to the Platform Server to collect records, it passes the credentials (user ID/password) to the Platform Server. The user ID must be added to either of two Unix groups:

Group	Description
<code>cfadmin</code>	Allows users to perform audit inquiry and to perform node and profile functions.
<code>cfbrowse</code>	Allows users to perform audit inquiry.

Note:

- The Platform Server does not need to be restarted after configuring it. The node and group membership changes are dynamically adjusted.
- These changes must be made on each Platform Server where transfer audit records need to be collected.

Configuring Command Center

Perform the following Command Center configurations to ensure that Command Centers collect audit records from one or more Platform Servers:

- Configure the Server Definition to turn on Collection
- Configure the Command Center Collection Service
- Start or Restart the Command Center Collection Service

Configuring Server Definition to Turn Collection On

To configure the server definition to turn on collection, perform the following steps:

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Select the server that you want to collect transfer audit records from.
3. Open the **Management Options** tab.
4. Set the options listed in the following table:

Option	Instruction
Manage Platform Server	Select the checkbox.
Collect Platform Server History	Set to Initiator, Responder, Or Both.
Collection Interval	For now, leave this as the default value: 10. If you want to collect records more frequently, you can lower this value.

5. Click **Update** to save the changes.

Result

The Command Center Server is ready for audit collections.

What to do next

Restart the Command Center Collection Service.

Configuring Command Center Collection Service

To configure the command center collection service, perform the following steps:

Procedure

1. Go to **Management > Command Center Services > Collection Service > Configure Collection Service**.

2. Ensure the following options are set:

Option	Instruction
Enabled	Set to Yes.
Collection Server Host Name	Set to the Command Center instance where the Collector should execute.
Default Collection Interval	Leave at the default value of 10. The server definitions override this value, if collections are needed at a different interval.

3. Click **Update** to save the configuration changes.

Result

When the Collection Server is restarted, the Collector retrieves Audit records from that Platform Server. You can view the Platform Server audit records through the **Reports > Audits > Search Audits** page.

If Search Audits does return any Platform Server audit records for this server, follow the instructions in the [Diagnosing Problems](#) section.

What to do next

You can view the Platform Server audit records through the **Reports > Audits > Search Audits** page.

You can search for audit records collected from a server through the **Search Audits > Search Criteria**.

Depending on how many days of audit records are being collected, it might take some time for today's audit records to be collected.

Start or Restart Command Center Collection Service

To start or restart the Command Center collection service, perform the following steps:

Procedure

1. Go to **Management > Command Center Services > Collection Service > Collection Service Status**.
2. Click **Stop Service** if the service is already executing.
3. Click **Start Service**.
4. Click **Service Status** to get the current collector status.

The screen should display the server definitions where transfer audit records are being collected.

Viewing Platform Server Audit Records

To view the already collected Platform Server audit records, perform the following steps:

Procedure

1. Go to **Reports > Audits > Search Audits**.
2. Open the **Search Criteria** box and set the following fields:

Field	Instruction
Audit Type	Set to Platform Server.
Server Name	Enter the name of server definition (optional).

3. Click **Search**.

Result

Command Center retrieves Platform Server Audit records from the database. A list of audit records collected from this server is displayed. Click **Audit Id** to get detailed information on a transfer.

Polling Platform Servers for Audit Record Inquiry

The procedures defined in this section also allow you to dynamically poll Platform Servers for completed transfers. This allows you to view completed Platform Server transfers for servers whose transfers are not being collected.

Procedure

1. Go to **Reports > Audits > Search Audits**.
2. Open the **Platform Server Manual Poll Criteria** box and set the following fields:

Field	Instruction
Server Name	Enter the name of server definition (optional).
Set the other parameters as needed. The default is to search for all transfers on the current day.	

3. Click **Search**.

Result

Command Center initiates a connection to the target Platform Server to retrieve audit records that match the "Platform Server Manual Poll Criteria". A list of audit records retrieved from this Platform Server is displayed. Click **Local Transaction Id** to get detailed information on a transfer.

Diagnosing Problems

The best way to diagnose problems with the Collection Service is to view the following file on the Command Center instance where the collector is executing:

```
<MFT-CC Install>/logs/audit/CollectorAudit-audit-MFT-app-yyyy-mm-dd.log
```

A new file is written every day; yyyy-mm-dd is the year, month and day. All collector requests for all servers are written to this file.

View this file and search for the server name where collection was enabled. One record is written to this file for each collection request.

- If a collection request fails, a descriptive error message is displayed.
- If a collection request is successful, a summary record is written indicating how many records were collected.

Managing Platform Servers

Platform Servers are peer-to-peer servers that allow you to transfer files between Platform Servers and Internet Servers.

Platform Servers can be configured on the individual Platform Server. MFT Command Center can also configure Platform Servers.

The following Platform Server components can be configured by Command Center:

Component	Description
Node Definitions	Defines the connectivity information for transfer partners.
Profiles	Defines credentials used when connecting to transfer partners.
Responder Profiles	Defines an alternative mechanism for authenticating incoming transfer requests.

Command Center can also perform these functions that are described in other sections in this guide:

Function	Description
Collect Transfers	Collects completed transfers from target Platform Servers.
Audit Poll	Searches for completed transfers on defined Platform Servers.
Execute Transfer	Initiates transfers on target Platform Servers
Active Transfers	Retrieves active transfers from target Platform Servers. There is also an option to cancel an active transfer.

The mechanism for configuring and updating Platform Servers is the same for nodes, profiles, and responder profiles. This guide provides detailed information about how to

update nodes and only general information about how to update profiles and responder profiles, since the procedures are similar.

In order to allow Command Center to support Platform Server Configuration, you must make the following changes to the configuration:

- Configuring Command Center to support Platform Server Configuration
- Configuring Platform Server Definitions

Configuring Command Center to Support Platform Server Configuration

Perform the following Command Center configuration functions to allow Command Centers to view or update Platform Server node definitions.

- Configure the Command Center server definition to turn on Platform Server management.
- Configure the necessary rights for the admin user.

Configuring Command Center to Manage Platform Server

You can configure the Command Center server definition to start Platform Server management by performing the following steps:

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Select the Platform Server that you want to configure.
3. Open the **Management Options** tab.
4. Ensure the **Manage Platform Server** checkbox is selected.
5. When all the changes have been made, click **Update** to save the changes.

Configuring Command Center Admin Rights

To configure the necessary rights for the admin user, perform the following steps:

Procedure

1. Go to **Partners > Users > Manage Users**.
2. Select the user that needs to configure Platform Server nodes.
3. Ensure the **FTAdminRight** is granted. This gives the user the admin right to view or update nodes, profiles, and responder profiles.
4. Click **Update** to save the changes.



Note: If the updated user is logged in, the user must log out and log in to enable the right.

Configuring Platform Server Definitions

You must make two configuration changes to Platform Servers to enable Command Center to configure Platform Server node definitions:

- Adding the user ID to the Unix `cfadmin` or `cftransfer` group
- Configuring Platform Server node definitions

Adding the User ID to the UNIX cftransfer Group

When Command Center connects to the Platform Server to perform Platform Server configuration, it passes credentials (user ID/password) to the Platform Server. The user ID must be added to the `cftransfer` UNIX groups.



Note: The Platform Server does not need to be restarted after making these changes. The node and group membership changes are dynamically adjusted.

Configuring Platform Server Node Definitions

To create node definitions for incoming Command Center requests, run the `cfnode` utility on the Platform Server command line. You must enter the IP name or IP address of the Command Center instance. You should enter `All` in **Command Center Support**.

This allows Command Center to perform administrative functions such as defining nodes and profiles. It also allows you to collect Command Center audit records and execute Platform Server transfers.

Result

Platform Server is now ready to perform third party transfers initiated by Command Center.

Performing Platform Server Node Configuration

Platform Server node definitions can be stored and retrieved in two locations:

- Node definitions stored in the MFT database.
- Node definitions defined on the individual Platform Servers.

Viewing, Adding, and Updating Node Definitions in the Database And on Platform Servers

To view, add, or update node definitions in the database and on Platform Servers, perform the following steps:

Procedure

1. Go to **Platform Server Management > Platform Server Nodes > Add Platform Server Node**.
2. Enter the required fields in the **Required Node Information** tab.
3. Enter the required information in the **Additional Node Information** tab.
4. To add or update definitions of the target Platform Servers, open the **Server List** tab and select the checkbox next to the server or servers that you want to update.
5. Click **Update Server** to add the node to all of the Platform Servers selected in the **Server List** tab.
6. Click **Add** if you want to add the node name to the database.

Adding from Existing Platform Node

To retrieve node definitions from the database or from Platform Servers, perform the following steps:

Procedure

1. Click **Add from Existing Platform Node**.

When you enter this page, a list of nodes defined in the database are displayed.

2. Click the **Get Nodes From Server** box.

A list of Managed Platform Servers is displayed.

3. Select a Platform Server.

A list of nodes defined on that Platform Server is displayed.

4. Once a list of nodes is displayed, click **Node Name**.

The **Add Node** page is displayed and fields are filled in from the selected node.

Managing and Updating Platform Server Nodes

To manage and update Platform Server nodes, perform the following steps:

Procedure

1. Go to **Platform Server Management > Platform Server Nodes > Manage Platform Server Nodes**. A list of nodes is displayed.
2. To delete database node definitions, select one or more boxes to the left of the node name and click the **Delete** icon. A delete request is sent to the selected Platform server to delete this node.
3. To display the **Update Platform Node** page, select a node name from the list of nodes.

Updating Platform Node Parameters

To update Platform Node parameters, you can perform the following steps.

Procedure

1. Update fields in the **Required Node Information** tab.
2. Update fields in the **Additional Node Information** tab.
3. To add or update definitions of the target Platform Servers, open the **Server List** tab and select the checkbox next to the server or servers that you want to update.
4. Click **Update Server** to add or update the node on all of the Platform Servers selected in the **Server List** tab.
5. Click **Update** if you want to update the node definition in the database.

Adding or Updating Platform Server Profiles

To add or update Platform Server profiles, go to **Platform Server Management > Platform Server Profiles > Manage Platform Server User Profiles**.

Adding or Updating Platform Server Responder Profiles

To add or update Platform Server responder profiles, go to **Platform Server Management > Platform Server Responder Profiles > Manage Platform Server Responder Profiles**.

Executing Platform Server Transfers

Platform Server provides multiple interfaces to initiate transfer-to-transfer partners:

- Command Line Interface
- DNI or pDNI event driven processing interface
- GUI interface (Windows) or panel interface (z/OS and IBMi)

File transfers can also be initiated by Command Center. This is sometimes called a third-party transfer. Command Center initiates a request to Platform Server A to transfer a file with Platform Server B. The flow of a Command Center initiated transfer is **Command Center > Platform Server Initiator > Platform Server Responder**

Or

Command Center > Platform Server Initiator > Internet Server Responder

There are multiple ways in which Command Center can initiate Platform Server transfers:

- Through the Command Center Admin GUI interface
- Through the REST call: /pstransfers/execute
- Through a platform transfer scheduler job
- Execute through the scheduler
- Execute through a Platform Server Send Command request
- Through the MFT BW Plug-in: Initiate Platform Transfer
- Through XML submitted directly to a JMS Queue that Command Center listens on for incoming requests
- Through the Command Center Platform Transfer Command Line Interface

This guide concentrates on defining and initiating transfer through the Command Center admin interface.

You can define the parameters for a Platform Server transfer using any of the following ways:

- By using a pre-defined Platform Server transfer definition.
- By defining all of the parameters required to perform a transfer. This is sometimes called ad-hoc transfer initiation.

Configuring Command Center and Platform Server

Perform the following actions to ensure that Command Center supports Platform Server configuration:

- Configuring Command Center to support executing Platform Server Transfers.
- Configuring Platform Server Node Definitions and Group membership.

Configuring Command Center to support executing Platform Server Transfers

You can configure Command Center to support executing Platform Server transfers by performing the following steps.

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Select the Platform Server that you want to configure.
3. Open the **Management Options** tab.
4. Select the **Manage Platform Server** checkbox.
5. When all changes have been made, click **Update** to save the changes.

Configuring Necessary Rights for Admin User

To configure the necessary rights for Admin users, perform the following steps:

Procedure

1. Go to **Partners > Users > Manage Users**.

2. Select the user that needs to execute Platform Server transfers.
3. Ensure the **FTTransferRight** is granted.

This right allows the user to execute Platform Server transfers.



Note: AdministratorRight does NOT allow you to execute Platform Server transfers.

4. Click **Update** to save the user change.



Note: If the updated user is currently logged in, they need to log out and log in to enable the right.

Result

Command Center has been configured to allow an admin to execute Platform Server transfers.

What to do next

Configure the Platform Server to accept Command Center initiated transfer requests.

Configuring Platform Server Definitions

You must make two configuration changes to Platform Servers to enable Command Center to configure Platform Server node definitions:

- Adding the user ID to the Unix cfadmin or cftransfer group
- Configuring Platform Server node definitions

Adding the User ID to the UNIX cftransfer Group

When Command Center connects to the Platform Server to perform Platform Server configuration, it passes credentials (user ID/password) to the Platform Server. The user ID must be added to the cftransfer UNIX groups.

i Note: The Platform Server does not need to be restarted after making these changes. The node and group membership changes are dynamically adjusted.

Configuring Platform Server Node Definitions

To create node definitions for incoming Command Center requests, run the `cfnode` utility on the Platform Server command line. You must enter the IP name or IP address of the Command Center instance. You should enter `All` in **Command Center Support**.

This allows Command Center to perform administrative functions such as defining nodes and profiles. It also allows you to collect Command Center audit records and execute Platform Server transfers.

Result

Platform Server is now ready to perform third party transfers initiated by Command Center.

Adding and Executing Platform Server Transfers

To add and execute Platform Server transfers, perform the following steps:

Procedure

1. Go to **Transfers > Platform Server Transfers > Add/Execute Platform Server Transfer**.
2. Enter the required transfer information listed in the table below.

Parameter	Description
Description	Add a description for the transfer.
Transfer Direction	Select Send File, Receive File, or Send Command.

Parameter	Description
Server Name	Select the Platform Server where the transfer is initiated.
Initiator File Name	Defines the initiator file name.
Responder Host Type	Defines whether the destination Platform Server is defined by a node definition or an IP address or IP name. Tip: We recommend using a node definition, but this requires that a node definition is created on the Platform Server initiator for the destination Platform Server.
Responder Host Name	Defines the node name or IP name of the destination node.
Responder Port Number	When Responder Host Type is set to IP Name, this defines the IP port of the destination Platform Server.
Responder File Name	Defines the Responder file name.
Department	Defines whether the Platform Transfer definition will be assigned to a department.

Credentials and Security Properties

Executing Platform Transfers requires two sets of credentials:

- Credentials sent by Command Center to the Initiating Platform Server.
- Credentials sent by the initiating Platform Server to the responder (destination) Platform Server.

Credentials sent by Command Center to the Initiating Platform Server

There are two ways to define the credentials sent by the Command Center to the Initiating Platform Server:

1. Define the credentials on the server definition in the **Partners > Server Credentials** tab. When defining credentials in the server definition, you do not need to define initiator credentials in the **Add/Execute Platform Server Transfer "Credentials and Security Properties"** tab.
2. Define the credentials in the **Add/Execute Platform Server Transfer "Credentials and Security Properties"** tab
 - a. Initiator User ID
 - b. Initiator Password/Confirm Initiator Password (password length upto 64 bytes)

Credentials sent by the Initiating Platform Server to the Responder (Destination) Platform Server

There are two ways to define the credentials sent by the Initiating Platform Server to the Responder (destination) Platform Server:

1. On the Platform Server, create a user profile for the initiating user and the destination node. When a transfer is initiated, the credentials in the Platform Server user profile will be added to the transfer request. When using Platform Server user profiles, you do not need to define responder credentials in the **Add/Execute Platform Server Transfer "Credentials and Security Properties"** tab.
2. Define the following credentials in the **Add/Execute Platform Server Transfer "Credentials and Security Properties"** tab:
 - a. Responder User ID
 - b. Responder Password/Confirm Responder Password (password length upto 64 bytes)

You can define additional parameters in the following tabs.

Tab	Description
Additional Transfer Properties	The WAIT field on this tab defines when the response is displayed: WAIT=YES (Waits for the transfer to complete) WAIT=NO (Waits only until the transfer request is accepted)
Email Notification	Here, you can define email recipients of successful and failed Platform Server. Emails are sent by Initiating Platform Server.

Tab	Description
Postprocessing Actions	Here, you can set custom actions or commands, which are run on Initiating and/or Responding Platform Servers upon transfer completion.
z/OS Properties	Here, you can set z/OS specific file transfer options.
UNIX Properties	Here, you can set UNIX specific transfer option. They are added to file transfer parameters by Initiating Platform Server.

Transfer Parameter Options

After you define the transfer parameters, you can perform either of the following options:

Option	Description
Add	Adds the defined transfer to the MFT database. You can use this Platform Server transfer definition to execute transfers.
Execute	Initiates a transfer request to the Platform Server. Depending on the Wait parameter setting, the control will be returned when the transfer completes (WAIT=YES) or when the transfer is accepted (WAIT=NO). A summary message that includes the Platform Server Transaction ID is also displayed.

The **Add/Execute Platform Transfer** also has a link that allows you to copy the parameters from an existing Platform Server transfer to this page. All parameters are copied except for initiator and responder passwords.

Updating and Executing Platform Transfer

To update and execute a Platform Transfer, perform the following steps:

Procedure

1. Go to **Transfers > Platform Server Transfers > Manage Platform Server Transfers**.

The **Results** table displays a list of defined Platform Server transfers. You can use the **Search Criteria** to filter the Platform Transfers displayed.

2. Click **Transfer ID** of an entry in the results table to display the **Update/Execute Platform Transfer** page.

This page is essentially the same as the **Add/Execute Platform Transfer** page. It allows you to change any of the parameters defined for this Platform Server Transfer.

3. When you have completed updating the parameter, you can select any of the following options:

Option	Description
Update	Update the Platform Server transfer in the database.
Execute	Execute the Platform Server transfer from the parameters on this page.
Execute from Database	Execute the Platform Server transfer from the parameters defined in the database.

Result

The Platform Server transfer is ready to be executed, either directly from the admin page or from the transfer saved to the database.

What to do next

The next step shows the various ways that a Platform Server Transfer can be initiated by Command Center.

Executing Platform Transfers

Platform Transfers can be executed in the following ways:

- Through REST calls
- Through the scheduler
- Through the Command Center

- Through the Platform Server Send Command Request
- Through MFT BusinessWorks Plug-in: Initiate Platform Transfer
- Through XML Submitted to JMS Queue

Through REST calls

The following REST calls allow you to perform Platform Server file transfers:

REST Call	Description
<code>/pstransfers/execute</code>	Allows you to perform an ad-hoc transfer. All transfer parameters are defined by the REST call parameters.
<code>/pstransfers/{transferId}/execute</code>	Allows you to perform a transfer using parameters in the pre-defined Platform Transfer. Transfer parameters can be overridden by REST call parameters.

The following REST call is the only REST call that indirectly allows you to execute a Platform Server transfer:

```
/jobs/jobName/{jobName}/groupName/{groupName}/execute
```

This REST call allows you to execute a scheduler job. When the scheduler job is defined as a Platform Transfer, then this REST call will allow you to execute a Platform Server transfer.

MFT supports Redoc. Redoc allows you to view the URL and the parameters defined for REST calls. Use the following link to view Redoc for Command Center REST calls:

```
https://<MFT-CC-Install>:8443/cfcc/public/docs/redoc_cc.html
```

Through Scheduler

Scheduler jobs can be configured with a **Job Type** of "Platform Transfer".

This REST call indirectly allows you to execute a Platform Server transfer:

```
/jobs/jobName/{jobName}/groupName/{groupName}/execute
```

This REST call allows you to execute a scheduler job. When the scheduler job is defined as a Platform Transfer, then this REST call will allow you to execute a Platform Server transfer.

The scheduler is described in more detail in the section titled "Using the MFT Scheduler".

Through Command Center Platform Transfer Command-Line Interface

Command Center includes a CLI (Command-Line Interface) that allows you to execute ad-hoc or predefined Platform Transfers. The CLI allows you to add Platform Transfers to a script. The Platform Transfer CLI is located here:

```
<MFT-Install>/distribution/PlatformTransfer
```

For more information on the Platform Transfer CLI, see *TIBCO® Managed File Transfer Command Center Utilities Guide*.

Through Platform Server Send Command Request

A Platform Server can initiate a request to Command Center to execute a job.

The Platform Server request must be a **Send** command. Use this format for the command:

```
'ExecuteJob JobName="SendPSFile',GroupName=DEFAULT'
```

You can also override parameters defined in the **Scheduler** job. The **Add Scheduler Job** help page has information about how Platform Servers can execute scheduler jobs.

Through TIBCO MFT BusinessWorks(BW) Plug-in: Initiate Platform Transfer

The MFT BW interface allows you to perform these functions:

- Request Platform Server transfer
- Wait for Platform Server transfer to complete

The MFT BW plug-in interface is not described in this guide.

Through XML Submitted directly To JMS Queue

When configuring the MFT JMS interface, you can define a JMS Queue that Command Center listens for incoming Platform Server or Internet Server transfer requests.

This is the same interface that BW uses to communicate with the Platform Server.

Sample XML is located in the following file:

```
<MFT-  
Install>/server/webapps/cfcc/example/JMS/TransferRequestPlatformServer.xml
```

FileShare, Mailbox, and Four Eyes

This section describes how FileShare, Mailbox, and Four Eyes are configured. Many of the configuration parameters are shared between FileShare, Mailbox, and Four Eyes.

MFT Internet Server provides four separate ways to upload and download files:

Internet Server Transfers

- Provides a granular way to transfer files. Users must be given access to transfer files.
- Multiple client protocols are supported.
- Server definitions define where files are located.
- Files can be uploaded to, or downloaded from any target server.
- PGP encryption and PGP signatures are supported for incoming and outgoing requests.
- Postprocessing actions can define commands to be executed when transfers complete.
- Alerts can perform actions when a transfer matches a trigger condition.
- MFT Internet Server transfers are discussed in the topic *Getting Internet Server transfers working*.

Mailbox

- Provides a secure way to send files to one or more users.
- Easy to implement; limited configuration is required.
- Files are stored in a repository defined to the MFT Internet Server.
- Emails are sent to the recipients with instructions on downloading the files.
- Data stored in the repository can be PGP-encrypted.
- Supports the concept of Full, Power, and Guest users.
- Senders are notified when recipients download files

- Supports browser clients only.

FileShare

- Provides a very basic directory and file sharing capability.
- Easy to implement; limited configuration is required.
- Files are stored in a repository defined to the MFT Internet Server.
- Data stored in the repository can be PGP-encrypted.
- Supports the concept of Full, Power, and Guest users.
- Browser clients are supported.
- Other clients can write data to, or read data from FileShare folders.

Four Eyes

- Secure checking and approval of files is enabled before sending the files.
- Files are stored in a repository separate from the mailbox.
- Data stored in the repository can be PGP-encrypted.
- Browser clients are supported.
- Other clients can write data with the Four Eyes principle.

Creating PGP System Key

To encrypt the repository using PGP, perform the following steps:

Procedure

1. Go to **Management > PGP Keys > System Keys > Create PGP Key**.
The **Create PGP System Key** page is displayed.
2. Enter the required information described in the table below.

Parameter	Description
Description	Enter a unique description.
Pass Phrase	Enter a pass phrase.
Confirm Pass Phrase	Enter the same pass phrase.
Expiration Date	Set the expiration date to 2050.
Key Size	Set to at least 2048.
Key Type	Set to DSA and ElGama1.
Hashing Algorithm	Set to SHA-256, SHA-384, or SHA-512.
Real name	Enter any data
Email Address	Enter an Email address.

3. When all the information is entered, click **Create Key** to create the PGP system key.

Result

Optional PGP encryption has been set up for the FileShare repository.

What to do next

Proceed to the next step to create a Repository Server Definition.

Creating Repository Server Definition

Mailbox, FileShare, and Four Eyes save all the files in a defined Server Definition.

**Important:**

- All Internet Server instances must have access to the data in the server definitions.
- FileShare and Mailbox use the same repository, whereas Four Eyes can be configured to use a separate repository server.

The following types of server definitions are supported.

Server Definition	Description
Local	Data is saved in any storage accessible by all Internet Server instances. When multiple Internet Server instances support Mailbox or FileShare, this storage should be in a NAS server or NFS Share.
Platform Server	Data is sent to a target Platform Server. When you have Internet Servers executing in the DMZ, the best practice is to use a Platform Server as the repository server. The Platform Server can be in the internal network and can accept requests from multiple Internet Server instances in the DMZ and internal network.
Amazon S3	Data is sent to an S3 storage accessible by all Internet Server instances.
Azure File	Data is sent to Azure File storage accessible by all Internet Server instances.
Google Cloud Storage	Data is sent to Google Cloud Storage accessible by all Internet Server instances.

! **Important:** It is extremely critical that these restrictions be followed:

- Do not delete a server definition used as a repository. Requests already initiated to this server will no longer work if this server definition is deleted.
- Do not change from using PGP to not using PGP, or from not using PGP to using PGP in a single server definition. If you want to change a server to use PGP or not use PGP, create a new server definition. Then, change the **File Share Configuration** to **Repository Server Name**.

To create a repository server definition, perform the following steps:

Procedure

1. Go to **Partners > Servers > Add Server**.
2. Set the following parameters:

Parameter	Instruction
Server Name	Set to a unique name that describes this as a repository server.
IP Address	Set to the IP address of the target Platform Server. If the server type is LOCAL, set this to any value.
IP Port	Set to the IP Port of the target Platform Server. If the server type is LOCAL, leave this blank
Server Type	Set to LOCAL or Platform Server.
Server Platform	Set to Windows or UNIX.

Creating Platform Server Definition

To create a Platform Server definition, perform the following steps:

Procedure

1. Click the **Server Credential** tab.

2. Enter the required information described in the table below:

Field	Description
Default User	Defines the Platform Server user ID.
Default Password	Defines the password for the platform Server user.
Confirm Password	Re-enter the password.

3. Click **Add**.

Encrypting Repository Data Using PGP

To encrypt the repository data using PGP, perform the following steps:

Procedure

1. Click the **PGP Information** tab.
2. Configure the following parameters listed in the table below:

Parameter	Description
PGP Enabled	Select this checkbox.
Private Key	Select the PGP Private Key defined in the prior step.
Encryption	Select this checkbox.
Encryption Algorithm	Set to AES-256.
Hashing Algorithm	Set to SHA-256, SHA-384, or SHA-512.
Compression Algorithm	Set to ZLIB if you want to compress repository data.

3. After entering the required information, click **Add** to create the server definition.

i Note: If a repository server has been used for FileShare or Mailbox requests, you **MUST NOT** change the server definition PGP options. This causes Mailbox or FileShare download request to fail with a PGP error. Instead, you should create a new server definition that has the desired PGP change.

Result

The FileShare Repository Server Definition setup is complete. You can use a similar process to set up the Four Eyes Repository.

What to do next

If PGP is enabled, proceed to the next step to associate the PGP key with the server definition.

If PGP is not enabled, proceed to step [Updating the Global Configuration](#).

Associating PGP Public Key with Server Definition

Perform this action if you are encrypting data in the repository by using PGP.

This action takes the PGP public key associated with the PGP private key and associates the key with the repository server definition. To perform this action, complete the following steps:

Procedure

1. Go to **Management > PGP Keys > System Keys > Manage PGP Keys**.
2. Click **Description**.
3. Select the PGP system key that you just created.
The **Update PGP System Key** page is displayed.
4. Open the **PGP Public Key** tab.
5. Copy the public key in the box to the clipboard. The following is a format of a PGP public key:

```

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: BCPG v1.64
. . . . . Public Key data . . . . .
-----END PGP PUBLIC KEY BLOCK-----

```

6. Go to **Management > PGP Keys > Public Keys > Add PGP Key**.

The **Add PGP Public Key** page is displayed.

7. Set the following parameters in the **Add PGP Public Key** page.

Parameter	Instruction
Apply Key to	Server
Select Server	Select the Repository Server
Status	Enabled
Set as Default Key	Yes

8. Paste the PGP public key copied in the prior section into the box at the bottom of this page.

9. Click Continue to add the key to the selected server.

The **Add PGP Public Key Confirmation** page is displayed.

10. Click **Continue** to add the public key.

Updating the Global Configuration

Mailbox, FileShare, and Four Eyes functions must be connected to an SMTP email server.

Mailbox uses emails in the following ways:

- When a user initiates a mailbox request, MFT sends emails to recipients notifying them that a file is available to download.
- When a recipient downloads a file, MFT optionally sends a notification to the sender that the file has been downloaded.

FileShare uses email when a user shares a folder with a user, an email is sent to the target user notifying them that a share has been initiated.

Four Eyes uses emails in the following ways:

- When a user initiates a Four Eyes request, MFT sends an email to the approver notifying them of the approval request.
- When an approver approves a request, MFT sends an email to the recipients notifying them that a file is available for download.
- Optionally, when an approver approves a request, MFT sends a notification to the sender that the request has been approved.
- Optionally, when a recipient downloads a file, MFT sends a notification to the sender that the file has been downloaded.

Emails are also used when a user forgets their password, a password reset email is sent to their email address.

To update the Global Configuration, perform the following steps:

Procedure

1. Go to **Configuration > System Configuration**.
2. On the **Global Settings** tab, enter the email server information described in the table below:

Parameter	Description
Email Host Name	Defines the DNS name of the SMTP server.
Email Host Port	Defines the IP port of the SMTP server.
Email Admin User Id	If required, enter the email admin user ID.
Email Admin Password	If required, enter the email admin password.
SMTP TLS	Defines if the SMTP server is using TLS.
Trust SMTP SSL Certificates	Defines if all SMTP TLS certificates can be trusted.

3. Click **Update** to save the configuration changes.

4. Open the local or remote settings for the Internet Server instance.
5. Update the Email URL parameter.

This parameter defines the URL that is written in emails. For example, the email URL is used when a mailbox notification email is sent to recipients. It defines the URL that is defined in the email to download the mailbox attachments.

6. Click **Update** to save the configuration changes.

Updating FileShare Mailbox Four Eyes Configuration

Most configuration parameters for Mailbox, FileShare, and Four Eyes are on the **FileShare Mailbox Four Eyes Configuration** page. This guide describes the required parameters to get Mailbox, FileShare, and Four Eyes working. Additional parameters are documented in the online help.

The following table lists the repository setting parameters.

Parameter	Instruction
Repository Server Name	Select the server defined for the FileShare/Mailbox repository. Note: You can change the Repository Server Name even after making Mailbox and FileShare requests. Do not delete the server or convert it to use PGP or disable PGP processing, as this could corrupt existing definitions. To enable or disable PGP, create a new server definition and update the Repository Server Name setting.
File Share	Enabled.
Mailbox	Enabled.
FileShare Sender Email Address	Defines how the sender email address is created for FileShare requests. You can select from the three options provided below:

Parameter	Instruction
	• User Definition: Uses the email address from the User that initiated the request. • Use default: Uses the Email Address defined by the Default FileShare Sender Email Address parameter. • Use default with Full Name: Uses the Email Address defined by the Default FileShare Sender Email Address parameter, prefixed with the Full Name from the User that initiated the request.
Default FileShare Sender Email Address	Defines the Sender Email address used when FileShare Sender Email Address is set to Use default or Use default with Full Name . This field is ignored when FileShare Sender Email Address is set to User Definition .
Mailbox Sender Email Address	Defines how the Sender Email address is created for Mailbox requests. You can select from the three options provided below: • User Definition: Uses the Email Address from the User that initiated the request. • Use default: Uses the Email Address defined by the Default Mailbox Sender Email Address parameter. • Use default with Full Name: Uses the Email Address defined by the Default Mailbox Sender Email Address parameter, prefixed with the Full Name from the User that initiated the request.
Default Mailbox Sender Email Address	Defines the Sender Email address used when Mailbox Sender Email Address is set to either the Use default option or Use default with Full Name option. This field is ignored when Mailbox Sender Email Address is set to User Definition .
Settings for Users Created by Senders	
Internal Email Domains	Defines which users are considered as internal (full and

Parameter	Instruction
	power) users and which users are configured as guest users. You can enter multiple email domains in this box, separated by a semicolon. For example, assume you set this parameter to: <code>acme.com</code> and <code>acmeparent.com</code>. When a new user is created as a result of a Mailbox or FileShare request if the user's email domain is <code>acme.com</code> or <code>acmeparent.com</code>, the user is considered a full user. If the user's email domain is not <code>acme.com</code> and <code>acmeparent.com</code>, the user is considered a guest user.
Create Users in External Email Domains	Defines whether users can create external users. Enabled: Allows full or power users to create guest users. Disabled: Guest users cannot be created.
Initial User Status	Enabled: Created users are enabled when created. Disabled: Created users are disabled when created
Common Settings	
Maximum Expiration	Defines the maximum number of days before a Mailbox or Four Eyes request expires.
Maximum Number of Recipients	Defines the maximum number of recipients for a Mailbox or Four Eyes request.
Restrict Attachment Action	Defines whether the attachment types are restricted.
Restrict Attachment Types	Defines the file name suffixes that cannot be uploaded when attachments are restricted.
Maximum File Size	Defined the maximum file size in MB.
Four Eyes Settings	

Parameter	Instruction
Four Eyes	Enabled
Four Eyes Repository Server Name	Selects the server defined for the Four Eyes repository. Note: You can change the Repository Server Name even after making Mailbox and FileShare requests. Do not delete the server or convert it to use PGP or disable PGP processing, as this could corrupt existing definitions. To enable or disable PGP, create a new server definition and update the Repository Server Name setting.
Four Eyes Sender Email Address	Defines how the Sender Email address is created for Four Eyes requests. You can select from the following three options: • User Definition: Uses the Email Address from the User that initiated the request. • Use default: Uses the Email Address defined by the Default Four Eyes Sender Email Address parameter. • Use default with Full Name: Uses the Email Address defined by the Default Four Eyes Sender Email Address parameter, prefixed with the Full Name from the User that initiated the request.
Default Four Eyes Sender Email Address	Defines the Sender Email address used when Four Eyes Sender Email Address is set to Use default or Use default with Full Name. This field is ignored when Four Eyes Sender Email Address is set to User Definition.

Updating User Definition

In order for users to use the Mailbox or FileShare features, a few user definitions must be made. To update user definitions, perform the following steps:

Procedure

1. Go to **Partners > Users > Manager Users**.
2. Use the **Search Criteria** to select the users that you want to support Mailbox and FileShare.
3. Select the user ID that you want to update.
4. Update the required user information listed in the following table.

Type	Parameter	Description
Usage	FileShare User capabilities	Allows FileShare or Mailbox.
	Non-File Share capabilities	Does not allow FileShare of mailbox.
	Mailbox User	Allows Mailbox capabilities.
User Type	Guest user	Can share or send files with full or power.
	User	Cannot create any users.
	Full User	Can create guests but cannot create internal users.
	Power	Can create guests or full users.
Email Address	Enter an Email Address.	Mailbox and FileShare users must have an email address defined.

5. Update the **Rights and Groups** page.
6. Ensure the **TransferRight** is assigned to the user.
7. When all changes have been made, click the **Update** button to save the changes.

Result

FileShare and Mailbox are ready for the user to use.

What to do next

Proceed to the next step to test the Mailbox feature.

Configuring Four Eyes Policy

To use Four Eyes, you must define a Four Eyes Policy. Ensure that you have at least two groups defined before adding a policy.

To add a Four Eyes Policy, perform the following steps:

Before you begin

You must enable Four Eyes.

Procedure

1. Go to **Transfers > Four Eyes Policies > Add Policy**.
2. From the dropdown list, provide all the necessary details, including the approver group and recipient group.
3. Click **Add** at the upper-right corner.
See online help for additional parameters.

Updating Transfer Page

To enable Four Eyes, you must add a Four Eyes download transfer definition. You can also do this on an existing Transfer Definition.

Procedure

1. Go to **Transfer > Internet Server Transfers > Add Transfer**.
2. Add the required fields.
3. Go to **Four Eyes Properties** and add a policy.



Important: You can add Four Eyes to either Download Transfer Definition or Upload Transfer Definition with a Four Eyes server.

Test the Mailbox Browser

To test the mailbox browser, perform the following steps:

Procedure

1. Go to the following URL:

```
https://your.dns.name.com:7443/mailbox
```



Note: You must change the DNS name and port to match your installation requirements.

2. There are three mailbox tabs. Select the required tab.

Tab	Description
Send Files	Allows you to send files to one or more recipients.
Inbox	Displays Mailbox requests that other users have sent to you. This tab is displayed when you enter the mailbox pages.
Sent Items	Displays Mailbox requests that you have sent to other users.

Sending Attachment to Another User

To send an attachment to another user, perform the following steps:

Procedure

1. Click the **Send Files** tab.
2. Enter the following parameters.

Parameter	Description
To	Enter the email address of the recipients.
Subject	Enter the subject of the request.
Attach	Click this button and select a file to send.

3. Enter any text in the text box at the bottom of this page.
4. When all the parameters are defined, click the **Send** button.
A confirmation page might be displayed.
5. If the confirmation page is displayed, select the recipient.

Result

The files are uploaded to MFT Internet Server and saved in the repository.

An email is sent to the recipients with instructions on how to download the attachments.

When the request is successful, you see the request in your **Send Items** tab, and the recipients should see the file in the **Inbox** tab.

i Note: Each time a Mailbox file is uploaded or downloaded, an Internet Server audit record is created. You can view the audit record through the **Reports > Audits > Search Audits** page.

Testing the FileShare Browser

To test the FileShare browser, perform the following steps:

Procedure

1. Go to the following URL:

```
https://your.dns.name.com:7443/fs
```

When you enter the FileShare page for the first time, no folders or files are listed.

i Note: You must change the DNS name and port to match your installation requirements.

2. To create a folder, click the create folder icon (folder with a small green plus sign).
3. Enter the folder name and a description for the folder.

The **Confidential** checkbox is a reminder that this folder might contain confidential information. It does not restrict any FileShare functionality.

4. Click **Create** to create the folder.

5. Double-click the folder created in step 4.

This folder becomes the current directory. You can now start Windows Explorer and drag one or more files to the folder.

6. As you drag files to the folder, they are uploaded to the MFT repository and the files are displayed in the current directory.

7. Click the **Share Folder** icon.

You can now select one or more users to share this folder with.

8. When you share a folder with a collaborator, you can assign one of the following roles.

Role	Description
Owner	Collaborator can view and update files and sub-folders. Owners can also share this folder.
Editor	Collaborator can view and update files and sub-folders.
Viewer	Collaborator can view files and sub-folders.

9. When you press **Share**, a request is sent to the server to share the folder.

An email is sent to the Collaborator indicating that a share request was made. When the Collaborator logs in, they can accept the share request and the folder will be listed on their FileShare home page.

Note:

- Each time a FileShare file is uploaded or downloaded, an Internet Server audit record is created. You can view the audit record through the **Reports > Audits > Search Audits** page.
- Each time a FileShare folder is shared, an event record is created. You can view the event records through the **Diagnostics > Events > Search Events** page.

Testing the Four Eyes Browser

To test the Four Eyes browser, perform the following steps:

Procedure

1. Go to the following URL:

```
https://your.dns.name.com:7443/foureyes
```

2. There are four tabs in Four Eyes, each shown based on the user role.

Tab	Description
Send Files	Shown when a user is a Sender. Allows the sender to send files to Approvers or optionally to the Recipient.
Sent Items	Shown when a user is a Sender. Displays Four Eyes requests that the sender has sent to other users.
Approval	Shown when a user is an Approver. Shows Four Eyes requests submitted by a sender for approval.
Inbox	Shown when a user is a Recipient. Shows Four Eyes requests either after approval by the Approver or directly from the sender.

Sending Attachment to Approver

To send an attachment to the approver, perform the following steps:

Procedure

1. Navigate to **Send Files**.
2. Enter the following information.

Parameter	Description
Approver	Click this button and select the approvers.
Recipient	Click this button and select the recipients.
Subject	Enter the subject of the request.
Attachment	Click this button to select the files to send.

3. Enter the approver and recipient text in the text boxes at the bottom of the page.
4. When all parameters are defined, click **Send**.

Result

- The file uploads to MFT Internet Server and is saved in the repository.
- An email with instructions is sent to the approver.
- When the approver approves the request, an email is sent to the recipient with instructions on how to download the attachments.
- When the request is successful, the sender can view the file in **Sent Items**, and the recipients can see the file in **Inbox**.

Testing Automated Four Eyes Requests

To test Automated Four Eyes requests, perform the following steps:

Procedure

1. Create a Four Eyes Server definition from where the files can be attached. For more information, see [Creating a Server Definition](#).
2. Create a Transfer definition to locate the files on the server and define the Four Eyes properties in the following manner:
 - a. When you select a Four Eyes server in the transfer definition, the transfer direction is **Upload**.
 - b. Under the **Four Eyes Properties** tab, select a policy where you can choose users from the approver group and recipient group.
 - c. Define an email subject sent to the recipient and the message texts for both approvers and recipients.
3. After adding the necessary fields, click **Add** at the upper-right corner.

Perform an upload transfer using the Defined Transfer definition from any MFT client (HTTP, FTP, SSH, and Platform Server).



Note:

For automated Four Eyes transfers, you can override approvers or recipients when initiating the transfer from the command line. To enable this, select **Allow Approvers to be overridden** and **Allow Recipients to be overridden** checkboxes under the **Four Eyes Properties** tab of the transfer definition.

For example,

```
cfsend ip:127.0.0.1 port:46465 uid:test pwd:test lf:file1.txt
rf: "/upload_
alias/file1.txt,&Approvers=app1,app2&Recipients=rec2,rec1"
```

For more information about these parameters, see the admin help pages.

Result

- The file uploads to MFT Internet Server and is saved in the repository.
- An email including instructions is sent to the approver.
- When the approver approves the request, an email is sent to the recipient with instructions on how to download the attachments.
- When the request is successful, the sender sees the file in **Sent Items**, and the recipients sees the file in the **Inbox**.

Configure and Start the AS2 Transfers

AS2 is an open protocol standard designed to transmit data securely over the Internet. AS2 is a send-only file transfer protocol. You cannot initiate a receive on AS2. AS2 clients can initiate AS2 transfers to send files to MFT. MFT clients can initiate transfers to send files to AS2 servers.

Before you begin

Enter the following information from the AS2 partner:

Parameter	Description
AS2 PartnerID	A string that identifies the partner.
Encryption Public Certificate	The certificate used to encrypt data sent to the AS2 partner.
Signing Public Certificate	The certificate used to verify the AS2 data signature.
HTTPS Public Certificate	When using HTTPS, this is the certificate of the target AS2 server.
AS2 Server URL	The IP address used when connecting to target AS2 servers.
Credentials	If required by the partner AS2 server.

To configure MFT for incoming and outgoing AS2 transfers, perform the following steps:

Procedure

1. [Configure the MFT AS2 transfer server.](#)
2. [Create an AS2 system Key for decrypting data.](#)

Alternatively, you can create an AS2 system key for signing data or you can use the AS2 decryption key to sign data. You can also create an AS2 system key that is used

when HTTPS certificate authentication is required for outgoing requests to a partner AS2 server.

3. [Create a user definition for incoming AS2 requests.](#)
4. [Create a server definition for incoming and outgoing AS2 requests.](#)
5. [Create transfer definitions for incoming and outgoing AS2 requests.](#)
6. [Start the AS2 service.](#)
7. [Send information about the MFT AS2 environment to the AS2 transfer partner.](#)
8. [Initiate a transfer to an AS2 partner.](#)

Configuring MFT AS2 Transfer Server

To configure the MFT AS2 transfer server, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > AS2 Server > Configure AS2 Server**.
2. Select the Internet Server instance that you want to configure.
3. Configure the following information:

Parameter	Instruction
Enabled	Set to Yes.
Receive URL	Set the IP name, port, and context (cfcc). Leave the other URL parameters as they are.
ASync Response URL	Set the IP name, port, and context (cfcc). Leave the other URL parameters as they are.
Local AS2 ID	Enter the AS2 ID of the MFT AS2 server.
Proxy Information	Enter proxy information if required.

4. When you have completed configuring the fields, click **Update** to save your changes.

Creating AS2 System Key to Decrypt Data

To create an AS2 system key for decrypting data, perform the following steps:

Procedure

1. Go to **Management > Protocol Keys > System Keys > Create Key**.

The **Create System Key** page is displayed.

2. Configure the following information:

Parameter	Instruction
System Key Type	AS2 system key.
Description	Enter a unique description.
Password	Enter and confirm the system key password.
Expiration Date	Set this according to your company's standards.
Key Size	Set to 2048 or higher.
Signing Algorithm	Set to SHA-256, SHA-384, or SHA-512.
Set as the Default key	Select this checkbox.
Common Name	Set to IP the name of your AS2 server.

3. After entering the information, click **Create Key** to create the AS2 system key.

AS2 requires a system key to decrypt and sign data. You can use the same key for both decryption and signing. Alternatively, you can create a separate key for signing AS2 data. If you want to create a separate key for signing data, follow the procedure for creating an AS2 decryption key. Change the description so that it is clear that this is a signing key.

Alternatively, you can create an AS2 system key that is used when certificate authentication is required for outgoing AS2 transfers to AS2 servers. This is somewhat rare. You can use

the same key that is used for decryption or signing. You can also create a separate key for HTTPS. If you want to create a separate key for HTTPS, follow the procedure for creating an AS2 decryption key. Change the description so that it is clear that this is an HTTPS key.

Creating User Definition for Incoming AS2 Requests

This user definition is used for incoming AS2 transfer requests. The server definition defines the user when transfers are received from an AS2 client. This user is defined on the transfer definitions to point to a target server such as an SFTP, or Platform Server.

 **Note:** An separate AS2 user ID must be created for each incoming partner AS2 server.

Procedure

1. Go to **Partners > Users > Add User**.

The **Add User** page is displayed.

2. Enter the required information listed in the table below.

Field	Description
User Id	Enter a unique user ID.
Full Name	Enter any fill name for this user.
Password	Enter any characters. Generally speaking, AS2 transfers do not perform password validation. If AS2 transfers require password validation (this is rare), enter the password here.
Usage	Non-file share user.

3. Ensure the **TransferRight** is assigned to the user.
4. After entering these fields, click **Add** to create the user.

You can also configure optional user properties.

Parameter	Description
Client Protocols Allowed	Set to AS2.
Change Password at Next Login	Clear this checkbox.

Server Definition for Incoming and Outgoing AS2 Requests

For all other protocols, server definitions are only used for outgoing requests. For AS2 however, server definitions are used for incoming and outgoing AS2 requests.

Incoming Requests

- The transfer partner initiates a connection request to MFT and specifies its partner ID.
- MFT searches the server definitions for a matching partner ID.
- When it finds the matching partner ID, it gets the user ID from the server definition.
- MFT then searches for transfer definitions for the user ID defined in the server definition
- The transfer definition points to a target server such as SFTP or Platform Server. This is the server where the incoming AS2 data is saved.

Outgoing Requests

- A client (SFTP, Platform Server, and HTTPS) logs in and initiates a transfer request to MFT.
- The client specifies a remote file name whose first parameter is the virtual alias.
- MFT searches the transfer definitions for that user and upload definition with a match on the virtual alias.
- The transfer definition selected points to an AS2 server definition.
- The data received from the transfer client is sent to the AS2 server.

Creating Server Definition for Incoming and Outgoing AS2 Requests

To create a server definition for incoming and outgoing AS2 requests, perform the following steps:

Procedure

1. Go to **Partners > Servers > Add Server**.
2. On the **Add Server** page, enter the following required information:

Parameter	Description
Server Name	Enter a unique server name.
IP Address	Enter the AS2 URL provided by the AS2 transfer partner.
IP Port	This is ignored for AS2 servers.
Server Type	Set to AS2.
Server Type	Set to UNIX.

 **Note:** Server credentials are rarely required for AS2 transfers. The public or private key pairs overrides the necessary security without credentials.

3. If server credentials are required by the Partner AS2 Server, enter them in the **Server Credentials** tab.

Parameter	Description
Local AS2 ID	Enter the AS2 ID for your system. If not defined, the AS2 ID defined in the Configure AS2 Server page is used.
Remote AS2 ID	Enter the AS2 ID that the Partner AS2 client or server sent to you. For

Parameter	Description
	incoming requests, this must match the Partner ID that the partner sends in the AS2 transfer flow.
UserId for incoming requests	Select the user that you created in the prior step. This user will be used for incoming AS2 requests.
AS2 Options: System Keys	
HTTPS System Key	Enter the AS2 key used for HTTPS certificate authentication to AS2 servers.
Encryption System Key	Enter the system key used for decrypting data.
Signing System Key	Enter the system key used for signing data.
Partner Public Certificates	
Encryption Public Certificate	Paste the encryption public key provided by the AS2 partner.
Signing Public Certificate	Paste the signing public key provided by the AS2 partner.
HTTPS Public Certificate	Paste the HTTPS public key provided by the AS2 partner.
Outgoing Parameters	
MDN Receipt	Defines the type of MDN receipt that you require:

Parameter	Description								
	<table><tr><th>MDN Receipt</th><th>Description</th></tr><tr><td>Sync</td><td>Synchronous MDN Receipt</td></tr><tr><td>Async</td><td>Asynchronous MDN Receipt</td></tr><tr><td>None</td><td>no MDN Receipt is required</td></tr></table>	MDN Receipt	Description	Sync	Synchronous MDN Receipt	Async	Asynchronous MDN Receipt	None	no MDN Receipt is required
	MDN Receipt	Description							
	Sync	Synchronous MDN Receipt							
	Async	Asynchronous MDN Receipt							
None	no MDN Receipt is required								
Encryption Algorithm	Defines parameters used when initiating AS2 transfers to a partner AS2 server.								

Result

The server definition has been configured for AS2 and is ready for transfer definitions to be created.

What to do next

Create the transfer definitions for incoming and outgoing AS2 transfer requests.

Creating Transfer Definitions for Incoming and Outgoing AS2 Requests

Transfer definitions are required to perform all Internet Server transfers. The transfer definitions are slightly different based on whether the request is an incoming AS2 request initiated by an AS2 client, or an outgoing AS2 request initiated by Internet Server to an AS2 server.

Incoming AS2 Requests

To create a transfer definition from an incoming AS2 request, perform the following steps:

Procedure

1. Go to **Transfers > Internet Server Transfers > Add Transfer**.

2. Enter the required information listed in the table below.

Parameter	Instruction
Client File Name	Set to any value.
Server File Name	Set to the desired server file name. If the AS2 client does not include the file name in the AS2 S/Mime extension (most do), then you should include the full file name in this field and set Directory Transfer to No. Since most AS2 clients include the file name in the AS2 S/Mime extension, set Directory Transfer to Yes and use the # (ClientFileName) token to define the file name.
Directory Transfer	Set to Yes if the AS2 client includes the file name in the S/Mime extension.
Description	Set to a description that describes the transfer request.
Authorized User Id	Select the AS2 user configured in the server definition AS2 Options > User ID for incoming requests . This user ID was defined in a prior step.
Authorized Group Id	Leave this field blank.
Server Name	Select the target server where the file is saved. This is typically an SFTP server or a Platform Server, although this could be any server type.
Transfer direction	Set to Upload to Server. Downloads are not supported for AS2 transfers.
Client Protocols Allowed	Set to AS2.

Parameter	Instruction
Department	Set as required.
Virtual Alias	This field is not used by incoming AS2 transfers.

3. Enter any other transfer parameter as required.
4. When you have entered all of the necessary parameters, click **Add** to create the transfer definition.

Outgoing AS2 Requests

To create a transfer definition from an outgoing AS2 request, perform the following steps:

Procedure

1. Go to **Transfers > Internet Server Transfers > Add Transfer**.
2. Enter the required information listed in the table below.

Parameter	Instruction
Client File Name	Set to any value.
Server File Name	Set to the desired server file name. We suggest setting this to the \$(ClientFileName) token. This sets the AS2 S/Mime filename token to the file name sent by the transfer client.
Directory Transfer	Set to Yes.
Description	Set to a description that describes the transfer request.
Authorized User Id	Select the user that initiates the file transfer request.
Authorized Group Id	Leave this field blank.
Server Name	Select the target AS2 server where the file is sent.

Parameter	Instruction
Transfer direction	Set to Upload to Server. Downloads are not supported for AS2 transfers.
Client Protocols Allowed	Set to All.
Department	Set as required.
Virtual Alias	This field is not used by incoming AS2 transfers.

3. Enter any other transfer parameter as required.
4. When you have entered all of the necessary parameters, click **Add** to create the transfer definition.

Information for AS2 Transfers

The following table lists the information you must send to your AS2 partners.

Information	Description
AS2 Partner ID	The MFT AS2 partner ID defined in the Configuration AS2 Server tab in the server definition.
Encryption Public Certificate	The public key associated with the AS2 Encryption System Key configured in the server definition. You can get this information from Management > Protocol Keys > System Keys > Manage System Keys. Select the system key; the public key is in the Public Certificate tab.
Signing Public Certificate	The public key associated with the AS2 Signing System Key configured in the Server definition. You can get this information from Management > Protocol Keys > System Keys > Manage System Keys. Select the system key; the public key is in the Public Certificate tab.
HTTPS Public Certificate	When using HTTPS for incoming AS2 requests, this is the certificate of the MFT HTTPS connector. You can extract this public key by using a browser to

Information	Description
	access Internet Server's HTTPS port.
AS2 Server URL	This information is in the Configure AS2 Server page in the Receive URL field. This information is required for AS2 clients to connect to the MFT AS2 service.
AS2 Async Response URL	This information is in the Configure AS2 Server page in the Async Response URL field. This information is generally transmitted during AS2 negotiation, but some partners might require this.

Executing AS2 Transfers

AS2 Clients Can Initiate Transfer to Internet Server

- An AS2 client initiates a transfer to the MFT **Receive URL** and sends its local AS2 ID.
- MFT matches the local AS2 ID against the server definition's partner AS2 ID.
- The server definition defines the user for file transfers with this AS2 partner.
- The user definition is used to search for the `Upload Transfer` definition.
- The transfer definition points to the target Server where the file will be saved.
- As the AS2 client sends data to MFT, the data is streamed to the target server.

MFT Clients (HTTP, Platform Server, SFTP, and so on) can Initiate Transfer To Send To AS2 Server

- An SFTP client initiates a transfer to MFT.
- Credentials (key or password) are used to log in as a user.
- The SFTP Client defines a `Virtual Alias` and a file name.
- The `Virtual Alias` is used to select a transfer definition.
- The transfer definition points to an AS2 server.

- As the SFTP client sends data to MFT, the data is streamed to the AS2 server.

Using the MFT Scheduler

The MFT Scheduler allows you to execute tasks based on a scheduled interval. The following table describes the three components of the MFT Scheduler:

Component	Description
Calendars	Calendars allow you to configure inclusion or exclusion dates for the scheduler triggers.
Jobs	Allows you to define the tasks that must be executed.
Schedule Triggers	Defines when scheduled jobs must execute.

The MFT Command Center is required to configure and execute scheduler jobs. Scheduler jobs execute on Command Center. However, Command Center initiates jobs on Internet Server instances for the following job types:

- Internet Transfer
- Purge Log Files

Multiple Command Center instances can execute in a High Availability Active/Active environment.

 **Note:** To execute in HA Active/Active mode, the time for all Command Center instances must be synchronized to within 1 second.

You can execute scheduler jobs in the following ways:

- Through the **Execute Now** button in the **Update Job** page.
- Through the scheduler triggers.
- Through a REST call: `/jobs/jobName/{jobName}/groupName/{groupName}/execute`
- Through a Platform Transfer `Execute Job` request initiated to Command Center.

Configuring and Starting Command Center Scheduler

You must configure the scheduler and start the scheduler before you execute scheduler jobs.

Configuring the Scheduler

To configure the scheduler, perform the following steps:

Procedure

1. Go to **Management > Command Center Services > Scheduler Service > Configure Scheduler Service**.
2. Select the Command Center instance that you want to configure.
3. Enter the required information listed in the table below.

Parameter	Instruction
Thread Pool	Set the number of threads to be used by the scheduler.
Misfire Threshold	Define the misfire timeout

4. Click **Update**.

Starting or Stopping the Scheduler

To start the scheduler, perform the following steps:

Procedure

1. Go to **Management > Command Center Services > Scheduler Service > Scheduler Service Status**.
2. Select the Command Center instance that you want to start or stop.
3. Click any of the buttons from the following table:

Option	Description
Status	Displays the current scheduler status on this Command Center instance.
Start	Start the scheduler on this Command Center instance.
Stop	Stop the scheduler on this Command Center instance.
Hold	Hold the scheduler on this Command Center instance.

Calendars

Calendars define inclusion or exclusion dates for scheduled jobs.

To view calendars, go to **Management > Scheduler > Calendar > Add Calendar** or **Management > Scheduler > Calendar > Manage Calendars**.

Creating Exclusion Calendar

Exclusion calendars define dates that must be excluded when a job is scheduled to be executed. This is the default type of calendar. You can set up holiday calendars that define the dates where a scheduler job must not execute.

To create an inclusion calendar, perform the following steps:

Procedure

1. Go to the **Add Calendar** Page.
2. Enter the required information in the fields described in the table below.

Field	Description
Name	Enter a descriptive name for the exclusion calendar.
Time Zone	Enter the time zone for the calendar. By default, the scheduler uses server

Field	Description
	time zone.
Type	Select Exclusion.
Days	Click dates in the calendar. When you click on a date, it is added to the Selected Days list.
Selected Days	Defines the selected days for the calendar. To remove a date from this list, click Remove .

3. Click **Add**.

Creating Inclusion Calendar

Inclusion calendars define dates that must be included when a job is scheduled to be executed. Scheduled jobs are executed only on the selected dates.

To create an inclusion calendar, perform the following steps:

Procedure

1. Go to the **Add Calendar** Page.
2. Enter the required information in the fields described in the table below.

Field	Description
Name	Enter a descriptive name for the inclusion calendar.
Time Zone	Enter the time zone for the calendar. By default, the scheduler uses server time zone.
Type	Select Inclusion.
Days	Click dates in the calendar. When you click on a date, it is added to the Selected Days list.

Field	Description
Selected Days	Defines the selected days for the calendar. To remove a date from this list, click Remove .

3. Click **Add**.

Scheduler Jobs

With the scheduler jobs you can define a variety of tasks that can be performed.

The following table lists the tasks that are supported.

Task	Description						
Platform Transfer	Initiate a Platform Transfer on a selected Platform Server to send a file to a target Platform Server or Internet Server.						
Internet Transfer	Initiate an Internet Server transfer to send a file to or receive a file from a target server.						
Non-Event Transfer Alert	Define a non-event transfer alert. Non-event alerts allow you to execute an alert when a transfer does not execute. Non-event transfer alerts are defined earlier in this document.						
Execute Command	Execute a command. Two types of commands can be executed: <table> <tr> <th>Command</th><th>Description</th></tr> <tr> <td>Local</td><td>Execute a command on the Command Center.</td></tr> <tr> <td>Platform Server</td><td>Execute a command on a target Platform Server.</td></tr> </table>	Command	Description	Local	Execute a command on the Command Center.	Platform Server	Execute a command on a target Platform Server.
Command	Description						
Local	Execute a command on the Command Center.						
Platform Server	Execute a command on a target Platform Server.						
Execute Java Class	Execute a Java class that you have created.						

Task	Description
Send Email	Send an email to one or more recipients.
Batch Job	Create a batch of multiple jobs. All of the jobs in the batch can be executed at the same time.
Purge DB Tables	Purge records from database tables. There are many database tables that accumulate records that are never deleted. This job type allows you to purge records on a regular basis.
Purge Log Files	Clean up log and trace files on Internet Server and Command Center instances.
Key Expiration Notification	Notify users when keys (public or system) are about to expire.

There are many parameters associated with scheduler jobs. The help pages provide detailed instructions on how to configure scheduler jobs.

Creating Scheduler Jobs

To create a scheduler job, perform the following steps:

Procedure

1. Go to **Management > Scheduler > Jobs > Add Job**.
2. Go to **Management > Scheduler > Jobs > Manage Jobs**.
3. Enter the required information listed in the table below

Field	Description
Enabled	Set to Yes.
Job Name	Enter a unique job name.

Field	Description
Description	Enter a description for the job
Job Type	Select the desired job type.
Batch Name	Select None.
Department	Leave this blank
Authorized User Id	Leave this blank
Authorized Group Id	Leave this blank

4. Enter the required information in the **Platform Transfer** tab.

Parameter	Description
Transfer	Select a predefined Platform Transfer record. To run a Platform Transfer job, you must predefine a Platform Transfer. When you select a Platform Transfer job, the parameters defined on that Platform Transfer definition are displayed on the right side of the tab under the Current Transfer Settings heading. You can override these parameters as needed by entering the parameters on the left side of the tab. Note: Only a subset of the defined parameters can be overridden. Parameters that are not overridden are taken from the selected Platform Transfer definition.
Run Transfer As	Defines the MFT user under whose authority the Platform Server job executes. These parameters can be overridden or they can use the definitions in the selected Platform Transfer.
Transfer Direction	Select Send File Or Receive File.

Parameter	Description
Server Name	Defines the Platform Server where the Platform Transfer is executed.
Initiator File Name	Defines the initiator file name. This is the local file name of the Initiating Platform Server
Responder Host Type	Defines whether the Initiating Platform Server will use a node name or an IP name to connect to the target Platform Server or Internet Server.
Responder Host Name	Defines the IP name or node name that the initiating Platform Server uses to connect to the target Platform Server.
Responder Port Number	Defines the Port Number used when responder host type is set to IP name.
Initiating User ID	Defines the user ID used when Command Center initiates a request to the Initiating Platform Server. If not defined, the server definition server credentials are used.
Initiating Password	Defines the password used when Command Center initiates a request to the Initiating Platform Server. If not defined, the Server definition server credentials are used.
Responder User ID	Defines the user ID used when the initiating Platform Server sends a request to a target Platform Server. If not defined, the Platform Server user profile can be used to define the credentials.
Responder Password	Defines the password used when the initiating Platform Server sends a request to a target Platform Server. If not defined, the Platform Server user profile can be used to define the credentials.
Wait For Completion	Defines whether the job waits for the Platform Transfer to complete.

Parameter	Description								
	<table><tr><th>Option</th><th>Description</th></tr><tr><td>Yes</td><td>Wait for the Platform Transfer to complete. Command Center polls the Initiating Platform Transfer at increasing intervals until the Platform Transfer completes.</td></tr><tr><td>No</td><td>The job completes as soon as the initiating Platform Server accepts the transfer request.</td></tr><tr><td>Synchronous</td><td>The job waits until the Platform Transfer executes. We suggest using this option only if the transfer is expected to complete in 60 seconds or less.</td></tr></table>	Option	Description	Yes	Wait for the Platform Transfer to complete. Command Center polls the Initiating Platform Transfer at increasing intervals until the Platform Transfer completes.	No	The job completes as soon as the initiating Platform Server accepts the transfer request.	Synchronous	The job waits until the Platform Transfer executes. We suggest using this option only if the transfer is expected to complete in 60 seconds or less.
Option	Description								
Yes	Wait for the Platform Transfer to complete. Command Center polls the Initiating Platform Transfer at increasing intervals until the Platform Transfer completes.								
No	The job completes as soon as the initiating Platform Server accepts the transfer request.								
Synchronous	The job waits until the Platform Transfer executes. We suggest using this option only if the transfer is expected to complete in 60 seconds or less.								
Wait Timeout	When Wait for Completion is set to Yes, defines the duration for which the scheduler waits before the request is terminated with a timeout error. Note that it is possible for the Platform Transfer to complete after a timeout occurs.								

5. When you have completed Platform Transfer parameters, you can perform either of the following actions.

Action	Description
Click Add to add the scheduler job.	The job is added with no scheduling information. The job does not execute until you define the scheduling information. Alternatively, you can execute the scheduler job through a REST call or through a Platform Server ExecuteJob request.
Define the Scheduling Information.	This defines when a job executes and the frequency that a job executes. See the section on Schedule Triggers . Then, click the Add button to add the scheduler job.

Schedule Triggers

Schedule triggers set the parameters when a scheduler job executes. These parameters define when the job starts, the frequency and the days of the week and months of the year.

To schedule triggers, perform the following steps:

Procedure

1. Enter the required information in the **Scheduling Information** tab.

Parameter	Description
Not Scheduled	The request is not scheduled. Use this when you want to execute the scheduler job through a REST call or through a Platform Transfer ExecuteJob request.
Execute Once	Execute one time at a defined date and time.
By Minute	Execute every X minutes. You can define the times of the day and days of the week when the job executes.
By Hours	Execute every X hours. You can define the times of the day and days of the week when the job executes.
By Day	Defines the time of day and the days of the week when the job executes.
By Month	Defines the time of day, day of the month, and the months the job executes.
By Calendar	Specify an inclusion calendar and a time of day to execute a job.

2. Enter the required information in the **Additional Information** tab.

Parameter	Description
Exclusive	Defines whether a job runs if it is currently running.
Recoverable	Defines whether a job runs if a scheduler operation fails.
Dependency Change/Job	Allows you to define a job that will run when: • This job executes unsuccessfully. • This job executes successfully.

3. Click **Add**.

Debugging Scheduler Jobs

To debug scheduler jobs, follow the following steps.

Procedure

1. Use the **Manage Jobs** page. This page displays **Next Fire Time** in the results table. This is the data/time the job is scheduled to run again.
2. Use the **Active Jobs** page to view jobs currently running.
3. Go to **Diagnostics > Events > Search Events** to view scheduler events.

All events are displayed. You can select Request Type scheduler to only select scheduler events.

The following directory consists of all the Scheduler job logs:

```
<MFT-Install>/logs/messages/Scheduler-yyyy-mm-dd.txt
```

4. TIBCO Technical support might request this file to diagnose scheduler problems.

Manage Scheduler

The Manage Scheduler page introduces enhancements to job management, similar to those made on the Manage Users page. These updates improve responsiveness and

efficiency when managing a large number of scheduled jobs, particularly with a large number of entries.

The configurable properties are provided in the following table.

Key enhancements:

- **Sync Jobs:** The new **Sync Jobs** button allows super-admins and users with scheduler rights to initiate job data transfer at any time. Department-admins cannot perform sync jobs.
- **Improved Performance:** The page efficiently handles a large volume of scheduled jobs, resolving previous unresponsiveness issues.
- **Pagination:** Pagination improves navigation and user experience when browsing many scheduled jobs.

These changes affect only the display and do not impact the operation of existing scheduled jobs or the scheduler's core functionality.

Delegated Administration

Delegated administration allows you to decentralize administrative tasks. It allows you to limit the resources that can be managed by individual administrators. Delegated administration uses the concept of a department. Users can be assigned to a department; users can also be configured to manage other departments. Delegated administration is performed by defining departments. You can then assign the following resources to a department:

- Users
- Groups
- Internet Server Transfers
- Platform Server Transfers
- Alerts
- Servers
- Jobs

This allows you to restrict the department administrators to list, add, and update only the definitions for their department or for departments that the administrator can manage. The term *Department Administrator* refers to a user with at least one administrative right that is assigned to a department.

In addition to limiting updating resources, being a department administrator restricts the data that can be displayed or managed on these pages:

- Search Events
- Search Alert History
- Search Audits
- Manage DNI Daemons
- Admin Changes
- Active Transfers

Visibility

When a resource (user, server, and so on) is assigned to a department, you can also assign a visibility to the resource. Visibilities can be public or private.

Visibility Type	Description
Public	The resource can be viewed by other department users. When creating or updating other resources, this resource can be viewed and possibly assigned to another resource.
Private	The resource cannot be viewed by other department users.

For example, assume a server definition was assigned to department A and the admin user is assigned to department B:

- If you set the visibility for the server definition to `Public`, then transfer definitions created by the department admin can see this server and assign transfers to this server.
- If you set the department for a Server definition to `Private`, then transfer definitions created by a department admin cannot see this server and cannot assign transfers to this server.



Note: Users that are not assigned to a department can view and manage resources assigned to a department, regardless of the visibility.

Creating Departments

Users assigned to a department are not authorized to create departments. Only users with the necessary rights that are not assigned to a department can add, view, or update departments.

To create a department, perform the following steps:

Procedure

1. Go to **Partners > Departments > Add Department**.
2. Enter the required information listed in the table below.

Field	Instruction
Department Name	Assign a unique name to the department.
Description	Enter a description for the department.

3. Click **Add**.

Adding Users to Departments

There are two ways to assign users to departments:

- Add or Update User
- LDAP Sync (through Add/Update Authenticators)

Adding or Updating User

To add or update a user to a department, perform the following steps:

Procedure

1. In the **Add/Update User** page, click the **Optional User Properties** tab.
2. Enter the required information listed in the table below.

Field	Instruction
Department	Set to the desired department.
Visibility	Set to Public or Private as needed.
Manage Departments	Select all of the departments that you want this user to manage. Department users are able to create and manage resources in the

Field	Instruction
	department they are assigned to and they can create and manage resources in the departments that they can manage.

- When you have finished configuring the department options, click **Add** to add the user or **Update** to update the user.

LDAP Sync

When you define an authenticator, you can configure the authenticator to synchronize the user's department based on an attribute in the LDAP server.

To add a user to a department using LDAP sync, perform the following steps:

Procedure

- Go to **Management > Authenticators > Add Authenticator** or **Configuration > Authenticators > Configure Authenticators**.
- Enter the required information listed in the table below.

Field	Instruction
Authenticator Properties	LDAP Attributes.
Department	Set this to the LDAP attribute that contains the department.

- To synchronize users, go to **Administration > LDAP Sync**.
You can synchronize a single user, a single authenticator, or all authenticators.
- When the LDAP Sync is complete, synchronized users must have the department set in their user definition.

Result

Users have been added and assigned to a department. The user is ready to be managed by department administrators.

What to do next

Proceed to the next step to see how other department resources are managed.

Managing Other Resources

Once you have assigned a user to a department and have assigned the necessary administrative rights to this user, the user can add or update other resources. When we use the term *resource*, we mean users, groups, Internet Server transfers, Platform Server transfers, alerts, jobs, or servers.

The following list indicates some of the rules associated with adding or updating other resources.

1. When an admin adds a resource, the default department is the department of the admin adding the resource.
2. When an admin not assigned to a department adds a resource, the resource is added with no department.
3. Admin users not assigned to a department can manage resources assigned to a department and resources not assigned to a department.
4. Admin users that can manage multiple departments, can view and update resources in any department they can manage.
5. Admin users that can manage multiple departments can set the department for resources to any department they can manage.
6. When a department resource is defined as public, department users that are not configured to manage the department of the resource cannot update or view the resource.

For example, department users that can manage DeptA cannot view or update transfer definitions assigned to DeptB, even if the transfer definition is defined as Public.

7. When a resource assigned to a department is defined as Public, department users that are not configured to manage the department of the resource, can still work with the resource.

Let us assume UserA can manage DeptA and UserB can manage DeptB.

- When UserB creates a transfer definition, it can assign UserA to the transfer definition because UserA is a public user.
- When UserB creates a group, it can assign UserA to the group, because UserA is a public user.

Viewing Resources

MFT creates a variety of database records and allows admins to search these tables. For example, records in the following table are created:

Record	Description
Search Events	Records are created for a variety of processes, including Command Center Initiated Platform Transfers, scheduler requests, JMS-initiated requests and FileShare requests.
Search Alert History	Records are created whenever an alert is executed.
Search Audits	Records are created for each transfer, both successful and unsuccessful transfers.
Admin Changes	Records are created for each time an admin change is made.

When an admin assigned to a department uses these features, the records displayed are restricted to:

- Records created for the department the admin user is assigned to.
- Records created for any department the admin can manage.

Configuring and Managing DNI Daemons

DNI (or pDNI) is a Perl based event-driven process that performs file transfers when it detects that a file has been created or modified. DNI typically performs transfers using a Platform Server Command (ftmscmd, cfsend, and cfrecv). It can also be configured to execute a script to perform a file transfer using SCP or some other file transfer script. DNI can also be configured to receive files from MFT Internet Server through the FTP protocol.

DNI can perform transfers in the following ways:

Transfer Option	Description
DNI Send	Send a file to a partner Internet Server or Platform Server when a file is created or modified.
DNI Receive	Receive a file from a partner Internet Server or Platform Server when a file is created or modified.
FTP Receive	Receive a file from a partner Internet Server using the FTP protocol when a file is created or modified.

This section shows how to set up DNI and how to configure DNI from the Command Center. To learn how DNI can be configured to send files to and receive files from, Internet Server transfer partners using SFTP (or any other protocol), see *Using DNI to Send and Receive files to Internet Server partners*.



Note: Managing DNI is only supported on Command Center. It is not supported on the Internet Server.

DNI templates can be managed by Command Center or you can configure and start the templates manually on the server where DNI is executing. This guide explains how to manage the templates through Command Center.

Installing DNI

DNI is distributed as a TAR file in the following directory:

Component	Directory
Platform Server for UNIX	<PSU-Install>/dni/dni.tar
Platform Server for Windows	<PSW-Install>/dni.tar
MFT Command Center	<MFT-Install>/distribution/dni/dni.tar



Note: Some products distribute the `dni.tar` file as a zip file in the following format: `pDNI.tar.Vvrm.zip`

Installing DNI

Installing DNI is the same for each platform. Perl is required to run DNI. Perl is standard on Unix systems. If you want to use DNI on Windows, you must download PERL for Windows. There are some suitable Perl distributions that run on Windows.

Procedure

1. If the distribution file is a zip file, extract the zip file.
2. Create a directory for DNI.
3. Use the `untar` command to the `dni.tar` file into the DNI directory.
4. Execute this command to define and encrypt the DNI management credentials:

```
perl DNIDaemon.pl c:DNIconfig.cfg encrypt
```

5. Enter the user ID and password that is used by Command Center to communicate with this DNI instance. Remember these credentials. They are configured on the MFT server definition in the DNI management User ID and the DNI management password fields.
6. Using a text editor like `vi` or notepad, edit the `DNIconfig.cfg` file.

7. Edit the following parameters:

Parameter	Instruction
ListenIPPort	The default port is 47777. If this port is being used, set to the desired port. You need to enter this information on the MFT server definition in the DNI management port field.
AdapterIPAddress	Set this to the IP address of the Command Center instance. Enter All to accept requests from all incoming IP addresses.
SuppressCMDPrompt	When running on Windows, set this to Yes if you want to suppress the command prompts for started DNI templates.
FastStart	Set to Yes if you want to speed up initialization processing when you have many DNI templates started.

For details about the DNIconfig configuration parameters, enter the following command:

```
perl DNIDaemon.pl help
```

Details about configuring the DNI daemon configuration file are displayed.

Starting the DNI Daemon

To start the DNI daemon, perform the following steps:

Procedure

1. Enter the following command from the DNI directory:

```
nohup perl DNIDaemon.pl c:DNIconfig.cfg&
```



Note: The & indicates that this will run in the background.

2. Once DNIDaemon is started, start the DNI configuration on Command Center.

Configuring Command Center to Support DNI Management

To add or update a server definition to support DNI management, perform the following steps:

Procedure

1. To add a server, go to **Partners > Servers > Add Server**.
2. To update a server, go to **Partners > Servers > Manage Servers**.
3. Enter the required information listed in the table below.

Parameter	Description
Server Name	Define a unique descriptive server name.
IP Address or fully	Define the IP name of the server where DNI is running.
IP Port	This is used by Platform Server but is not used by DNI management.
Server Type	Must be Platform Server.

4. Enter the required information under the **Management Options** tab.

Parameter	Instruction
Manage Platform Server	Select this checkbox.
DNI Management Port	Enter the port defined in <code>DNISConfig.cfg ListenIPPort</code> parameter.
DNI Management User Id	Enter the DNI Management User ID that you set when configuring <code>DNIDaemon.pl</code> .

Parameter	Instruction
DNI Management Password	Enter the DNI Management Password that you set when configuring DNIDaemon.pl
DNI Confirm Password	Confirm the password you entered.

5. Click **Add** or **Update**.

What to do next

When the server is configured with DNI Management, proceed to the Command Center DNI Management pages.

Managing DNI Daemons

Command Center allows you to perform the following DNI functions:

- Add, list, delete, and update DNI templates.
- Start and stop DNI templates.
- List, delete, and view DNI log files.

To manage DNI daemons, perform the following steps:

Procedure

1. Go to **Platform Server Management**.

A list of all servers that were configured to manage DNI templates is displayed

2. The **Manage DNI Templates** page is broken up into two parts:

Component	Description
Results Table	One line is displayed for each DNI template found on the target DNI server.

Component	Description
Template Data	When you click a template name, the template configuration is displayed in this text box. The Template Data also has a Generate Template button. This feature is discussed in more detail later in this document in the section titled Generate Template .

- From the **Manage DNI Templates** page, click one of the existing templates.
The **Template data** text box is populated with the information from this template.
- Edit the information in the **Contents** text box.
- After updating the template, you can perform one of the following actions:
 - Change the **Template Name** to a new name and click **Add**. The template is added to the DNI template directory. If the template already exists, an error is displayed and the **Add** action fails.
 - Click **Update** to update the template defined by the **Template Name** field.
- When you have created or updated a template, find the entry for that template in the **Manage DNI Daemon Templates** results table. Two columns are located on the right side of the table.

Column	Description
Action	Allows you to start or stop a template.
Log	Allows you to view log files for the template.

- Click **Start** on the template that you just added or updated.
A request is sent to the DNI daemon to start the template.
The start request can take a few seconds to complete. A message is displayed indicating whether the start was successful. A start request can fail for several reasons. The most common reasons are:
 - An invalid parameter or invalid parameter value was defined.
 - The LocalDirectory for a DNI Send was not found.

- Post Action SuccessFile, FailureFile, and NetworkErrorFile point to a directory that does not exist (when the Post Action is Move).

Regardless of whether the start request was successful or failed, you can view the log file to see what happened.

8. Click **View Logs** for the template.

A list of log files is displayed. The format of the log file is:

```
templateName.YYYYMMDD.HHMMSS.log
```

The most recent log files are listed first.

9. Click the **Log File Name**.

The log file is downloaded and your browser prompts you to open the file or save the file. You can open the file with a text editor.

If the DNI template fails to start, the cause of the failure is displayed. If the start was successful, you can see the log of transfers attempted.

i Note: Just because the template **Start** was successful does not mean that transfers are working for that template. It is possible that the template parameters were defined correctly, but the Platform Server command did not work. The log file shows you if transfers are working correctly and if not, the reason that transfers are failing.

Generating Templates

DNI templates are just configuration files that can be updated by a text editor. The **Manage DNI Daemon Templates** page provides a centralized text editor to manage these templates. But even using the **Manage DNI Daemon Templates** pages can be confusing for a non-technical user, or even for a technical user that does not work with DNI frequently. The **Generate Template** button on the **Manage DNI Daemon Templates** page provides a GUI interface to create DNI templates.

Procedure

1. When you click **Generate Template** from the **Manage DNI Daemon Templates** page,

the **Create DNI Template** page is displayed. This page allows you to configure the following DNI template fields:

- Required Parameters
- Transfer File Parameters
- Platform Server Transfer Parameters
- FTP Transfer Parameters
- Scan and Transfer Parameters
- Transfer Disposition Parameters
- DNI Schedule Parameters
- Miscellaneous Parameters
- High Availability Parameters

For more information about these parameters, see the admin help pages.

Configuring Template Parameters

You can configure template parameters to execute on Windows or UNIX. When running on Windows, this page generates the `FTMSCMD` command. When running on UNIX, this page generates the `cfsend/cfrecv` commands.

Request Types

The following table lists request types.

Type	Description
Send	This is a DNI Send.
Receive	This is a DNI Receive.
Receive FTP	This is a DNI Receive FTP

When you change the request type, this page changes to reflect the parameters required by each request type.

To configure request types, perform the following steps:

Procedure

1. When you click **Send**, the **Local Directory** parameter is displayed.
2. When you click **Receive**, the **Remote Directory** parameter is displayed.
3. When you click **Receive FTP**, the **Remote FTP Directory** parameter is displayed and the **Platform Server Transfer Parameters** section is changed to **FTP Transfer Parameters**.
4. After entering all of the required and optional parameters, click **Generate Template**. You are brought to the **Manage DNI Daemon Templates** page and the **Template Data** is populated with the generated template data.
5. Make changes to the template.
6. Enter the template name.
7. Click the **Add** button to add the new template.
8. Click the **Update** button to update the existing template.
9. Once the template is added or updated, you can start the template by clicking the **Start** Action link.

You can only generate a new template through the **Create DNI Template** page. You cannot parse an existing template with this page.

10. Once you have generated the template, if you want to modify the template, edit the data in the **Template Data** text box.

Using DNI to Send and Receive files to Internet Server Partners

This section describes how you can configure pDNI to send files to target SSH servers and receive files from target SSH servers.

pDNI was designed to allow Platform Servers to automate transfers to other Platform Servers. Since Internet Server supports the Platform Server protocol, pDNI can be used to automate transfers with almost any server configured to Internet Server.

pDNI supports three modes of operation:

Operation	Description
DNI Send	Allows you to send files to a transfer partner
DNI Receive	Allows you to receive files from a transfer partner
DNI Receive FTP	Allows you to receive files from a transfer partner through the FTP or FTPS protocol

This section explains how to use pDNI to transfer files with partner SSH servers. Since MFT hides the actual file transfer server from the initiating client through the use of Virtual Aliases, the same techniques can be used to transfer files with servers supporting other protocols, like FTP, Azure Storage, Amazon Storage, Google Storage, and so on.

See [Configuring and Managing DNI Daemons](#) to install and configure pDNI.

There are two modes of pDNI:

- [Send files to Target SSH Servers](#)
- [Receive files from Target SSH Servers](#)

File Transfer Initiated by pDNI to Target SSH Server

- Go to **pDNI > Platform Server > Internet Server > SSH Server**.
- pDNI scans a directory to determine if a file should be sent. pDNI can be configured to wait for a predefined interval to ensure that the file size and date or time modified has not changed before initiating the transfer.
- pDNI detects a file that needs to be sent.
- pDNI initiates a Platform Server send command (UNIX: `cfsend` Windows: `ftmcmd`) to send the file to Internet Server. The `cfsend` or `ftmcmd` remote file name will point to a virtual alias defined in an Internet Server transfer definition.
- Internet Server finds a match on the initiating user and the transfer definition virtual alias for a file upload. The matching transfer definition defines the target server where the file will be transferred. In this case, the target server is an SSH server.
- Internet Server initiates a connection to the target SSH server.
- The SSH server validates the incoming credentials (password or SSH key).
- As packets are received, the SSH server writes the file to the defined location.
- Platform Server sends data packets to Internet Server. As packets are received, Internet Server converts the data from the Platform Server protocol to the SSH server protocol and sends the packets to the defined SSH server.
- pDNI performs defined post actions to ensure that the file is not transmitted again. If the transfer is unsuccessful, pDNI can retry the transfer at the next scan interval.

Sending Files to Target SSH Server

To send files to a target SSH server, perform the following steps:

Procedure

1. [Configure the Internet Server Platform Server service and start the Platform Server service.](#)
2. [Configure a server definition to define the connectivity and credentials for the target SSH server.](#)

3. [Create a user definition for the incoming client request.](#)
4. [Configure Platform Server to connect to Internet Server.](#)
5. [Create a transfer definition that uses the defined user and server definitions.](#)
6. [Create and configure the pDNI template.](#)
7. [Start the pDNI template.](#)
8. [Verify that the pDNI template is transferring files.](#)



Note: The first four steps are common for DNI Send and DNI Receive.

Configure the Internet Server Platform Server Service and Start the Platform Server Service

The section titled [Configuring and Starting MFT Transfer Services](#) shows how to configure and start the Platform Server service. This service must be started before pDNI can initiate transfers with target SSH Servers.

Configuring a Server Definition to Define the Connectivity and Credentials for Target SSH Server

To configure a server definition to define the connectivity and credentials for target SSH server, perform the following steps:

Procedure

1. Go to **Partners > Servers > Add Server**.
2. Enter the required information listed in the following table.

Parameter	Instruction
Server Name, IP Address, IP Name, IP Port	As required.
Server Type	Set to SSH.
Server Platforms	Set to UNIX.

3. Configure the server credentials if you are using the user ID and password authentication.
4. Configure SSH Options if you want to use SSH key authentication.
5. Click **Add** to add the server definition.

Retrieving Target Server SSH Key

To retrieve target server SSH key, perform the following steps:

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Select the server you just added.
3. Click **Retrieve SSH public key through this server**.

Creating a User Definition for Incoming Client Request

To create a user definition for incoming client requests, perform the following steps:

Procedure

1. Go to **Partners > Users > Add User**.
2. Enter the required information.
3. In the **Rights and Groups** section, ensure the user is assigned TransferRight.
4. Open the **Optional User Properties** tab.

5. To accept all incoming protocols, set **Client Protocols** to All.
6. To accept Platform Server requests, set **Client Protocols** to Platform Server.
7. To accept Platform Server SSL/TLS requests, set **Client Protocols** to Platform Server SSL/TLS.
8. Clear the **Change Password at Next Login** checkbox.
9. Click **Add** to add the user definition.

Result

The user is ready to accept incoming Platform Server transfer requests.

What to do next

Proceed to the next steps to configure the Platform Server Node and profile to connect to Internet Server.

Configuring Platform Server Definitions to Connect to Internet Server

To communicate to Internet Server, create the following Platform Server definitions.

Definition	Description
Node	Defines connectivity information to connect to Internet Server.
Profile	Defines credentials needed to authenticate to Internet Server.

On Unix, creating node and profile definitions required the user to be a root user or in the `cfadmin` group. On Windows, creating node and profile definitions required the user to be an administrator user or in the `cfadmin` group.

Creating Node Definition

To create a node definition, perform the following steps:

Procedure

1. Enter the following command:

```
cfnode n:mftisnode
```

2. Enter the appropriate number for each prompt.



Note: Enter the TCP port defined for the Internet Server Platform Server service.

Creating Profile Definition

To create a profile definition, perform the following steps:

Procedure

1. Enter the following command:

```
cfprofile n:mftisnode
```

2. Enter the user ID and password for the Internet Server user created in a prior step, when prompted.
3. If asked for a local user, ensure that the local user is set to the Unix or Windows user that started the DNI daemon.
4. If not asked for a local user, ensure the user that executed the `cfprofile` command must be the same user that started the DNI daemon.

Create Transfer Definition for Defined User and Server Definitions

To create a transfer definition for the defined user and server definitions, perform the following steps:

Procedure

1. Go to **Transfers > Internet Server Transfers > Add Transfer**.

2. Enter the required information listed in the table below.

Parameter	Instruction
Client File Name	Enter any value; this field is ignored.
Server File Name	Enter the target directory where the file is written.
Directory Transfer	Set to Yes.
Description	Set to a descriptive text value.
Authorized UserId	Select the user ID created in the prior step.
Server Name	Select the server created in a prior step.
Transfer Direction	Set to Upload.
Client Protocols Allowed	To accept all incoming protocols, set to All. To accept Platform Server requests, set to Platform Server. To accept Platform Server SSL/TLS requests, set to Platform Server SSL/TLS.
Virtual Alias	This value defines the directory that the user sees when logging into Internet Server. We suggest not entering any spaces and leave out slashes and backslashes. The Virtual Alias is used in the DNI Template > DNI Command parameter.

3. Enter the required information in the **Additional Transfer Information**.
4. To create files and return error if the file exists, set the write mode to Create.
5. To create or replace existing files, set the write mode to Create-Replace.

6. To create or replace existing files and to create directories as needed, set the write mode to `Create-Replace-New`.
7. Click **Add** to add the transfer definition.

Creating and Configuring the pDNI Template

The next step after completing all of the Internet Server definitions is to configure the pDNI template.

To configure the pDNI template, perform the following steps:

Procedure

1. Go to **Platform Server Management > Manage DNI Daemons**.
2. Select the server name where you want to create the DNI template.
A list of pDNI templates defined on that server is displayed.
3. Select an existing template for the Platform Server where you want the template to execute.

By default, pDNI includes the following sample send templates.

Sample Send Template	Description
<code>dnitemplate.send</code>	Unix system uses the <code>cfsend</code> command
<code>dnitemplate.wsend</code>	Windows system uses the <code>ftmcmd</code> command

After selecting a template name, the pDNI template is displayed in the **Template Data** at the bottom of the page.

4. Make the following changes to the template.

Parameter	Instruction
<code>LocalDirectory</code>	Set to the directory that you want to scan for files to send.

Parameter	Instruction								
DNICommand	set to the command that you want to execute. Below is the sample command for Unix (cfsend): <pre>cfsend n:mftisnode lf:"\$(LocalFileName)" rf:"/VirtualAlias/\$(LocalFile)" sm:y</pre> Below is the sample command for Windows (ftmscmd): <pre>ftmscmd /send /file /node:mftisnode "\$(LocalFileName)"/VirtualAlias/\$(LocalFile)"</pre> Note: The DNI Template RemoteDirectory parameter must be set to the virtual alias defined on the transfer definition in the step <i>Create a Transfer Definition for the defined User and Server definitions</i>.								
SubDirectory	Enter Yes to transfer files in subdirectories; otherwise enter No.								
SuccessAction	There are three options you can set: <table> <tr> <th>Option</th><th>Instruction</th></tr> <tr> <td>Leave</td><td>Keep the file in the directory but do not transfer again until the file is modified.</td></tr> <tr> <td>Delete</td><td>Delete the file transferred.</td></tr> <tr> <td>Move</td><td>Move the file transferred to the location defined by the SuccessFile parameter.</td></tr> </table> By default, this option is set to Leave.	Option	Instruction	Leave	Keep the file in the directory but do not transfer again until the file is modified.	Delete	Delete the file transferred.	Move	Move the file transferred to the location defined by the SuccessFile parameter.
Option	Instruction								
Leave	Keep the file in the directory but do not transfer again until the file is modified.								
Delete	Delete the file transferred.								
Move	Move the file transferred to the location defined by the SuccessFile parameter.								
NetworkErrorAction	Set to Retry for this test.								

Parameter	Instruction
FailureErrorAction	Set to <code>Retry</code> for this test.
ScanInterval	Set to <code>1</code> (Minute).
Template Name	Enter the name of the new template to be created.

5. Click **Add** to add the new template.

Starting pDNI Template

After saving the new template, a list of templates is displayed and you see the new template.

To start a pDNI template, perform the following steps:

Procedure

1. On the right side of the pDNI template, click **Start**.

Result

Internet Server communicates with DNIDaemon to start the template. A message displays whether the file has been started successfully.

A start request can fail for a variety of reasons. The most common reasons are:

- An invalid parameter or invalid parameter value was defined.
- The `LocalDirectory` for a `DNI Send` was not found.
- **Post Action** `SuccessFile`, `FailureFile`, and `NetworkErrorFile` point to a directory that does not exist (when the **Post Action** is `Move`)

If the template does not start successfully, you can follow the procedure "*Verify that the pDNI template is transferring files*" to determine the cause of the failure.

Verifying the pDNI Template is Transferring Files

Regardless of whether the start request was successful or failed, you can view the log file. The log file tells you whether the start request was successful or failed. It also shows you

the status of attempted transfers.

To verify that the pDNI template is transferring files, perform the following steps:

Procedure

1. Click **View Logs** for the template.

A list of log files is displayed. The format of the log file is:

```
templateName.YYYYMMDD.HHMMSS.log
```

The most recent log files are listed first.

2. Click the **Log File Name**.

The log file is downloaded and your browser typically prompts you to open the file or save the file. You can open the file with a text editor.

Result

If the start fails, the cause of the failure is displayed.

If the start is successful, you can see the log of transfers that were attempted.

 **Note:** Just because the template **Start** was successful, that does not mean that transfers are working for that template. It is possible that the template parameters were defined correctly, but the Platform Server command did not work. The log file shows you if transfers are working correctly and if not, the reason that transfers are failing. If you do not see any transfers attempted, it is possible that the LocalDirectory being scanned for files is empty.

What to do next

Check to see if pDNI is executing transfers.

1. Go to **Reports > Audits > Search Audits** page to view transfers initiated by the pDNI daemon.
2. Download and view the pDNI log file. The pDNI log file shows the following:
 - If any files were selected to download.

- The status of any file transfer request.

Receiving Files from Target SSH Servers

The following line depicts a typical DNI Receive from a target SSH server.

SSH Server Retrieve directory list > Internet Server > pDNI: Platform Server

SSH Server Receive files > Internet Server > pDNI: Platform Server

- pDNI issues a command to get a directory list from the Internet Server and the target SSH Server. The command points to a virtual alias defined in the Internet Server Transfer definition. Internet Server finds a match on the initiating User and the Transfer definition virtual alias for a file download. The matching transfer definition defines the target Server where the file is transferred. In this case, the target server is an SSH server.
- Internet Server initiates a connection to the target SSH server, passes credentials, issues a directory list command to the SSH server, and returns the directory list back to the Platform Server. pDNI can be configured to wait for a predefined interval to ensure that the file size and date or time modified has not changed before initiating the transfer.
- pDNI detects a file that needs to be received.
- pDNI initiates a Platform Server receive command (UNIX: cfrecv Windows: ftmscmd) to receive the file from Internet Server. The cfrecv or ftmscmd remote file name points to a virtual alias defined in an Internet Server transfer definition.
- Internet Server finds a match on the initiating user and the transfer definition virtual alias for a file download. The matching transfer definition defines the target server where the file is transferred. In this case, the target server is an SSH server.
- Internet Server initiates a connection to the target SSH server.
- The SSH server validates the incoming credentials (password or SSH key).
- The SSH server reads the file and sends data packets to Internet Server.
- As packets are received, Internet Server converts the data from the SSH protocol to the Platform Server protocol and sends the packets to the defined Platform Server.
- Platform Server receives data packets from Internet Server and writes the data to the

requested file.

- When the transfer completes, the PS-IS and IS-SSH connections gracefully terminates and audit records are written by both Platform Server and Internet Server.
- pDNI performs defined Post Actions to ensure that the file is not transmitted again. If the transfer was unsuccessful, pDNI can retry the transfer at the next scan interval.

To receive files from target SSH servers, perform the following steps:

Procedure

1. [Configure the Internet Server Platform Server service and start the Platform Server service.](#)
2. [Configure a server definition to define the connectivity and credentials for the target SSH server.](#)
3. [Create a user definition for the incoming client request.](#)
4. [Configure Platform Server definitions needed to connect to Internet Server.](#)
5. [Create a transfer definition that uses the defined user and server definitions.](#)
6. [Create and configure the pDNI template.](#)
7. [Start the pDNI template.](#)
8. [Verify that the pDNI template is transferring files](#)

i Note: The first four steps are common for DNI Send and DNI Receive and are reproduced here. You do not need to perform these steps if they have already been performed.

Configure the Internet Server Platform Server Service and Start the Platform Server Service

The section titled "[Configuring and Starting MFT Transfer Services](#)" shows how to configure and start the Platform Server service. This service must be started before pDNI can initiate transfers with target SSH servers.

Configuring a Server Definition to Define the Connectivity and Credentials for Target SSH Server.

The Platform Server service must be started before pDNI can initiate transfers with target SSH servers. To configure a server definition to define the connectivity and credentials for the target SSH server, perform the following steps:

Procedure

1. Go to **Partners > Servers > Add Server**.
2. Enter the required information listed in the following table.

Parameter	Instruction
Server Name, IP Address, IP Name, IP Port	As required.
Server Type	Set to SSH.
Server Platforms	Set to UNIX.

3. Configure the server credentials if you are using the user ID and password authentication.
4. Configure SSH Options if you want to use SSH key authentication.
5. Click **Add** to add the server definition.

Retrieving Target Server SSH Key

To retrieve target server SSH key, perform the following steps:

Procedure

1. Go to **Partners > Servers > Manage Servers**.
2. Select the server you just added.
3. Click **Retrieve SSH public key through this server**.

Creating a User Definition for Incoming Client Request

To create a user definition for incoming client requests, perform the following steps:

Procedure

1. Go to **Partners > Users > Add User**.
2. Enter the required information.
3. In the **Rights and Groups** section, ensure the user is assigned TransferRight.
4. Open the **Optional User Properties** tab.
5. To accept all incoming protocols, set **Client Protocols** to All.
6. To accept Platform Server requests, set **Client Protocols** to Platform Server.
7. To accept Platform Server SSL/TLS requests, set **Client Protocols** to Platform Server SSL/TLS.
8. Clear the **Change Password at Next Login** checkbox.
9. Click **Add** to add the user definition.

Configuring Platform Server to Connect to Internet Server

To communicate to Internet Server, create the following Platform Server definitions.

Definition	Description
Node	Defines connectivity information to connect to Internet Server.
Profile	Defines credentials needed to authenticate to Internet Server.

On Unix, creating node and profile definitions required the user to be a root user or in the `cfadmin` group. On Windows, creating node and profile definitions required the user to be an administrator user or in the `cfadmin` group.

Creating Node Definition

To create a node definition, perform the following steps:

Procedure

1. Enter the following command:

```
cfnode n:mftisnode
```

2. Enter the appropriate number for each prompt.

i Note: Enter the TCP port defined for the Internet Server Platform Server service.

Creating Profile Definition

To create a profile definition, perform the following steps:

Procedure

1. Enter the following command:

```
cfprofile n:mftisnode
```

2. Enter the user ID and password for the Internet Server user created in a prior step, when prompted.
3. If asked for a local user, ensure that the local user is set to the Unix or Windows user that started the DNI daemon.
4. If not asked for a local user, ensure the user that executed the `cfprofile` command must be the same user that started the DNI daemon.

Create Transfer Definition for Defined User and Server Definitions

To create a transfer definition for the defined user and server definitions, perform the following steps:

Procedure

1. Go to **Transfers > Internet Server Transfers > Add Transfer**.
2. Enter the required information listed in the table below.

Parameter	Instruction
Client File Name	Enter any value; this field is ignored.
Server File Name	Enter the target directory where the file is written.
Directory Transfer	Set to Yes.
Description	Set to a descriptive text value.
Authorized UserId	Select the user ID created in the prior step.
Server Name	Select the server created in a prior step.
Transfer Direction	Set to Upload.
Client Protocols Allowed	To accept all incoming protocols, set to All. To accept Platform Server requests, set to Platform Server. To accept Platform Server SSL/TLS requests, set to Platform Server SSL/TLS.
Virtual Alias	This value defines the directory that the user sees when logging into Internet Server. We suggest not entering any spaces and leave out slashes and backslashes. The virtual alias is added to DNI template RemoteCommand and DNICommand when DNI executes the directory list request and the file download.

3. Click **Add** to add the transfer definition.

Creating and Configuring the pDNI Template

At this point, all of the Internet Server definitions have been completed. You now need to configure the pDNI template.

To configure the pDNI template, perform the following steps:

Procedure

1. Go to **Platform Server Management > Manage DNI Daemons**.
2. Select the server name where you want to create the DNI template.
A list of pDNI templates defined on that server is displayed.
3. Select an existing template for the Platform Server where you want the template to execute.

By default, pDNI comes with two sample receive templates:

Sample Receive Template	Description
dnitemplate.recv	Unix system uses the cfrecv command.
dnitemplate.wrecv	Windows system uses the ftmshcmd command

After selecting a template name, the pDNI template is displayed in the **Template Data** at the bottom of the page.

4. Make the following changes to the template.

Parameter	Instruction
RemoteDirectory	Set to the target server directory that you want to scan for files to receive (like download).
RemoteCommand	Set to the command to execute to get the directory list. Both of these commands are Send Command requests. DNI fills in additional information required by these commands. Below is the sample command for UNIX (cfshnd).

Parameter	Instruction
	<pre>cfsend n:mftisnode trtype:c eb:y</pre> Below is the sample command for Windows (ftmscmd). <pre>ftmscmd /tcp /send /command /dt:e /node:mftisnode</pre>
DNICommand	Set to the command that you want to execute. Below is the sample command for Unix (cfrecv): <pre>cfrecv n:mftisnode lf:"/data/targetdir/\${RemoteFile}" rf:"\${RemoteFileName}" sm:y</pre> Below is the sample command for Windows (ftmscmd): <pre>ftmscmd /recv /file /node:mftisnode "c:\data\targetdir\\${RemoteFile}" "\${RemoteFileName}"</pre> Note: The virtual alias in both commands must be set to the virtual alias defined on the transfer definition in the step <i>Create a Transfer Definition for the defined User and Server definitions</i>.
SubDirectory	Enter Yes to transfer files in subdirectories; otherwise enter No.
SuccessAction	There are three options you can set:

Parameter	Instruction								
	<table> <tr> <th>Option</th><th>Instruction</th></tr> <tr> <td>Leave</td><td>Keep the file in the directory but do not transfer again until the file is modified.</td></tr> <tr> <td>Delete</td><td>Delete the file transferred.</td></tr> <tr> <td>Move</td><td>Move the file transferred to the location defined by the SuccessFile parameter.</td></tr> </table>	Option	Instruction	Leave	Keep the file in the directory but do not transfer again until the file is modified.	Delete	Delete the file transferred.	Move	Move the file transferred to the location defined by the SuccessFile parameter.
Option	Instruction								
Leave	Keep the file in the directory but do not transfer again until the file is modified.								
Delete	Delete the file transferred.								
Move	Move the file transferred to the location defined by the SuccessFile parameter.								
	Currently, set this file to Leave.								
NetworkErrorAction	Set to Retry for this test.								
FailureErrorAction	Set to Retry for this test.								
ScanInterval	Set to 1 (Minute).								
Template Name	Enter the name of the new template to be created.								

5. Click **Add** to add the new template.

Starting pDNI Template

After saving the new template, a list of templates is displayed and you see the new template.

To start a pDNI template, perform the following steps:

Procedure

1. On the right side of the pDNI template, click **Start**.

Result

Internet Server communicates with DNIDaemon to start the template. A message displays whether the file has been started successfully.

A start request can fail for a variety of reasons. The most common reasons are:

- An invalid parameter or invalid parameter value was defined.
- The LocalDirectory for a DNI Send was not found.
- **Post Action** SuccessFile, FailureFile, and NetworkErrorFile point to a directory that does not exist (when the **Post Action** is Move)

If the template does not start successfully, you can follow the procedure "*Verify that the pDNI template is transferring files*" to determine the cause of the failure.

Configuring MFT for LDAP Authentication

MFT Internet Server and Command Center support the following authentication methods:

Authentication Method	Protocols Supported
Database userid/password	All protocols
LDAP userid/password	All protocols
SAML	HTTP/HTTPS only
OIDC	HTTP/HTTPS only
Key/Certificate	All protocols

When MFT authenticates an LDAP user, it validates the user ID and password against the LDAP server. MFT never updates LDAP data. Therefore, MFT cannot reset a password for LDAP users and LDAP users cannot reset their LDAP password through MFT.

Data in the LDAP server is synced to the MFT database. This means that data in LDAP is compared against data in the MFT database. Users are added to the MFT DB and deleted from the MFT DB as needed; rights are added and deleted as needed.

This section describes how to configure MFT Internet Server and Command Center for LDAP Authentication.

To get the LDAP authentication working, perform the following steps:

Procedure

1. Define the LDAP authenticators.
2. Sync the LDAP authenticators to the MFT database.

Defining LDAP Authenticators

An LDAP authenticator defines the following information about the LDAP environment:

- *The Authenticator Name* The authenticator name is prefixed to the LDAP user ID to form the MFT user ID. For example, if the authenticator name is Auth1 and an LDAP user ID is User1, the MFT user ID is Auth1-User1.
- *Connectivity Information.* Defines the URL to connect to the LDAP system.
- *Credentials.* Provide access to LDAP.
- *LDAP Search*
- *LDAP Attributes*
- *Right Management*
- *Group Management*

LDAP authenticators work best when the LDAP server supports number of properties in the user object. MFT, then, uses LDAP search filters to detect MFT users and perform group right checking. Otherwise, MFT must scan LDAP groups to determine if users are MFT users and the rights that should be granted to users. LDAP search filters are much more efficient than searching groups.

Adding an LDAP Authenticator

To define the LDAP authenticators, complete the following steps:

Procedure

1. Go to **Configuration > Authenticators > Add Authenticator**.
2. Enter the required information in each authenticator property type.

Authenticator Properties: Authenticator

Property	Instruction
Name	Enter a short name of the authenticator without spaces. This name is prefixed, along with a dash, with the LDAP user ID to form the MFT user ID.

Property	Instruction
Type	Set the LDAP Server Type. If the LDAP server is not listed, set this to "other", unless the LDAP server provides emulation of Active Directory or was based on Sun Directory Server.
Server Host Name	Select the Internet Server and Command Center instances that uses this LDAP connector. Generally, all servers typically uses all LDAP authenticators.
Enabled	Select this checkbox to enable the authenticator.



Note: MFT uses the authenticator to validate incoming credentials only if the **Enable** checkbox is selected.

However, an enabled authenticator also restricts the following actions:

- Deleting users from an enabled authenticator.
- Changing any LDAP Synced Rights from an enabled authenticator.

Authenticator Properties: LDAP Connectivity

Property	Instruction
Host Name or URL(s)	Enter the host name of the target LDAP server.
Bind User DN	Enter the DN (Distinguished Name) of the user object used to log in to LDAP. This user requires read rights to LDAP; update rights are not needed.
Bind Password	Enter the password for the Bind user DN.
Confirm Password	Re-enter the password for the Bind user DN

Property	Instruction
Port	Defines the LDAP port of the LDAP server. Any port can be used but MFT defaults to these ports: 389 Non-SSL mode 636 SSL/TLS mode
Use TLS	Select this checkbox when using SSL/TLS. This changes the port to 636. If you clear the checkbox, the port is set to 389.

Authenticator Properties: LDAP Search

Property	Instruction
User Base DN	Defines the base in the LDAP tree where all users are defined. The levels searched below this base depend on the Search Scope parameter.
Sync Group DN	Defines the DN of the group that contains the users that should be synced. This must be defined only if you are not using the search filter. Search filters are more efficient and more flexible than using the Sync Group DN.
Search Filter	Defines the search filter used to search for users to sync.
Search Scope	Set this to SUBTREE_SCOPE.

Authenticator Properties: LDAP Attributes

The LDAP attributes define the LDAP fields that correspond to the MFT User table fields. These fields change depending on the LDAP Server "Type" defined.

Property	Definition
User Name	Defines the LDAP attribute used as the user name.
Full Name	Defines the LDAP attribute used as the full name.
Email Address	Defines the LDAP attribute used as the email address.
Phone Number	Defines the LDAP attribute used as the phone number.
Department	Defines the LDAP attribute used as the user's department. The department is used for delegated administration.
Usage	Defines the user type of the user. The user can be a file share user, non-file share user, or a mailbox user.
User Type	Defines if the user is a guest user, full user, or a power user. These options are enabled if the user is a file share user or a mailbox user.
Expiration Date	Defines the expiration date of the users synced using this authenticator.
Visibility	Visibility of the users synced using this authenticator can be private or public.

Authenticator Properties: Right Management

This tab defines the rights that are synced through the LDAP authenticator.

Property	Instruction
TransferRight	Assign TransferRight to all users in this authenticator. Select this checkbox if you want to assign TransferRight to all users synced through this authenticator.

Property	Instruction
Right Group Base DN	Defines the DN where the LDAP group names are located. This field is only used when the LDAP group name does NOT have an "=" in it. If there is an "=" in the LDAP group name, the LDAP group name should point to the DN of the group; this field is ignored. We generally suggest using the fully qualified DN in the LDAP group name.
Enabled	Select the appropriate Right checkbox to sync the right.
Right Name	Defines the MFT right.
LDAP Group Name	Defines the LDAP Group Name or fully qualified DN of the LDAP Group Name. We suggest using the fully qualified DN.

Authenticator Properties: Group Management

This section defines the Groups to synchronize. Unlike rights, group names are displayed only when a group is enabled for LDAP from the **Add or Update Group** page.

Property	Instruction
Group Management Base DN	Defines the base in the LDAP tree where Groups are defined. This field sets the default location for LDAP Group Names. Note: If an LDAP group name is fully qualified (contains an "=") or enclosed in parentheses, this field is ignored for that group.
Enabled	Selects the appropriate Group checkbox to sync the group.
Group Name	Defines the MFT Group name. Note: This field cannot be changed.

Property	Instruction
LDAP Group Name or Search Filter	Defines the name of the LDAP group associated with the Group Name. Note: If the LDAP Group Name contains an "=", it must be the fully qualified LDAP group DN; otherwise, the LDAP Group must be defined in the Group Management Base DN . If the LDAP group name starts with "(" and ends with ")", the data within the parentheses is treated as an LDAP Search Filter.

i Note: When a Group is not enabled, the LDAP Group Name or Search Filter resets to the default value.

3. When you have entered all the required information, click **Add** to add the LDAP authenticator.

Deleting Users from an Enabled Authenticator or Changing Rights or Groups for an Enabled Authenticator

To delete users from an enabled authenticator or to change Rights or Groups for an enabled authenticator, perform the following steps:

Procedure

1. Clear the **Enabled** checkbox and save the authenticator.
2. Enter the required changes.
3. Select the **Enabled** checkbox and save the authenticator.

Testing the LDAP Authenticator

To test the LDAP authenticator, perform the following steps:

Procedure

1. Go to **Configuration > Authenticators > Manage Authenticators**.

A list of authenticators is displayed with summary information about the authenticators.

2. Click the **Test** button to connect to the LDAP Server and view statistics about the number of users found to be synced, and the number of users in each Group where Rights or Groups are synced.

What to do next

If the test fails, a generic message is displayed. To get the details of the error, refer to the `catalina.out` file or turn on tracing for the admin user. For more information see [Debug LDAP Problems](#).

Syncing LDAP to Database

To sync LDAP to the database, perform the following steps:

Procedure

1. Go to **Administration > LDAP Sync**.

This page allows you to sync individual users or to sync all users for an authenticator.

2. To sync a single user, select **Sync user**.

You must define the user ID to sync without the authenticator name.

3. To sync all users, select **Sync All Users/Groups in these Authenticators**.

4. To sync all groups, select **Sync All Groups in these Authenticators**. Only the Groups having the LDAP Group box checkbox selected are synced.

You can sync an individual authenticator or you can sync all authenticators.

When syncing all authenticators, it might take a long time for the sync to complete. The page shows the completion percentage of the sync request.

When a Sync operation is performed, MFT writes a summary of the Sync request to the following file:

```
<MFT-Install>/logs/message/ldap_sync_report-MFT-yyyy-mm-dd.txt
```

Debug LDAP Problems

Perform the following steps if you face any issues while testing or syncing an authenticator:

- Verify the IP address and IP port. Also verify whether SSL/TLS has been enabled.
- Verify the credentials (Bind User DN and Bind Password).
- Make sure that firewall rules allow connectivity from all MFT Servers to the LDAP Server.
- Use an LDAP browser to verify the connectivity information and credentials.
- Using an LDAP browser, check the defined **Search Filter** to make sure that it is valid and returns the correct information. MFT does not validate the search filter. It is passed directly to LDAP.

When a Sync operation is performed, MFT writes a summary of the Sync request to the following file:

```
<MFT-Install>/logs/message/ldap_sync_report-MFT-yyyy-mm-dd.txt
```

This file can show some errors that occurred when an LDAP Sync is performed. For example, if there is a connectivity issue, an error message is displayed in this file.

i Note: The Manage Authenticators "Test" does not display error messages in the `ldap_sync_report` file.

If you are running an LDAP "Test", errors are displayed in one of the following locations:

- *Catalina.out* If tracing is not enabled for the user executing the test.
- *User Trace file* If tracing is enabled for the user executing the test.

Sync Users at Login

When an LDAP user logs in, the user is synced automatically. However, if this is the first time the user has synced, or if the rights have been changed, the user might not have sufficient rights to log in. Hence, the request might fail. If the user logs in again, the request should be successful.

LDAP and SSO (Single Sign-On)

MFT provides two methods of Single SignOn:

1. SAML
2. OIDC

When a user logs in through Single Sign-On, the Identity Provider (IDP) passes the MFT user ID to MFT through assertions. The user ID does not contain the authenticator name. MFT checks for a matching user ID in the following order:

1. Check for a match on the user ID in the Users DB table.
2. Check for a match on the authenticator defined in the SSO configuration. MFT prefixes the authenticator name to the user ID and search for a match.

For both SAML and OIDC, you must configure the authenticators that are searched.

 **Note:** For OIDC, the email address is typically passed to MFT in the assertions. In this case, the authenticator is not used. When using OIDC, the LDAP authenticators are only used when OIDC is configured to send the user ID in the assertions.

Both SAML and OIDC allow you to configure the LDAP authenticators that are checked when validating incoming sign-on requests. You can define:

- *No Authenticators:* No authenticators is checked
- *All Authenticators:* All authenticators is checked
- *Enabled authenticators:* Select one or more enabled authenticators

 **Note:** When checking authenticators, MFT only checks the database. It does not connect to the LDAP authenticator. Hence, you must sync the user before SSO is enabled for that user.

Configuring Single SignOn support using OIDC (OpenID Connect)

OIDC (OpenID Connect) is an authentication protocol that is built on top of OAUTH2. OIDC allows you to implement Single SignOn by providing a single set of credentials that can be used to access multiple applications.

For more information, see the following sections:

- [Different ways OIDC works with MFT](#)
- [Steps to Configure OIDC](#)

Types of Single SignOn

MFT supports two types of Single SignOn:

- SAML (Security Access Markup Language): SAML is an older technology and is more difficult to implement than OIDC.
- OIDC (OpenID Connect): OIDC is secure and simple to implement and is supported by a wide variety of OIDC Service Providers, including Azure, Google, and OKTA.

Different Ways OIDC Works With MFT

OIDC works with MFT in one of the following ways:

The user connects to a Portal. The Portal redirects the user to the OIDC log in page. After the user logs in to OIDC, they are redirected back to the Portal with a list of applications that they can use. When the user selects MFT, they are redirected to MFT along with an OIDC token. MFT validates the OIDC token, extracts the user information from the OIDC token and completes the log in. This method is often used when users can connect to multiple applications.

One of the advantages of using OIDC is that the OIDC Authentication Server can implement multi-factor authentication. Since MFT will use the OIDC Authentication Server for its log in, MFT can take advantage of the OIDC multi-factor authentication.

Steps to Configure OIDC: Configure the OIDC Authentication Server

To configure the OIDC authentication server, perform the following steps:

Procedure

1. [Construct the MFT OIDC Redirect URL.](#)
2. [Configure the OIDC Authentication Server.](#)
3. [Add the OIDC Provider.](#)
4. [Configure the OIDC information for Internet Server and Command Center instances.](#)
5. [Test out the OIDC log in.](#)

Constructing MFT OIDC Redirect URL

To configure the OIDC authentication server, perform the following steps:

Before you begin

You need to get the MFT redirect URL.

Procedure

1. Go to **Configuration > Single SignOn > Manage MFT OIDC Instances.**
2. In the **Configure MFT OIDC** box, copy the "redirect URL" parameter value. The parameter is in the format:

```
https://yourHost:port/cfcc/login/servlet/oidc/redirect
```

You need to enter the correct host name and port for your Internet Server. If you are using multiple Internet Server instances behind a load balancer, you should enter the DNS Name of the load balancer. You need this information when you configure the OIDC Authentication Server.

Configuring the OIDC Authentication Server

This step is specific to the OIDC Authentication Server that you select. Hence, it is not discussed in detail here. There are a few important things to consider:

- You must configure the MFT Redirect URL from [Step 1](#).
- You must get the "OIDC Well-Known Configuration URL" from the OIDC Authentication Server. This is used in a future step to retrieve configuration information for the OIDC Authentication Server.
- When you configure the OIDC Authentication Server, two fields are created that must be saved and entered in [Step 3](#) of this procedure:
 - Client ID
 - Client Secret

Adding the OIDC Provider

To add the OIDC provider, perform the following steps:

Procedure

1. Go to **Configuration > Single SignOn > OpenId Connect > Add OIDC Provider**.
2. There are two sections for this page.
 - Configure OIDC Authentication Server

Field	Description
Name	Defines a unique name for this OIDC Authentication Server.
Description	Allows you to enter a text description of this OIDC Authentication Server configuration.
Client ID	When you create the OIDC Authentication Server, it creates a Client ID. Copy the Client ID from prior step and enter it in this field.

Field	Description
Client Secret from the prior step	When you create the OIDC Authentication Server, it creates a Client Secret. Copy the Client Secret from prior step and enter it in this field.
Login Button Text	To display a login button for this OIDC Authentication Server, enter the button text. This is only done when users connect to MFT and MFT redirects the user to the OIDC Authentication Server.
LDAP Authenticators	If one or more of the users are Synced LDAP users, select the LDAP Authenticators where they are defined.

- Retrieve Properties From OIDC Server

In this box, you retrieve the OIDC Authentication Server configuration information. Enter the "OIDC Well-Known Configuration URL" for the OIDC Authentication Server. Some OIDC Authentication Server (example: Google) have a single URL. Others (example: OKTA) have a different URL for each OIDC Client.

3. Select the **Retrieve** link.

The OIDC configuration information is filled in.

i Note: This step assumes that the Command or Internet Server has internet access to retrieve the configuration information. If it does not, you can enter the information manually.

4. After entering the required information, click **Add**.

Configuring the OIDC Information for Internet Server and Command Center Instances

To configure the OIDC information for Internet Server and Command Center instances, perform the following steps:

Procedure

1. Go to **Configuration > Single SignOn > Manage MFT OIDC Instances**.
2. Select the server instance where OIDC should be configured.

Two tables are displayed.

Configure MFT OIDC

Field	Instruction
Description	Enter a text description of this MFT OIDC instance.
Enabled	Set to Yes
OIDC Providers	Select one or more OIDC providers for this MFT instance.
InitiatedLogin URL	Here is a sample: <pre>https://yourHost:port/cfcc/login/servlet/oidc/init</pre> You need to set the host name and port.

HTTP Proxy Properties

Enter proxy information if required.

3. After entering the required information, click **Update**.

You can create a shortcut in the <MFT-Install>/server/webapps/root folder. This shortcut can be used to redirect direct users to the OIDC Server log in page.

If you have only a single OIDC Authentication Server, then the shortcut should point directly to the InitiatedLogin URL. Example:

```
https://yourHost:port/cfcc/login/servlet/oidc/init
```

If you have more than one authentication server, you need to tell MFT which OIDC Authentication Server to use. You can do this by appending the issue URL and the Client ID to the InitiatedLogin URL. Example:


```
https://yourHost:port/cfcc/login/servlet/oidc/init?oidcredirect=browser&iss=https://accounts.google.com&mftoidcclientid=theClientIdValue
```

Testing the OIDC Log In

To test OIDC log in, perform the following steps:

Procedure

1. Enter the InitiatedLogin Url defined in [step 2](#) from a browser.
MFT redirects you to the OIDC Authentication Server.
2. Depending on the status, you might need to enter your credentials.
3. Add support for OIDC Single Log Out.

Result

The OIDC Server redirects you to the MFT page. If you are on Command Center, you are redirected to the Admin page. If you are on Internet Server, you are redirected to the Transfer page.

Configuring OFTP2 Transfers

OFTP2 (ODETTE File Transfer Protocol) is an updated version of the OFTP standard. It was defined to address the data transfer requirements of the European automotive industry.

Types of OFTP2 Transfers

MFT supports two types of OFTP2 transfers:

- Incoming OFTP2 Transfers: The OFTP2 transfer server listens on two ports (clear text and TLS) for incoming OFTP2 requests.
- Outgoing OFTP2 Transfers: Transfers to target OFTP2 servers. Outgoing transfers can use TLS or clear text ports.

OFTP2 Restrictions

While OFTP2 supports client initiated **Send and Receive** transfers, MFT only supports **Send** transfers.

- OFTP2 clients can initiate a **Send** file to MFT.
- MFT can initiate a **Send** file to a target OFTP2 server.

OFTP2 does not implement a directory list capability:

- MFT users cannot navigate through an OFTP2 directory
- There is no concept to **Create** or **Create Replace** files on target OFTP2 servers.

OFTP2 file names are virtualized.

- File Names are defined by a Virtual File Name.
- The way that the Virtual File Names are defined is up to the OFTP2 server writing the file.

When an OFTP2 client initiates a transfer to the MFT OFTP2 transfer server, the target server definition cannot be an OFTP2 server. For example, this is not supported:

OFTP2 Client > MFT OFTP2 Transfer Server > OFTP2 Server

Required Information

Before you begin to configure OFTP2, you need the following information from the OFTP2 partner:

Parameter	Description
Partner Odette ID	A string that identifies the partner.
Partner Password	The password that is validated by the partner.
TLS Support	Whether the OFTP2 request is encrypted by TLS.

The following partner public certificates must optionally be provided:

Partner Public Certificate	Description
TLS Public Certificate	The TLS certificate associated with the Partner TLS private key.
Session Authentication Public Certificate	Certificate used to authenticate the OFTP2 partner.
Encryption Public Certificate	The certificate used to encrypt data sent to the OFTP2 partner.
Signing Public Certificate	The certificate used to verify the OFTP2 data signature.
EERP Public Certificate	Certificate used by the End-to-End Response Protocol.

Securing incoming OFTP2 Requests

Incoming OFTP2 Requests are secured by the following criteria:

Criteria	Requirement
Odette ID	Must match the partner Odette ID defined in a server definition.

Criteria	Requirement
Password	Must match the partner password defined in a server definition.
TLS Certificate	Must match the OFTP2 client TLS certificate.
Session Authentication	Provides a means of authenticating OFTP2 requests.



Note: The Partner Odette ID and password are required. TLS certificate and session authentication are optional. We strongly suggest using OFTP2 session authentication and TLS client certificates (where supported) to secure OFTP2 transfers.

The section titled "Create a Server Definition for Incoming and Outgoing OFTP2 requests" has more detail about configuring these parameters.

More about OFTP2 Passwords

OFTP2 passwords are only 8 characters and are defined in the RFC as upper case only. While MFT requires OFTP2 clients to send an OFTP2 password and MFT validates the password, OFTP2 password authentication is not a good way to secure OFTP2. That is why we strongly suggest using session authentication and TLS certificate authentication. Both Local and Partner Passwords are required fields. Some OFTP2 clients and servers do not validate the incoming password. When the OFTP2 partner does not validate the incoming password, you can set the LOCAL password to any value.

OFTP2 Virtual File names

OFTP2 does not support directories and files. Rather, it supports a virtual file name. The virtual file name is only 26 bytes. Here is how MFT handles OFTP2 virtual file names.

Incoming Requests:

- MFT searches for a transfer definition where the Virtual Alias matches the OFTP2 virtual file name. If found a match, MFT uses that definition.
- If no match is found, MFT uses the first transfer definition defined for the OFTP2 user.

Outgoing Requests:

MFT sets the virtual file name to the first 26 bytes of the server file name as defined in the Transfer definition.

Steps to Configure MFT for OFTP2 Transfers

To configure MFT for incoming and outgoing OFTP2 transfers, perform the following steps: These steps are discussed in more detail later.

Procedure

1. [Create OFTP2 System Keys](#)
2. [Configure the MFT OFTP2 Transfer Server](#)
3. [Create a User Definition for incoming OFTP2 Requests](#)
4. [Create a Server Definition for Incoming and Outgoing OFTP2 requests](#)
5. [Create Transfer definitions for Incoming and Outgoing OFTP2 requests](#)
6. [Start the OFTP2 Service](#)
7. [Send Information about the MFT OFTP2 environment to the OFTP2 transfer partner](#)
8. [Executing OFTP2 transfers](#)

Creating OFTP2 System Keys

The five OFTP2 system keys are defined in the following table.

System Key	Description
TLS System Key	Used to secure the TLS connection. The TLS protocol provides encryption, so when using TLS, you typically do not need to define an encryption system key.
Authentication System Key	Used to authenticate the OFTP2 client and the OFTP2 server. This provides non-repudiation for the OFTP2 client and the OFTP2 server.
Encryption System Key	Used to decrypt data sent by the OFTP2 client. The OFTP2 client encrypts the data using the public key so that it can only be decrypted by a server

System Key	Description
	with the OFTP2 private key. Generally speaking, when using TLS, you do not need to perform OFTP2 encryption, since TLS encrypts the data.
Signing System Key	Used to sign a file. Files are signed using a system key so that any target system with the associated public key can verify the signature.
EERP System Key	Used when the OFTP2 client requests an EERP (End-to-End Response).

To create an OFTP2 system key, perform the following steps:

Procedure

1. Go to **Management > Protocol Keys > System Keys > Create Key**.

The **Create System Key** page is displayed.

2. Enter the required information described in the table below:

Parameter	Instruction
System Key Type	OFTP2 system key.
Description	Enter a unique description.
Password	Enter and confirm the system key password.
Expiration Date	Set this according to your company's standards.
Key Size	Set to 2048 bits or higher.
Signing Algorithm	Set to SHA-256, SHA-384, or SHA-512.
Set as the Default key	Select this checkbox.
Common Name	Set to IP the name of your OFTP2 server.

3. After entering these fields, click **Create Key** to create the OFTP2 system key.

MFT supports up to five OFTP2 system keys. Not all OFTP2 system keys are required. You can use the same OFTP2 key for multiple OFTP functions. If you need to create multiple

OFTP2 system keys, use the description field to define the way that the OFTP2 system key is used.

Note: When you define OFTP2 system keys, you should send the corresponding public key to the OFTP2 client.

Displaying OFTP2 Public Key

You can display the OFTP2 public key by following these steps:

Procedure

1. Go to **Management > Protocol Keys > System Keys > Manage Keys**.
2. Select the OFTP2 key.
3. Click the **Public Certificate** tab.
4. Copy and paste the OFTP2 system key and send the key to the partner.

MFT does not support the automated key exchange supported by some OFTP2 clients and servers.

Configuring MFT OFTP2 Transfer Server

To configure the MFT OFTP2 Transfer Server, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > OFTP2 Server > Configure OFTP2 Server**.
2. Select the Internet Server instance that you want to configure.
3. Enter the required information described in the table below.

Parameter	Instruction
Enabled	Set to YES.

Parameter	Instruction
IP Port	Defines the listening port for incoming OFTP2 non-TLS requests. We do not recommend using the clear text IP Port. Using the TLS port provides a higher level of security than the clear test IP port, even though data can be encrypted on the clear text IP port.
TLS Port	Defines the listening port for incoming OFTP2 TLS requests. We strongly recommend using TLS for all incoming OFTP2 connections.
Require Client Certificate	Defines whether clients initiating TLS requests require a certificate. It is a good practice to require client certificates. The client certificate can then be validated against the TLS certificate defined in the server definition.
OFTP2 System Key	Defines the OFTP2 system key used by the OFTP2 TLS listener.

- When you have completed configuring the fields, click **Update** to save your changes.

Note: Once the OFTP2 transfer server is configured, you must configure the OFTP2 server definitions, OFTP2 users, and the OFTP2 transfer definitions. When these are complete, you can start the OFTP2 server. See [Starting OFTP2 Service](#) for instructions on starting the OFTP2 transfer server.

Create a User Definition for incoming OFTP2 Requests

This user definition is used for incoming OFTP2 transfer requests. The server definition defines a user that is used when transfers are received from an OFTP2 client. This user is defined on the transfer definitions to point to a target Server such as an SFTP or Platform Server.

Note: A separate OFTP2 user ID must be created for each incoming partner OFTP2 server.

Procedure

1. Go to **Partners > Users > Add User**.

The **Add User** page is displayed.

2. Enter the required information described in the table below:

Parameter	Instruction
User Id	Enter a unique user ID.
Full Name	Enter any full name for this user.
Password	Enter any characters. OFTP2 Transfers do not use the password defined in the user definition.
Usage	Non-FileShare user.

3. In the **Rights and Groups** tab, ensure **TransferRight** is assigned to the user.
4. In the **Optional User Properties** tab, set the properties described in the table below:

Property	Instruction
Client Protocols Allowed	Set to OFTP2.
Change Password at Next Login	Clear this checkbox.

5. After entering these fields, click **Add** to create the user.

Create a Server Definition for Incoming and Outgoing OFTP2 Requests

For all protocols except AS2 and OFTP2, server definitions are only used for outgoing requests. For OFTP2, however, server definitions are used for incoming and outgoing OFTP2 requests.

Incoming Requests

- The transfer partner initiates a connection request to MFT and specifies its Odette ID.
- MFT searches the server definitions for a matching Partner Odette ID.
- When it finds the matching Odette Partner ID, it gets the user ID from the server definition.
- MFT then searches for transfer definitions for the user ID defined in the server definition.
- MFT checks if the OFTP2 virtual file is a match on the Virtual Alias.
- The Transfer Definition points to a target Server such as SFTP or Platform Server. This is the server where the incoming OFTP2 data is saved.

Outgoing Requests

- A client (SFTP, Platform Server, HTTPS) logs in and initiates a transfer request to MFT.
- The client specifies a remote file name whose first parameter is the Virtual Alias.
- MFT Searches the transfer definitions for that user for and upload definition with a match on the Virtual Alias
- The transfer definition selected points to an OFTP2 Server definition
- The data received from the transfer client is sent to the OFTP2 server.

Creating an OFTP2 Server Definition

To create an OFTP2 server definition, perform the following steps:

Procedure

1. Go to **Partners > Servers > Add Server**.

The **Add Server** page is displayed.

2. Enter the required server information described in the table below:

Information	Instruction
Server Name	Enter a unique server name.
IP Address	IP name or IP address provided by the OFTP2 transfer partner.
IP Port	Enter the port the OFTP2 transfer partner is listening on.
Server Type	Set to OFTP2 .
Server Type	Set to UNIX .

Server Credentials

Server credentials are not used for OFTP2 transfers.

OFTP2 Options: General Information

The following table lists the options under General Information.

Option	Instruction
Local Odette ID	Enter the Odette ID for your system.
Local Password	The password sent to the OFTP2 partner. It is up to the partner whether this password is authenticated. Not all OFTP2 software validates the password.
Partner Odette ID	Enter the Odette ID for your OFTP2 transfer partner. The transfer partner must provide this information.
Partner Password	The password sent by the OFTP2 partner. MFT validates this password for incoming and outgoing requests.
UserId for incoming requests	Select the user that you created in the prior step. This user is used for incoming OFTP2 requests.

OFTP2 Options: Outgoing Parameters

Outgoing parameters are used when MFT initiates an OFTP2 transfer to a target OFTP2 server.

Option	Instruction
Use TLS	Defines whether TLS is used for outgoing connections. Using TLS is the most secure way of performing OFTP2 transfers. We strongly suggest using TLS for all OFTP2 transfers.
Session Authentication	Defines whether the MFT OFTP2 client requests session authentication.

OFTP2 Options: Incoming Parameters

Incoming Parameters are used when OFTP2 client initiates a transfer to the MFT OFTP2 server.

Option	Instruction
Require Session Authentication	Defines whether Session Authentication is required. If set to "Yes", the OFTP2 client must request Session Authentication.

OFTP2 Options: Sending Files

Sending File parameters are used when MFT sends a file to a target OFTP2 server.

Option	Instruction
Sign Files	Defines whether files are signed by the OFTP2 Signing System Key. Files are signed with the MFT OFTP2 Signing System Key. Anyone with the MFT OFTP2 Signing Public Key can validate the signature.
Encrypt Files	Defines whether files are encrypted by the OFTP2 Encryption System Key. If TLS is not enabled, you must set this to YES. OFTP2 data is encrypted using the OFTP2 partner's public encryption key so

Option	Instruction
	that only a partner with the OFTP2 system key can decrypt the data.
Request EERP	Defines whether an EERP is required.
Compress Files	Defines whether data is compressed. MFT uses the ZLIB compression algorithm to compress data.

OFTP2 Options: Receiving Files

Receiving File parameters are used when an OFTP2 client sends a file to the MFT OFTP2 server.

Option	Instruction
Require Sign Files	Defines whether incoming files must be signed.
Require Encrypted	Defines whether incoming files must be encrypted. While encryption can be done by the TLS protocol, this parameter defines whether OFTP2 encryption is required. As discussed before, if you are using TLS, you do not need to define OFTP2 encryption. To ensure that TLS is used, we recommend only defining the TLS Port and not defining the clear text IP port for the OFTP2 transfer server. This is discussed in the section Configure the MFT OFTP2 Transfer Server .
EERP Receipt Delivery	Currently, only Sync EERP is supported.

OFTP2 Options: System Keys

System Keys are used for various OFTP2 functions. Note that you can use the same OFTP2 system key for multiple functions. System keys are only required when the corresponding OFTP2 function is enabled. For example, if you are using TLS and do not enable encryption, you do not need to define the encryption system key.

Option	Instruction
TLS System Key	Defines the MFT System Key used for TLS.
Authentication System Key	Defines the MFT system key used for OFTP2 session authentication.
Encryption System Key	Defines the MFT system key used for OFTP2 encryption.
Signing System Key	Defines the MFT system key used for OFTP2 signing.
EERP System Key	Defines the MFT system key used for OFTP2 EERP.

The public key for these system keys should be sent to the OFTP2 transfer partner. See the section titled "Create OFTP2 System Keys" for information on viewing the public keys associated with system keys.

OFTP2 Options: Partner Public Certificates

Partner public certificates are provided by the OFTP2 transfer partner. They are associated with the OFTP2 system keys defined by the transfer partner. Partner public certificates are used for various OFTP2 functions and are only required when the corresponding OFTP2 function is enabled. For example, if you are using TLS and do not enable encryption, you do not need to define the Encryption Public Certificate.

Option	Instruction
TLS Public Certificate	Defines the Partner Public Certificate used for TLS.
Authentication Public Certificate	Defines the Partner Public Certificate used for OFTP2 session authentication.
Encryption Public Certificate	Defines the Partner Public Certificate used for OFTP2 encryption.
Signing Public Certificate	Defines the Partner Public Certificate used for OFTP2 signing.
EERP Public Certificate	Defines the Partner Public Certificate used for OFTP2 EERP.

Create Transfer definitions for Incoming and Outgoing OFTP2 Requests

Transfer definitions are required to perform all Internet Server transfers. The transfer definitions are slightly different based on whether the request is an incoming OFTP2 request initiated by an OFTP2 client, or an outgoing OFTP2 request initiated by Internet Server to an OFTP2 server.

Incoming OFTP2 request initiated by an OFTP2 client

Example: **OFTP2 Client > MFT > SSH Server**

To create an incoming OFTP2 request initiated by an OFTP2 client, perform the following steps:

Procedure

1. Go to **Transfers > Internet Server Transfers > Add Transfer**.
2. Enter the required information described in the table below:

Parameter	Instruction
Client File Name	Set to any value.
Server File Name	Set to the desired Server file name. Note that the <code>ClientFileName</code> token refers to the Virtual File Name defined by the OFTP2 client and might not be a valid file name. You can use tokens to define the file name for the target server.
Directory Transfer	Can be set to Yes or No.
Description	Set to a description that describes the transfer request.
Authorized User Id	Select the OFTP2 user configured in the server definition > OFTP2 Options > User ID for incoming request . This user ID was defined in

Parameter	Instruction
	a prior step.
Authorized Group Id	Leave this blank.
Server Name	Select the target server where the file is saved. This is typically an SFTP server or a Platform Server, although this could be any server type.
Transfer direction	Set to Upload to Server. Downloads are not supported for OFTP2 transfers.
Client Protocols Allowed	Set to OFTP2.
Department	Set as required.
Virtual Alias	If you want the client OFTP2 virtual file to be compared to the Virtual Alias in more than one transfer definition, you can enter the Virtual Alias. Otherwise, the first transfer definition for the OFTP2 user is selected.

3. Enter any other transfer parameters as required.
4. When you have entered all of the necessary parameters, click **Add** to create the transfer definition.

Outgoing OFTP2 request initiated by Internet Server to an OFTP2 Server:

Example: **SSH Client > MFT > OFTP2 Server**

To create an outgoing OFTP2 request initiated by an OFTP2 client, perform the following steps:

Procedure

1. Go to **Transfers > Internet Server Transfers > Add Transfer**.

2. Enter the required information described in the table below:

Parameter	Instruction
Client File Name	Set to any value.
Server File Name	Set to the desired server file name. Since OFTP2 only supports a 26-byte virtual file name, MFT truncates the file name to the first 26 bytes.
Directory Transfer	Set to Yes .
Description	Set to a description that describes the transfer request.
Authorized User Id	Select the user that initiates the file transfer request.
Authorized Group Id	Leave this blank.
Server Name	Select the target server where the file is sent.
Transfer direction	Set to Upload to Server. Downloads are not supported for OFTP2 transfers.
Client Protocols Allowed	Set to All.
Department	Set as required.
Virtual Alias	Set to the Virtual Alias used by the MFT transfer client.

3. Enter any other transfer parameters as required.
4. When you have entered all of the necessary parameters, click **Add** to create the transfer definition.

Starting OFTP2 Service

To start the OFTP2 service, perform the following steps:

Procedure

1. Go to **Administration > Transfer Servers > OFTP2 Server > OFTP2 Server Status**.
2. Select the OFTP2 server host.
3. Click **Server Status** to get the current status of the OFTP2 service on this host.
4. Click **Start Server** to start the OFTP2 service on this host.
5. Click **Stop Server** to stop the OFTP2 service on this host.

Result

Once the OFTP2 service is started, FT accepts incoming OFTP2 transfers.

Send Information about the MFT OFTP2 Environment to OFTP2 Transfer Partner

The following table lists the MFT OFTP2 environment parameter information sent to the OFTP2 transfer partner.

Parameter	Description
Local Odette PartnerID	The MFT OFTP2 partner ID defined in the server definition.
Local Password	The OFTP2 password that is sent to the OFTP2 partner.
TLS Public Certificate	The public key associated with the OFTP2 TLS system key configured in the Server definition. You can get this information from Management > Protocol Keys > System Keys > Manage System Keys . Select the system key. The public key is in the Public Certificate tab.

Parameter	Description
Encryption Public Certificate	The public key associated with the OFTP2 "Encryption System Key" configured in the server definition. You can get this information from Management > Protocol Keys > System Keys > Manage System Keys . Select the system key. The public key is in the Public Certificate tab.
Session Authentication Public Certificate	The Public Key associated with the OFTP2 "Authentication System Key" configured in the Server definition. You can get this information from Management > Protocol Keys > System Keys > Manage System Keys . Select the system key. The public key is in the Public Certificate tab.
Signing Public Certificate	The Public Key associated with the OFTP2 "Signing System Key" configured in the server definition. You can get this information from Management > Protocol Keys > System Keys > Manage System Keys . Select the system key. The public key is in the Public Certificate tab.
EERP Public Certificate	The Public Key associated with the OFTP2 "EERP System Key" configured in the Server definition. You can get this information from Management > Protocol Keys > System Keys > Manage System Keys . Select the system key. The public key is in the Public Certificate tab.

Executing OFTP2 Transfers

OFTP2 clients can now initiate transfers to MFT Internet Server to a target server

- An OFTP2 client initiates a transfer to the OFTP2 server port.
- If using TLS, MFT optionally requests and validates a TLS client certificate.
- MFT extracts the partner's Odette ID.
- MFT matches the Odette ID against defined server definition's partner Odette ID.
- MFT validates the password sent by the OFTP2 client.
- If defined, MFT performs session authentication with the OFTP2 client.
- The server definition defines the user used for file transfers with this OFTP2 partner.
- The user definition is used to search for the **Upload Transfer** definition.

- The transfer definition points to the target Server where the file is saved.
- As the OFTP2 client sends data to MFT, the data is streamed to the target Server (i.e. SFTP).
- If a signature is required, MFT verifies the signature sent by the client.

MFT Clients (HTTP, Platform Server, SFTP...) can initiate transfers to send to an OFTP2 server.

- An SFTP client initiates a transfer to MFT.
- Credentials (key or password) are used to log in to a user.
- The SFTP client defines a Virtual Alias and a file name.
- The Virtual Alias is used to select a transfer definition.
- The transfer definition points to an OFTP2 server.
- MFT connects to the target IP name and IP port defined.
- If using TLS, MFT validates the server's TLS certificate.
- MFT verifies the partner's Odette ID.
- MFT validates the password sent by the OFTP2 client.
- If defined, MFT performs session authentication with the OFTP2 client.
- As the SFTP client sends data to MFT, the data is streamed to the OFTP2 server.
- As MFT is sending data, it optionally signs the data.

Configuring MFT for SAML SSO

TIBCO MFT Internet Server and TIBCO MFT Command Center support Single Sign On (SSO) when using SAML (Security Assertion Markup Language). When using SAML for SSO, TIBCO MFT Internet Server and TIBCO MFT Command Center perform the role of service provider (SP).

Before you begin

You must install and configure a SAML identity provider (IdP) before configuring SAML for the MFT server.

 **Note:** Each SAML implementation is different and often requires significant work to integrate MFT into the SAML infrastructure. Typical SAML implementations will require TIBCO Professional Services to work in conjunction with your SAML support staff to ensure a smooth SAML implementation.

To configure TIBCO MFT Internet Server and TIBCO MFT Command Center SAML integration, you must perform the following operations:

1. [Creating SAML Private Keys](#)
2. [Importing SAML Identity Provider Metadata](#)
3. [Configuring SAML Service Provider Metadata](#)
4. [Generating SAML Service Provider Metadata](#)
5. [Sending SAML Service Provider Metadata to the Identity Provider](#)
6. [Restarting the MFT Server](#)
7. [Updating MFT Shortcuts](#)

SAML is configured on a server by server basis. Each MFT server that needs to use SAML must be configured independently of the other MFT servers.

For detailed descriptions of individual SAML fields, see the help information for the SAML administrator pages.

Note: After the SAML configuration is updated, you must restart the MFT Server. The SAML information is loaded at startup time and cannot be refreshed.

Creating SAML Private Keys

You can create SAML private keys through the **Administration > Protocol Keys > System Keys > Create Key** option.

The following figure shows the Create System Key page:

On this page, set the **System Key Type** field to SAML System Key, enter the required information, and then click **Create Key**.

After the SAML system key is created, you can reference this key on the Configure SAML Service Provider MetaData page.

As an alternative, you can import a SAML key from a JAVA keystore through the **Administration > Protocol Keys > System Keys > Import Key** option.

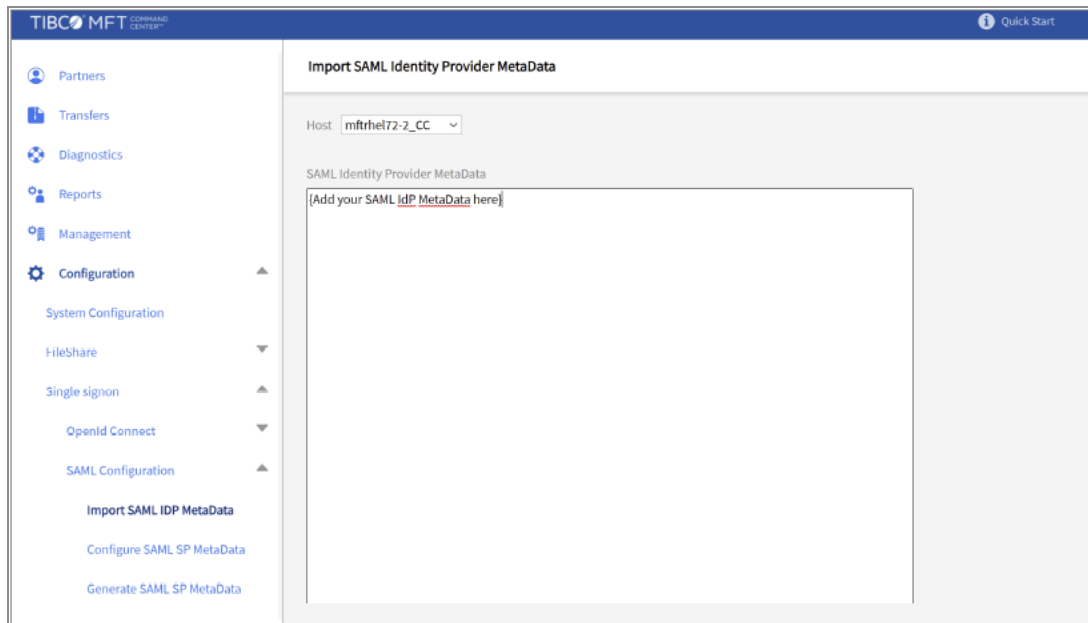
Importing SAML Identity Provider Metadata

You can import SAML identity provider metadata through the **Administration > SAML > Import SAML IDP MetaData** option.

The identity provider will provide the metadata that must be imported into MFT. The identity provider metadata is typically distributed in a file and consists of XML that describes the identity provider. It typically contains the following information:

- X.509 certificates used to sign and encrypt SAML data
- Single Sign On and Single Log Out end points

The following figure shows the Import SAML Identity Provider MetaData page:



Paste the data in the identity provider metadata into this page, and then click **Import**. MFT will validate that the data is in a proper XML format and contains valid identity provider data.

Configuring SAML Service Provider Metadata

You can configure SAML service provider metadata through the **Administration > SAML > Configure SAML SP MetaData** option.

The following figure shows the Configure SAML Service Provider MetaData page:

This page configures the following MFT SAML attributes:

Parameter	Description
Enabled	Defines whether SAML should be enabled (Yes) or disabled (No)
Service Provider Id	Defines the SAML service provider name. Note: It must be unique across all SP servers in the SAML environment.
SAML User Id Attribute	Defines the SAML attribute that MFT will use as the user ID.
SAML Host URL	Defines the URL of the MFT server.
SAML Encrypt Key	Defines the SAML system key that will be used to encrypt SAML messages.

Parameter	Description
SAML Sign Key	Defines the SAML system key that will be used to sign SAML messages.
LDAP Authenticators	Defines the LDAP authenticators that will be scanned for a match on the SAML user ID. You can select multiple authenticators that will be scanned for matches on the user ID.

When a successful SAML authentication occurs, MFT will extract the user ID from the SAML attribute defined by the **SAML User Id Attribute** field. If this user is defined by an MFT LDAP authenticator, MFT needs to determine which authenticator defines the user ID.

For example, assume that two LDAP authenticators (Customer and Internal) have been defined and the user acctuser has been authenticated by SAML. MFT will perform the following checking. The first match defines the user ID used for the session.

- Search the database for a match on the user acctuser.
- Search the database for a match on Customer-acctuser.
- Search the database for a match on Internal-acctuser.

Note: You must make sure that a user ID defined by SAML is unique within all authenticators defined.

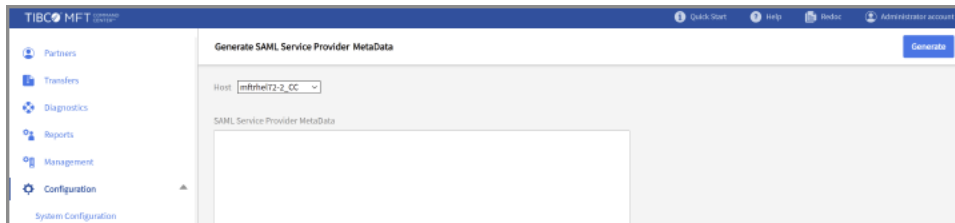
After entering the necessary information, click **Update** to update the database.

Generating SAML Service Provider Metadata

You can generate SAML service provider metadata through the **Administration > SAML > Generate SAML SP MetaData** option.

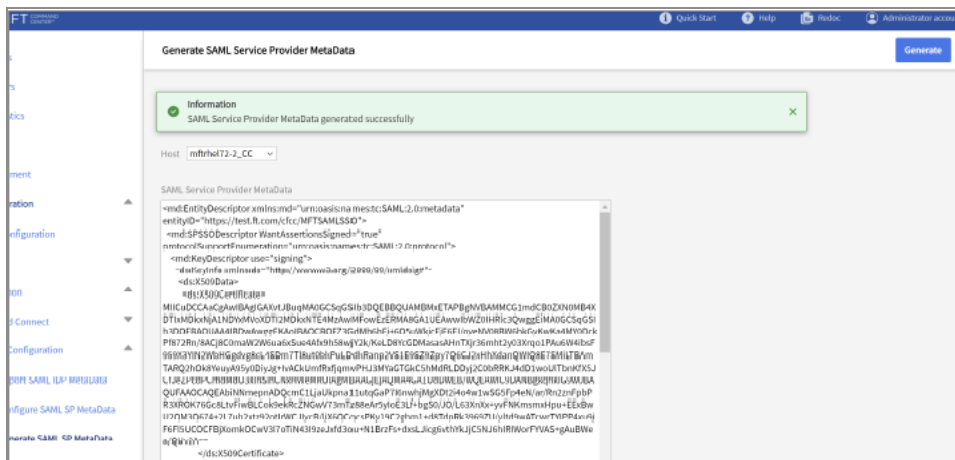
Note: Before generating SAML service provider metadata, you must configure SAML service provider metadata on the Configure SAML SP MetaData page.

The following figure shows the Generate SAML Service Provider MetaData page:



Click **Generate** to generate the service provider metadata. A text box that contains the service provider metadata is displayed. This information must then be sent to the SAML identity provider.

The following figure shows sample SAML metadata:



Sending SAML Service Provider Metadata to the Identity Provider

After generating the SAML service provider metadata, you can send the metadata to the identity provider.

Copy and paste the data information generated in [Generating SAML Service Provider Metadata](#) into a file, save the file, and send the file to the identity provider.

Optionally, depending on the requirements of the identity provider, you might need to send the service provider metadata as text in an email.

Restarting the MFT Server

When configuring TIBCO MFT Internet Server and TIBCO MFT Command Center SAML integration, in some conditions, you must restart the MFT server.

In the following conditions, you must restart the MFT server:

- When you import new identity provider metadata.
- When the security provider configuration is changed.

Updating MFT Shortcuts

You can update the MFT shortcuts to redirect users to the SAML login pages.

The following shortcuts are located in the `<MFT Install>\servers\webapps\ROOT` directory:

- `samladmin`: redirects you to the administrator page after SAML authentication is completed.
- `samlbrowser`: redirects you to the FT Browser page after SAML authentication is completed.

You can use these shortcuts or rename them to names you choose. When the user goes to one of these pages, they will be redirected to SAML for authentication. When authentication is completed, the user will be redirected to the page defined by the shortcuts. If you change the context from the default of `cfcc`, you must change the context in these files.

The following shortcuts are located in the `<MFT Install>\servers\webapps\context\login` directory:

- `ssoadmin`: redirects you to the administrator page after SAML authentication is completed.
- `ssobrowser`: redirects you to the FT Browser page after SAML authentication is completed.

These file names are hardcoded in the MFT code. When the user is authenticated by SAML, the user generally specifies the client that they want to use. When authentication is completed, the user will be redirected to the desired client based on the URLs in these files. If you change the context from the default of `cfcc`, you must change the context in

these files.

i Note: If the installation context is not "cfcc", you must update the samladmin, samlbrowser, ssoadmin, and ssobrowser shortcuts to point to the correct context.

Configuring MFT for Multi Factor Authentication

TIBCO MFT Internet Server and TIBCO MFT Command Center support two methods of Multi-Factor Authentication (MFA):

- **Email:** Users receive a one-time password via email.
- **Time-based One-Time Password (TOTP):** Users generate a one-time password using an authenticator application, such as Google Authenticator or Microsoft Authenticator.

The following sections guide you through configuring MFA for some or all of your MFT instances.

1. [Common MFA Configuration](#)
2. [Manage MFT MFA Instances](#)

Common MFA Configuration

Use the Common MFA Configuration page to define the MFA configuration parameters that are shared across all MFT instances.

To access this page, navigate to **Configuration > Multi-Factor Authentication > Common MFA Configuration**.

You require the **AdministratorRight** to view and modify the Common MFA Configuration settings.

Updating Common MFA Configuration

As an Administrator, you can update Email MFA parameters, TOTP MFA parameters and parameters that are common to all the MFA methods described in the Configuring MFT for MFA section.

To update the Common MFA Configuration, perform the following steps:

Procedure

1. Navigate to the **Common MFA Configuration** page.

The following screenshot shows the Common MFA Configuration page.

Common MFA Configuration Update

Email Multi factor Authentication

Email PassCode Length: (4 to 8 character))

Email PassCode expiration: Minutes

Email PassCodes per session:

TOTP Multi factor Authentication

Allow Users to Re-Register: ☐ Yes ☒ No

Common MFA parameters

PassCode Try Count: (0-9, 0 means no limit)

MFA Excluded Users and IP Addresses:

Values separated by comma or new line
IP Addresses should be enclosed in []

MFA For FTP/SFTP Logins: ☐ Enabled ☒ Disabled

MFA For OIDC/SAML: ☐ Enabled ☒ Disabled

2. To enable Email MFA, update the parameters in the **Email Multi factor Authentication** section.
3. To enable TOTP MFA, update the parameters in the **TOTP Multi factor Authentication** section.
4. Update any parameters in the **Common MFA parameters** section that apply to all MFA methods.
5. Click the **Update** button on the top right corner of the screen to save your changes.

Manage MFT MFA Instances

Use the Manage MFT MFA Instances page to configure the MFA for TIBCO MFT Internet Server and TIBCO MFT Command Center instances

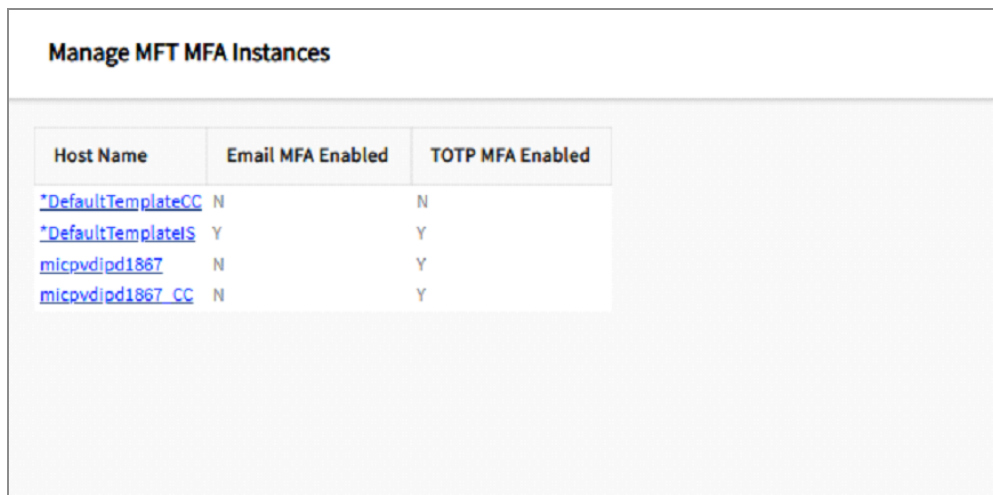
To access this page, navigate to **Configuration > Multi-Factor Authentication > Manage MFT MFA Instances**.

All the Internet Server and Command Center instances, along with the default templates are displayed in the Results Table.

Results Table

Results Table lists all the MFT instances that are configured for MFA. To modify the settings for a specific instance, click its Host Name in the table. A detailed page is displayed allowing you to update the entry if you have the necessary authorization.

The following screenshot shows the result table.



Host Name	Email MFA Enabled	TOTP MFA Enabled
*DefaultTemplateCC	N	N
*DefaultTemplateIS	Y	Y
micpvdipd1867	N	Y
micpvdipd1867_CC	N	Y

Updating an MFT MFA Instance

To update an MFT MFA Instance, perform the following steps:

Procedure

1. From the Results Table on the **Manage MFT MFA Instances** page, click the **Host Name** of the instance you wish to update.

The **Update MFT MFA Instance** page is displayed. The following screenshot shows this page.

The screenshot shows a web interface for updating MFT MFA settings. At the top, the title is 'Update MFT MFA Instance | micpvdipd1867_CC'. To the right of the title are two buttons: 'Back to MFT MFA Instances List' and 'Update'. Below the title, the section is titled 'Configure MFT MFA'. It contains three rows of configuration options:

Configuration Option	Value
Host Name	micpvdipd1867_CC
Email MFA Enabled	☐ Yes ☒ No
TOTP MFA Enabled	☒ Yes ☐ No

2. On the **Update MFT MFA Instance** page, enable or disable the Email or TOTP MFA methods for the selected host.
3. Click **Update** in the upper right corner to apply the changes.

Diagnosing and Debugging Problems

MFT provides a variety of ways to debug the product. This section describes the following three debugging features of MFT:

- MFT Admin Diagnostic or Debugging Pages
- Tracing
- Common Debugging Scenarios

MFT Admin Diagnostic or Debugging Pages

Diagnostics page

When you open a case with TIBCO support, TIBCO support engineers typically ask you for the diagnostics page.

Procedure

1. Go to **Diagnostics > Diagnostics**.
2. Select the required Internet Server or Command Center instance.
The Diagnostics information is displayed for that server.
3. Click **Save Server Diagnostics to File**.

The **Diagnostics** page is downloaded to your computer.

The following information is displayed in the **Diagnostics** page:

- Version and Hotfix information
- The directory where MFT is installed
- The number of active DB Connections
- JVM Memory usage
- Whether FIPS mode is enabled
- Summary information about active transfers
- The current time on the MFT instance

- Traces set
- JVM System Properties, including the Java version that is used
- Environment Variables
- SSL Cipher Suites
- web.xml Context Parameters
- All files in the WEB-INF/lib directory

Search Audits page

MFT creates an audit record for each transfer completed either successfully or unsuccessfully. Note that for an audit record to be written, the transfer must have started. If a transfer terminates before the transfer starts, an audit record is not written. For example, audit records are not written in the following circumstances:

- An SCP request initiates a transfer but the MFT credentials are not valid and the log in fails.
- A Platform Transfer initiates a request, but the Virtual Alias does not exist for the denied user. In this case, the Platform Server Client writes an audit record, but Internet Server does not write an audit record.

In addition to details about the transfer that was completed, the **Audit Detail** page also contains the following information:

- The status of the transfer
- The date or time the transfer started and completed
- Byte counts
- The Internet Server host where the transfer executed
- Descriptive error messages when the transfer completed unsuccessfully
- The client user that initiated the transfer
- Client and server protocols used
- Client and server ciphers used
- Client and server PGP information

Search Alerts page

MFT creates an alert audit record for each alert that has been triggered. This alert record contains the following information:

- The alert definition that triggered the alert
- Information about the criteria that initiated the alert
- Transfer information
- Login information
- The date or time the alert was triggered
- The alert actions that executed
- The status of the alert actions executed (successful or failed)

Error Events page

The error event pages include information about directory list requests that have failed. When a browser or FTP/SFTP client accesses a Virtual Alias and requests a directory list, if the directory list fails, no files are displayed. When a Platform Server client requests a directory list (through a DNI Receive or a Directory Receive request), if an error occurs, no file names are returned.

MFT does not display the cause of the failure to the clients. For information about the cause of the failure and a descriptive error message, see the **Error Events** page.

Procedure

1. Go to **Diagnostics > Error Events**.
A list of error events for the current day is displayed.
2. Use the **Search Criteria** to filter the records returned.
3. Click **Error Id** to get detailed information about the error.

Events Page

The events page displays information about events that have executed. The following event types are displayed:

- Command Center initiated Platform Server transfers
- Scheduler Jobs that have executed
- JMS initiated Internet Server and Platform Server transfers
- File Share requests

Procedure

1. Go to **Diagnostics > Search Events**.
A list of events for the current day is displayed.
2. Use the **Search Criteria** to filter the records returned.
3. Click **Error Id** to get detailed information about the error.

You can use this page to track requests. You can also use it to determine if scheduler jobs are executed and whether they executed successfully.

Server Status Page

The **Server Status** page displays the connectivity status of monitored servers. When you define a server definition, you define whether the server should be monitored by the **Server Status** page. Click the server definition **Management Options** tab. The **Check Server Status** parameter defines whether this server definition must be monitored. You can also select the Command Center or Internet Server instance that must initiate the monitor requests through the **Check Server Status On** parameter.

Procedure

1. Go to **Diagnostics > Server Status > Server Status**.
The results table displays a record for each monitored server. The status column shows the status of the connection to the server.
2. Select the required **Server Name**.
The **Server Status Detail** page is displayed. This page shows detailed information about the current status and historical information about this server.

i Note: Similar information is displayed in the Summary dashboard in a graphical representation. The summary dashboard is discussed below.

Host Status Page

The Host Status page displays the current status of each defined Internet Server and Command Center instance. The following information is displayed for each server:

Parameter	Description
CPU Usage	The CPU % used by the Internet Server or Command Center host.

Parameter	Description
Threads	The number of threads used by the Internet Server or Command Center host.
Transfers executing now	The number of transfers executing on the Internet Server host. Since Command Center does not execute transfers, this line chart is not displayed for Command Center.

Procedure

1. Go to **Diagnostics > Server Status > Host Status**.

Result

This information is refreshed every 10 seconds by default. You can click the three bars on a chart and a line chart is displayed that shows how the CPU, memory or transfers change over each scan interval.

Admin Changes Page

The Admin Change page keeps track of all administrative changes that have been made. The following changes are detected and saved in the database:

Component	Change
Create	A component ID was created.
Update	A component ID was updated.
Delete	A component ID was deleted.
Start	A component ID was started. This includes Internet Server and Command Center services. Starting the Internet Server and Command Center servers are also captured.
Stop	A component ID was stopped. This includes Internet Server and Command Center services. Stopping the Internet Server and Command Center servers are also captured.

Component	Change
Hold	The Command Center Scheduler service was put on hold.
PS Delete	A Platform Server Transfer was cancelled.
PS Update	This is reserved for future enhancements.

The changes are logged whether initiated by the admin pages, the CLI or through REST calls.

Procedure

1. Go to **Configuration > Admin Changes > Search Admin Changes**.

The results table shows all of the admin changes made today. You can define selection criteria to filter the information displayed.

2. Click the **ID** field to display the **Admin Change Detail** page.

This page displays the following information:

- Component changed.
- Date and time the change was made.
- User that made the change.
- IP Address that initiated the change.
- The MFT Server where the change was made.
- The component type and ID of the change.
- For Update requests, the old and new values are displayed.
- For Delete requests, the old value is displayed.

Dashboard

The dashboard pages provide information about the MFT Servers and Internet Server and Platform Server transfers. There are two types of dashboards:

Summary Dashboards

Display Server Status information and dashboards about transfers today, this week and this month.

Procedure

1. Go to **Reports > Dashboards > Summary**.

Four dashboards are displayed:

Summary Dashboards

Dashboard	Description
Server Status	This shows a graphical representation of the Server Status page (See above). You can click a slice of the pie chart to get detailed information about the servers with that status.
Internet Transfers today	The pie chart shows the successful and failed Internet Server transfers today.
Internet Transfers this week.	The pie chart shows the successful and failed Internet Server transfers today and the prior 6 days.
Internet Transfers this month	The pie chart shows the successful and failed Internet Server transfers today and the prior 29 days.

This page gives you a good representation of the current health of the MFT system.

Transfer Dashboards

Display dashboards for Internet Server and Platform Server transfers. You can select the dashboard "From Date" and "To Date". You can also select whether you want to display Internet Server or Platform Server transfers.

Internet Server Transfers and Platform Server Transfers provide different views of the data. The Transfer Dashboard help page has detailed information on the different views:

Internet Server Transfer Views

View	Description
All Transfers by Host	Lists transfers by Internet Server Hosts
All Transfers by Protocol	Lists transfers by Internet Server target Server protocol
Failed Transfers by Host	Lists failed transfers by Internet Server Hosts
All Transfers by Protocol	Lists failed transfers by Internet Server target Server protocol
Transfers by Target Server	Lists transfers by target server.

Platform Server Transfer Views

View	Description
All Platform Server Transfers	Lists transfers by MFT Server definitions.
Failed Platform Server Transfers	Lists Failed Platform Server transfers by date

This dashboard is a good way for users to spot failure trends. You can quickly change the view to see which protocols or servers have the most failures.

Active Transfers

Active Transfers

Command Center has two pages that display active transfers.

Active Transfer	Description
Internet Server	Displays active Internet Server transfers.
Platform Server	Displays active Platform Server transfers.

Internet Server Active Transfers

To initiate Internet Server active transfers, perform the following steps:

Procedure

1. Go to **Administration > Active Transfers > Internet Server Transfers**.

Result

A dashboard is displayed with information about the active transfers running on each Internet Server. Below that is a table that shows summary information about active transfers running on all servers. If you click an individual server, summary information is displayed for transfers running on that server.

The information in this dashboard is updated approximately every 30 seconds.

Platform Server Active Transfers

To initiate Platform Server active transfers, perform the following steps:

Procedure

1. Go to **Administration > Active Transfers > Platform Server Transfers**.

Result

In order to see Active Platform Transfers, you must select the Platform Server transfer from the **Remote Platform Server** dropdown box. After selecting the server, a list of active transfers is displayed. For Platform Server for z/OS, active, inactive, and held transfers are also displayed.

 **Note:** Platform Server must be at the V8.1.0 level or higher to support Active Platform Server transfers.

If the Platform Server supports Alter, you can cancel active transfers.

 **Note:** To see Active Transfers, the Command Center user must have **FTAdminRight**. To cancel active transfers, the Command Center user must have **FTAdminAlterRight**.

MFT Tracing

MFT supports various types of tracing.

Tracing	Description
Login Tracing	Helps debug log in problems
SSH Tracing	Helps debug SFTP transfer problems
JMS Tracing	Helps debug JMS issues
MFT Tracing	Helps debug Admin or file transfer issues
Other types of tracing	

Login Tracing

Login tracing is used when a user has trouble logging in to MFT. This tracing can be helpful for either debugging key or certificate authentication issues, or, SAML single sign-on or OIDC single sign-on issues.

To enable log in tracing, perform the following steps:

Procedure

1. Go to **Configuration > System Configuration**.
2. Click the **Server Settings** tab.
3. Select the MFT host where you want to enable log in tracing.
4. Set the **Logon Trace Level** to "All Messages".
5. All trace data are written to a file in the logs or trace directory:

```
login-trace-MFT-app-YYYY-MM-DD.txt
```

For more information about the format of the login-audit trace file, see [KB article: 000035701](#).

SSH Tracing

SSH tracing helps you to debug SSH protocol and SSH Transfer issues. SSH Tracing must be enabled only at the direction of TIBCO Technical Support. There are two types of SSH Tracing:

SSH Tracing	Description
SSH Client	Traces SSH connections when MFT is acting as an SSH client and connecting to target SSH servers.
SSH Server	Traces SSH connections when SSH clients connect to the MFT SSH server.

To enable SSH tracing, perform the following steps:

Procedure

1. Go to **Configuration > System Configuration**.
2. Click the **Server Settings** tab.
3. Select the MFT host where you want to enable SSH tracing.
4. Click **SSH Client** or **SSH Server** tracing as needed.

We do not recommend tracing both client and server at the same time, since it produces a lot of trace entries.

5. Set the **SSH Trace Level** to All Messages.
6. All trace data is written to the following file: <MFT-Install>/server/logs/ssh-trace.txt

i Note: When enabling SSH tracing, try to do it for as short a period as possible. SSH tracing traces all SSH Client and or server requests and can produce a log of output. If possible, try to reproduce the problem on a test system so that the ssh trace data is limited.

JMS Tracing

JMS tracing traces all Command Center data processed by the JMS service. For example, JMS initiated file Internet Server or Platform Server transfer requests can be traced by JMS tracing.

Procedure

1. Go to **Management > Command Center Services > JMS Service > Configure JMS Service**.

At this point, you can set JMS tracing globally for all Command Center instances, or you can enable JMS tracing for individual Command Center instances. Since incoming JMS requests can be processed by any Command Center listening on the queue, sometimes you need to enable JMS tracing for all instances.

Enabling JMS tracing for all instances

Procedure

1. In Server properties, set **Trace level** to All Messages.

Enabling JMS tracing for all individual MFT instances

Procedure

1. Select the server where you want to enable tracing.
2. In the **Local/Remote Server** properties, set **Trace level** to All Messages.

MFT Tracing

MFT tracing can help to debug a variety of admin and transfer problems. Unless the problem is an SSH protocol issue, MFT tracing is typically sufficient to debug most issues. You can add tracing at the following levels:

Tracing Level	Description
System	The System Configuration page allows you to enable tracing for individual Internet Server or Command Center instances. This level of tracing produces a lot of trace data. Hence, only do this for short periods of time.
User	Tracing is enabled for all admin and transfer functions performed by this user. This is a good trace to enable when debugging admin problems.
Server	Tracing is enabled for all transfer functions and Command Center functions performed with this server. This is a good trace to enable when debugging transfer problems for a particular server. However, if a lot of data is transferred to this server, the traces can be pretty large and can affect transfer speed. Note that transfers to Platform Servers write a lot of data; all data sent to or received from the Platform Server is traced.
Transfer	Tracing is enabled for all transfer functions for this transfer definition. This is a good trace to enable when debugging transfer problems for a particular server when the problem typically happens for a single transfer definition.

MFT Traces are written to the logs or trace folder. Two types of trace files are typically written:

Types of Trace File	Example
User traces	user-trace-MFT-app-2020-12-18.txt
Transfer traces	IC1800001H-trace-cfjava-app.txt IC1800001H-trace-cfjava-comm.txt IC1800001H-trace-sshproxy-app.txt IC1800001H-trace-sshproxy-comm.txt

It is not a good idea to leave traces on for too long. The **Diagnostics** page shows you if traces are enabled. There are two places that show trace settings:

Trace Setting Places	Description
Active Trace Object Count	Shows how many trace objects are active.
Trace Settings	Shows details on the objects where tracing is enabled.

Other Types of Tracing

In the event of issues with an SSL negotiation, you might be prompted to enable SSL tracing. SSL tracing is enabled by adding a Java Option to the `setenv.sh`, `setenv.bat`, or Windows Service Java options.

As most SSL issues occur during the handshake, you can enable SSL handshake tracing.

If you want to debug all SSL data, use the following command to turn on all SSL tracing:

```
-Djavax.net.debug=ssl
```

The output for both traces goes to the `catalina.out` file.

i Note: There is no way to dynamically turn SSL tracing on/off. If you enable SSL tracing, it is on for the entire time that the MFT process is enabled. If you configure the system to debug all SSL tracing, a lot of output is displayed. Whenever possible, the best practice is to enable SSL tracing on a test system.

Common Debugging Scenarios

This section discusses some common problems and general comments about debugging these issues. The debug pages and traces that were discussed in other topics are cross-referenced in these sections. For detailed information, see the respective referenced topic.

Key or Certificate Log ins are Failing

For Incoming requests: (i.e. Client connects to MFT server)

- Make sure that the MFT Service (FTP, SSH, and Platform Server) Client Authentication is configured for Certificate or Password, Certificate and Password or Certificate Only. Otherwise, the MFT Server does not request a client certificate.
 1. Go to **Configuration > System Configuration**.
 2. Look for the **Global FTP/SSH/Platform Server** settings.
- Make sure that the client has sent their key or certificate to the MFT admin.
- Make sure that the MFT Admin has associated the key or certificate with the incoming user ID.
- Make sure that the SSH client is sending the correct user ID to the MFT server. Look in the `../logs/audit/login-audit-yyyy-mm-dd.log` file for an entry that matches the time of the incoming request.
- If necessary, enable log in tracing on the Internet Server. See the section above on log in tracing.

For Outgoing requests: (i.e. MFT client connects to target server)

- After adding the partner server, go to **Partner > Servers > Manage Servers**, select the server and retrieve the public key for that server.
- The server definition defines the system key that is used when communicating to the target system. For SSH, the transfer definition can override the server system key.
- Make sure that we have sent the public key associated with the system key to the Transfer partner.
- Make sure that you have defined the used and a dummy password in the server or transfer definition. Some protocols need to know the user associated with the public

key.

- If outgoing requests still get authentication errors, contact the transfer partner and find out the error message they are getting.
- You can turn on enable tracing for the target SSH server. Depending on the cause of the failure, you might get some additional information on the cause of the failure; however more likely, the authentication exception is displayed.

A Platform Server Client receives error: "This file is not eligible to be transferred. No suitable File Transfer Definition for this alias found"

There are a few possible causes of this failure:

- The Platform Server client user does not have a transfer definition that matches the requested Virtual Alias. Use the **Manage Transfers** page to search for a transfer definition that matches the user, Virtual Alias and Upload/Download.
- The **Platform Server Remote File** parameter defines the Virtual Alias. For example: RF:abc/file1.txt. "abc" is the Virtual Alias" and file1.txt is the file name.
- The Virtual Alias is case-sensitive and must exactly match the directory name in the Platform Server Remote File Name field. Note that MFT removes leading and trailing slashes.

i Note: This issue could also occur on SCP Clients. On GUI FTP and SFTP clients, all Virtual Aliases configured for the user is displayed. Hence, this error is never received.

Files are not displayed in the user's download directories

Most likely MFT was unable to contact the target server to a directory list.

1. Look at the Error Events.
2. Go to **Diagnostics > Error Events**.
3. Looks for an error event that matches the time the error occurred.

An exception is generated when a user accesses an admin page

When an error occurs on an admin page, the stack trace for the error is not displayed on the screen. If tracing is enabled, the stack trace must be written to the user trace. If tracing is not enabled, the stack trace must be written to the `catalina.out` file in `<MFT-Install>/server/logs`.

Generally speaking, TIBCO Technical Support needs the stack trace for this issue. You must also go to the **Diagnostics** page for the server and click **Save Server Diagnostics to File**.

An SSH client initiates a file transfer that is failing

First you need to find out when the transfer failed. There are a few places where the transfer could fail:

1. The transfer started. You can tell if a transfer started by looking for an audit record. The Audit record should have two fields that gives you more information about the failure:
 - a. Transfer Status Msg
 - b. Proxy Status Msg
2. If an audit record was written, then the SFTP Client negotiation was successful, the user successfully logged in and the Virtual Alias was found. Most likely the failure was connecting to the target server to upload or download the file.
3. If an audit record was not written, then there are a few possible errors:
 - a. If the error was an authentication failure, a record must be written to this file: `../logs/audit/login-audit-yyyy-mm-dd.log`. If you see an error in this file, then the SSH negotiation was successful, but authentication failed.
 - b. There was a negotiation error in negotiating the SSH session. The `catalina.out` file might contain a stack trace with the error received. More likely, the SSH client might show the cause of the error, since the SFTP client determines the SSH algorithms actually used. If you cannot determine the cause of the failure, you might need to enable tracing. Since the user has not logged in yet, a user trace does not help. You might need to enable SSH Server Tracing. See the information above about enabling SSH tracing.

A Platform Server initiates a file transfer that is failing

Unlike other protocols, the cause of the error is not hidden from other Platform Server clients and servers. A Platform Server audit record is written for each Platform Server transfer attempted. Use the `cfinq` utility (UNIX and Windows) or Platform Server panels (z/OS and IBMi) to see the cause of the failure. An Internet Server Audit Record is only written if the transfer actually starts. This means that the following must have occurred:

- The Platform Server Client was able to connect to the Internet Server port.
- SSL/TLS negotiation has been successful.
- Authentication using the credentials (Certificate or password) for the user was successful.
- The Virtual Alias specified in the remote file name matched a valid transfer definition for that user.

User traces are often helpful in cases like this. You can enable traces for the user ID for the transfer definition. Typically, it is a good idea to enable tracing on the object with the least usage; this is typically the transfer definition. Traces for Platform Server transfers write information about all data sent over the TCP Connection; these traces can get very large and can slow down transfers.



Important: Be sure to turn tracing off as soon as possible after you have collected the necessary information.

An LDAP Sync is failing

When an LDAP Sync fails, only general information is displayed on the admin console. You might need to look at the `catalina.out` and possibly enable tracing to determine the cause of the failure. Here are some things you can do to debug LDAP sync issues:

1. Is the issue just for LDAP Sync? Can LDAP users log in to MFT? If LDAP users can log in to MFT, then the connectivity to LDAP is good; the problem might be just for LDAP Sync.
2. Test the LDAP Authenticator.

Go to **Configuration > Authenticators > Manage Authenticators**.

Select the **Test** button next to the authenticator that you want to test. This displays a count of users that would be synced as well as the rights that would be synced.

3. Try to sync one user.

Go to **Administration > LDAP Sync**.

Click **Sync user**.

Enter the user ID in format: *authenticator-userid*

Click **Sync**.

Only this user is synced.

4. TIBCO Support might request a trace of the LDAP Sync. Enable tracing for the user performing the LDAP Sync. Then perform the LDAP Test or the LDAP Sync and then disable tracing for the user. The user trace should show more information about the LDAP Sync failure.
5. Each LDAP Sync writes information to the following file:

```
<MFT-Install>/logs/messages/ldap_sync_report_messages-MFT-yyyy-mm-dd.txt
```

6. TIBCO Technical Support might request this file.

A Transfer Event Alert is not executing

Transfer Alerts are executed when a transfer is about to be written to the Audit database. If no record is written to the Audit database, then no alert is executed. Alerts are executed differently for Internet Server and Platform Server transfers.

Internet Server Transfers: Alerts are executed on the Internet Server where the transfer executed

Platform Server Transfers: Alerts are executed on the Command Center where the Platform Transfer was collected

Before proceeding, you need to determine if the alert was not executed, or if the alert was executed but the alert action failed. You can search the alerts that have been executed.

1. Go to **Reports > Alert History > Search Alerts**.

All alerts that have been executed today are displayed.

2. Select the entry where the "Alert Description" matches the alert definition. You can then look for the action to see if it executed successfully.
3. If you see a matching Alert History record, the alert did execute. Now you must check

the Alert Actions:

4. Click the **Alert Audit Id** to get the detailed information on the alert.
5. Look for the Alert Action. The action status is displayed for all alert actions executed. If the request failed, the **Status** field displays the exception that caused the request to fail.

If you do not see a matching alert history record, then the alert did not execute. You must perform the following checks:

1. Verify that the transfer completed and was written to the audits table.
2. Go to **Reports > Audits > Search Audits**.
3. Set the filter criteria to search for the transfer that should have generated an alert. If no transfer was found, then the transfer was not executed; therefore, no alert was generated.
4. If a transfer audit record was found, compare the transfer audit parameters to the **Alert Trigger Criteria**. Make sure that all of the Trigger criteria defined matches the audit record. Depending on how the trigger criteria are defined, the trigger comparisons might be case-sensitive.
5. If you still cannot determine why the alert is not written, then TIBCO Technical Support might need you to enable tracing. For an Internet Server transfer, enable tracing for the transfer definition. For Platform Server, you need to enable tracing for the Collector. To do this, enable tracing for user "Collector", run the Platform transfer again, wait for the platform transfer to be collected and then turn off tracing.

A Transfer Non-Event Alert is not working

Transfer non-event Alerts must be configured in two places:

- *Alert Definition*: Defines the Trigger criteria and the actions to be executed
- *Scheduler definition*: Defines when the job checks if the transfer has executed.

There are a few reasons that a non-event alert is not reporting that transfers have not executed:

- The Non-Event Transfer Alert scheduler job was not configured or was configured incorrectly.
- A transfer actually executed that matched the alert trigger criteria. So, the alert is not

triggered.

Check the events (**Diagnostics > Events > Search Events**) to see if the alert executed.

The Collector is not collecting transfers from a Platform Server

When a server is not collecting transfers, there are a few possibilities:

- If you just enabled collection for this Platform Server, you must restart the collector for this change to take effect.
- There is no connectivity to the target Platform Server or the target Platform server is not active or the IP name or port is incorrect.
- The credentials passed to the Platform Server are incorrect.
- The Platform Server is rejecting the Collection request. For collections to work, the Platform Server needs the following configurations:
 - The Platform Server Node definition for the Command Center must be defined with Command Center Support=Audit or All.
 - The user that is executing the collector request must be a member of the cfadmin or cfbrowse groups.

Each time a collection request is made to a Platform Server, Command Center writes a record to this file:

```
CollectorAudit-audit-MFT-app-YYYY-MM-DD.log
```

This file displays a status message that includes this information:

- Start and end date/time of the collection request.
- The Server name (called Nodename in this file) that was being collected.
- The Status Message (displays a descriptive message if the request fails).
- The number of transfers collected(NumCollected) and the number written to the database (NumProcessed).
- The ID of the last record collected.

The records in this file should tell you why the Collection is not working.

A scheduler request is not executing properly

1. To verify if the scheduler job executed,

Go to **Diagnostics > Events > Search Events**.

One record is written to the **Events** database for each scheduler job that is executed. Use the **Search Criteria** to filter the events that are returned. If you see the job, click the *Event Id* to get the detailed record. The detail record is displayed. The detail record shows the following information:

- The start and end times for the job
- The Job Type (Event Type)
- The Status: Success or Failure
- Status message: Descriptive error if the request failed.

If you do not see the job, look for this file:

```
<MFT-Install>logs/message/Scheduler-yyyy-mm-dd.txt
```

A summary record is written to this file for each job executed.

2. To check if the job is currently running,

Go to **Management > Scheduler > Jobs > Active Jobs**.

This display lists the jobs currently running.

3. To display the configured jobs,

Go to **Management > Scheduler > Jobs > Manage Jobs**.

Use the results table to limit the jobs that are returned.

The results table shows two important fields:

Field	Description
Last Fire Time	The last time the job was fired (i.e. executed)
Next Fire Time	The next time the job will be fired (i.e. executed)

A JMS initiated Transfer request does not execute

Command Center allows users to initiate Internet Server or Platform Server transfers by submitting XML to a JMS queue. Command Center might not execute a transfer due to the following reasons:

1. Verify that Command Center can connect to JMS and can listen on the queue.
2. Go to **Management > Command Center Services > Configure JMS Service**.
3. Select the Command Center instance where the test runs and click **Test**.
4. MFT attempts to connect to the target JMS server for all of the Command Center queue and topics. If this fails, a generic message is displayed. You need to look in the catalina.out to see the actual cause of the failure.
5. Check the following step:
 - a. The JMS jar files are copied to: <MFT-Install>/server/webapps/cfcc/WEB-INF/lib
6. After copying these files, MFT must be restarted.
7. The JMS Server URL is correct. Check the DNS Name and the port.
8. Some other process might be reading the JMS Queue. Use a JMS utility to verify the number of listeners for the queue.
9. Make sure that the request is written to the JMS Queue defined in: **Configure JMS Service: Transfer and Job Request Properties: Queue Name**
10. If the JMS request message stays in the queue and is not consumed, make sure the JMS request was initiated with the correct property. These properties are used as selectors by the Command Center JMS Service:
 - a. Platform Transfers: Platform-Request
 - b. Internet Transfers: Internet-Request
11. If necessary, enable tracing on the JMS Service
12. Go to **Management > Command Center Services > JMS Service > Configure JMS Service**.
13. Set **Trace level** to All Messages.
14. Execute the process that adds the message to the JMS Queue. Two types of trace field are written:

a. EMS-Server...-trace-MFT-app-yyyy-mm-dd.txt

15. This displays the date read from the JMS queue.

```
JMSServlet-trace-MFT-app-yyyy-mm-dd.txt
```

16. This file shows the processing performed for the requested function.

FTP Client directory list or transfer requests are failing

This assumes that the FTP Client can connect to the MFT FTP server and authenticate to the MFT FTP server. FTP transfers are susceptible to problems due to the requirement for two TCP connections for directory list, put, and get requests.

When a directory list fails, it generally means that the data connection has timed out or has been rejected. When a data connection fails, get and put file transfer requests also fail.

Here are some things that should be checked.

- Check that firewall ports are defined correctly. In particular, when using a non-standard port (other than 21), the firewall needs to be configured for FTP processing.
 - The MFT FTP Service has two parameters that must be configured correctly.
1. Go to **Administration > Transfer Servers > Configure FTP Server**.
 2. Set **External IP Address** to Yes.
 3. Set **External IP Address** to the IP address or IP name of the Internet Server host.
 4. Go to **Configuration > System Configuration**.
 5. In **Global FTP** settings, limit **Local Ports Set** to Yes.
 6. Starting Port Set to a number like 40000 or 30000. Along with the "Number of Ports to use", this defines the local TCP Ports that are used for data connections. You might need to supply this information to the firewall admins at the client and server side.
 7. The number of ports to use are defined by how many TCP ports can be used for FTP data connections.
 8. Generally speaking, PASV mode is the simpler to get working with firewalls. Hence, it is probably better for clients to use PASV mode when connecting to the MFT FTP server.

9. MFT supports two types of FTPS connections:
 - a. Explicit Clients connect to the Clear text FTP Port (usually 21) and issue a PORT command to enter SSL mode. The PORT command must be received before MFT processes the credentials.
 - b. Implicit Clients connect to the FTP Implicit SSL port. All traffic is SSL-encrypted.
10. When using Implicit SSL mode, make sure the client is connecting to the correct port. The default port for Implicit SSL is 990.

Using REST calls to configure Internet Server and Command Center

MFT provides the following types of REST calls:

- Configure MFT Administrative functions (Command Center and Internet Server)
- Perform Platform Server functions (Command Center only)
- Perform File Transfers (Internet Server only)

This describes the REST calls that are used to configure MFT administrative functions. The same rules apply when performing Platform Server Functions. [Performing File Transfers](#) is discussed in another section.

Supported Admin and Command Center REST Calls

MFT supplies REST calls to perform the most common administrative functions which are described in the following table.

Function	Description
Departments	List, add, delete, and update departments.
ErrorEvent	Retrieves Error Event details.
Groups	List, add, delete, and update groups.
IS Audits	Search, delete, and get Internet Server Audit records.
Keys	Search, add, update, delete, and retrieve key from server.

Function	Description
LDAP Sync	Synchronize a user or an authenticator.
Lockout Release	Release locked users, IP address, and systems.
MFT Version	Retrieve MFT version information.
PGP Public Keys	Search, add, update, and delete PGP public keys.
PGP System Keys	Search, add, update, delete, and import PGP system keys.
Roles	Retrieve and assign user to a role.
Server Credentials	List, add, update, or delete server credentials.
Servers	List add, update, and delete server definitions.
System Keys	List, add, update, delete, export, import, and retrieve system key.
Tags	List, add, update, and delete tags.
Transfer Servers	Stop, start, or get status of a transfer service.
Transfer Servers Config	Retrieve and update transfer servers configuration.
Transfer Server Status	Start, stop, or get status of transfer service.
Transfers	List, add, delete, or update transfer definitions.
Users	List, add, delete, or update user definitions.

MFT supplies REST calls to perform the following Command Center and Platform Server functions.

Function	Description
Active Transfers	Retrieve active Internet Server transfers.

Function	Description
Alerts	List add, update, or delete alerts.
DNI	List DNI daemons and templates to add, delete, and update templates.
Events	Retrieves event details.
Host Status	Retrieves status of managed servers.
Internet Server Transfers	Initiates an Internet Server transfer.
Jobs	List, add, update, delete, and execute scheduler jobs.
PS Audits	Search, delete, and get a Platform Server audit record.
PS Nodes	Get, add, update, and delete bank nodes. Get, add, update, and delete Platform Server nodes.
PS Responder Profiles	Get, add, update, and delete bank Responder Profiles. Get, add, update, and delete Platform Server Responder Profiles.
Platform Server Transfers	List, add, delete, update, and execute Platform Server transfers.
PS Responder Profiles	Get, add, update, and delete bank User Profiles. Get, add, update, and delete Platform Server User Profiles.
Platform Server Active Transfers	Retrieve and cancel active Platform Server transfers.
Server Status	Retrieve Server Status summary and details.
Services	Start, stop, and get status of Command Center services.
Statistics	Search for Transfer statistics.

Documentation on REST Calls

REST calls are documented in the following places:

- [JSON files](#)
- [Redoc Documentation](#)

JSON files

Three files are located in this directory:

```
<MFT-Install>/server/webapps/cfcc/public/docs
```

File	Description
admin.json	Defines administrative REST calls.
admincc.json	Defines Command Center and Platform Server REST calls.
ft.json	Defined File Transfer REST calls

These files define the support REST calls. They also define the parameter names and the enumerations for the parameters.

These json files can be imported in Postman and TIBCO ActiveMatrix BusinessWorks™ to execute the rest calls.

Redoc documentation

Redoc is an OpenAPI tool that generates a graphical representation of the REST calls. MFT uses Redoc for API documentation.

API Documentation can be accessed by clicking on the **Redoc** link at the top of each Internet Server or Command Center admin page.

Format of the REST Call

The following is a sample REST call URL:

```
https://mft.server.com:8443/cfcc/rest/admin/v6/servers
```

REST calls use the following format:

REST Call	Format
Admin Calls	https://your.mft.server.com:8443/cfcc/rest/admin/v6/function/parameters
Command Center Calls	https://your.mft.server.com:8443/cfcc/rest/admincc/v6/function/parameters
FT Transfer Calls	https://your.mft.server.com:7443/cfcc/rest/ft/v6/function/parameters

URL Parameters

Parameter	Description
https	It's a best practice to use HTTPS and not HTTP.
your.mft.server	Defines the DNS name of the MFT server.
8443/7443	Defines the TCP port of the MFT server.
cfcc	Defines the MFT context.
rest	Must be set to rest.
v6	Defines the current REST version. Each new MFT release supports a new version of REST. The old versions of REST are supported for backward compatibility.
function	Defines the function to perform.

Parameter	Description
	Examples are: users, servers, pstransfers
parameters	Defines parameters passed to the REST call. Parameters are typically set on GET, PUT, and DELETE calls.

Authentication

MFT uses basic authentication. Hence, MFT credentials must be set in the HTTP Authentication Header.

REST Data

For GET and DELETE requests, the object being retrieved or deleted is defined in the REST URL.

For PUT requests, the object being updated is defined in the REST URL. The update is defined in JSON data in the HTTP body.

For POST requests, the object being created is defined in JSON data in the HTTP body.

Testing REST Calls

Before writing code to create REST calls, we suggest using a tool to understand how the REST calls work. You can try one of these utilities to test the REST calls.

1. curl. Here is a sample curl request to retrieve the details of user "admin".

```
curl --tlsv1.2 -v -k -u admin:changeit -X GET
https://mft.server.com:8443/cfcc/rest/admin/v6/users/admin
```

2. To create an object, you can create a file that contains the JSON data. Then, reference the file in the curl command using the -d parameter:

```
curl --tlsv1.2 -v -k -u admin:changeit -d "@AddServer.json" -H
"Content-Type: application/json" -X POST
https://mft.server.com:8443/cfcc/rest/admin/v6/servers
```

3. TIBCO BusinessWorks™.

4. Postman or a similar tool.

Configuring and Using JMS

This section explains how JMS is used in Internet Server and Command Center. It also shows the steps necessary to configure and use JMS in Command Center and Internet Server.



Note: Command Center is required to configure JMS. If you have not installed Command Center, then you cannot use JMS.

JMS Support Overview

JMS is a messaging service that allows applications to exchange messages with other applications. MFT supports the following types of messages:

Message	Description
Queues	Messages are stored in a Queue and will remain there until the messages are consumed.
Topics	Publish /subscribe method. One or more applications can subscribe to a topic and can read messages published to the topic.

MFT JMS support was originally developed along with the MFT BusinessWorks Plug-in but the same interface used by BusinessWorks can be utilized by customer applications.

Both MFT Command Center and MFT Internet Server support JMS. The following JMS features are available:

Command Center

- Initiate Internet Server File Transfer
- Initiate Platform Server File Transfer

- Execute Scheduler Job
- Audit Inquiry
- Send Alerts for Platform Server Transfers
- Platform Server transfer completion notification

Sample XML files are located in:

```
<MFT-Install>/server/webapps/cfcc/example/jms:  
AuditRequest.xml  
ExecuteJobRequest.xml  
TransferRequestInternetServer.xml  
TransferRequestPlatformServer.xml
```

Internet Server

- Send Alerts for Internet Server Transfers
- Internet Server transfer start and completion notification
- Send file to JMS Queue
- Receive file from JMS Queue



Note: Command Center is required to configure the JMS interface. Hence, JMS cannot be used unless you have installed MFT Command Center.

Supported JMS Software

MFT should support any JMS software that is compliant with the JMS standard and has compatible JMS jar files that can be used by MFT. MFT has been tested with the following JMS software, although not all versions have been tested:

- TIBCO EMS
- ActiveMQ
- IBM MQ

The key to getting JMS working is to use the proper jar files and configure the MFT JMS interface properly. Since most MFT customers use TIBCO EMS, this section is tailored for TIBCO Enterprise Message Service™ (TIBCO EMS).

Installing TIBCO EMS JAR Files

To install TIBCO EMS JAR files, perform the following steps:

Procedure

1. Three JAR files are required for EMS 8.0 and above:

- tibjms.jar
- tibcrypt.jar
- jms-2.0.jar

2. Get the above files from the EMS installation.

3. Copy these files to:

```
<MFT-Install>/server/webapps/cfcc/WEB-INF/lib
```

4. Restart the MFT server.

i Note: This procedure must be performed on all Command Center and Internet Server instances that uses EMS.

Configuring JMS On Command Center

To configure JMS on Command Center, perform the following steps:

Procedure

1. Go to **Management > Command Center Services > JMS Service > Configure JMS Service**.

i Note: The **Configure JMS Service** help pages have a detailed explanation of parameters needed to configure JMS. The help pages also has a detailed explanation of how to perform problem determination if JMS connections fail.

2. Enter the necessary connectivity parameters in the **Server Properties** tab.
3. Set **Enabled** to Yes.

There are also four tabs that define the queues or topics used by MFT.

Tab	Description
Alert Properties	Defines the topic name where alerts are written. Note that the alert definition allows you to override the topic name and also allows you to write alerts to a defined queue instead of to a topic.
Audit Properties	Defines the queue name where Command Center listens for incoming Audit Inquiry requests. This tab allows you to define JMS request and Response Types. The Request Type is used by Command Center to filter audit requests. MFT sets the response type so that applications can filter the MFT response.
Transfer Notification Properties	Defines the JMS topic where transfer notification properties are written. Also, defines the type of Internet Server and Platform Server notifications that are written. You can set the JMS message type for start, end, and activity notifications.
Transfer and Job Request Properties	Defines the queue name used to initiate transfers. There are three types of transfers supported. Each type allows you to define JMS request and response types. The Request Type is used by Command Center to filter transfer requests. MFT sets the Response type so that Applications can filter the MFT response. The following transfer types are supported:

Tab	Description									
	<table><tr><th>Transfer Type</th><th>Description</th></tr><tr><td>Platform Server Transfers</td><td>Initiate a 3rd party Platform Server transfer. This means that Command Center initiates a request to Platform Server "A" to send a file to Platform Server "B".</td></tr><tr><td>Internet Server Transfers</td><td>Initiates an Internet Server file transfer.</td></tr><tr><td>Execute Jobs</td><td>Execute a scheduler job</td></tr></table>	Transfer Type	Description	Platform Server Transfers	Initiate a 3rd party Platform Server transfer. This means that Command Center initiates a request to Platform Server "A" to send a file to Platform Server "B".	Internet Server Transfers	Initiates an Internet Server file transfer.	Execute Jobs	Execute a scheduler job	
Transfer Type	Description									
Platform Server Transfers	Initiate a 3rd party Platform Server transfer. This means that Command Center initiates a request to Platform Server "A" to send a file to Platform Server "B".									
Internet Server Transfers	Initiates an Internet Server file transfer.									
Execute Jobs	Execute a scheduler job									

- Select individual Internet Server and Command Center instances. This allows you to override three parameters for each Internet Server or Command Center instance:
Enabled
JMS Server URL
Trace Level
- When you have finished configuring the JMS parameters, click the **Update** button to save the changes.

Testing JMS Connections

Use the **Test** button to test the JMS configuration. You can test the JMS configuration on any installed Internet Server or Command Center through the **Select Server to Test** dropdown box. The **Test** button performs the following tests:

- Verify the connectivity to the target JMS Server as well as the credentials (if required)
- Verify that the Queues and Topics have been defined correctly.

If the test fails, only a generic message is displayed. You might need to check the `catalina.out` for the actual error messages. Follow the help page "Problem Determination" instructions.

Starting the JMS Service

Before you start the JMS service on an Internet Server or Command Center instance, you should test the JMS connection on that instance. Follow the instructions in "Testing JMS Connections" to test the instance before attempting to start it. If the test fails, then you cannot start the instance.

To start the JMS service, perform the following steps:

Procedure

1. Go to **Management > Command Center Services > JMS Service > JMS Server Status**.
2. On this page, select the Internet Server or Command Center instance.
3. Perform the required action:

Action	Description
Server Status	Displays the current JMS service status on the selected instance.
Start Service	Start the JMS service status on the selected instance.
Stop Service	Stop the JMS Service status on the selected instance.

Using JMS on Command Center and Internet Server

Using JMS on Command Center

Once you start the JMS Service on Command Center, Command Center listens for incoming transfer and audit requests. No other configuration needs to be performed.

Using JMS on Internet Server

Internet Server supports the following JMS capabilities:

- Writes Internet Server, starts activity, and ends notifications.
- Sends alerts.

For more information see the [Configuring Alerts to Send JMS Messages](#).

- Writes data to a JMS queue.

For more information see the [Configuring JMS Servers and Transfers](#).

- Reads data from a JMS queue.

For more information see the [Initiating Transfers to a JMS Queue](#).

- Initiates an Internet Server transfer to read data from a JMS queue and write to a target Virtual Alias.
- Initiates an Internet Server transfer to read data from a Virtual Alias and write data to a JMS queue.

Configuring Alerts to Send JMS Messages

Alerts can be configured to send JMS messages for the following events:

- A user logs in triggering an alert.
- A non-event transfer triggers an alert because a transfer has not completed in the defined time frame.
- A transfer has completed triggering an alert.

- Platform Server alerts are created when Command Center collects an audit record from a target Platform Server.
- Internet Server alerts are created when an Internet Server transfer completes.

To create an alert for an Internet Server transfer, perform the following steps:

Procedure

1. Go to **Transfers > Alerts > Add Transfer Event Alert**.
2. Configure the **Required Alert Information** and **Alert Trigger** tabs.
3. Click the **Alert Action > JMS** tab.
4. Select the **Send JMS Message** checkbox.
5. Configure a comment.
6. Configure a JMS type.

Result

When MFT writes the alert message, it sets the JMS type attribute. Then, JMS applications reading the queue or topic can filter messages based on the JMS type.

Viewing Alerts

To view alerts through the current page, perform the following steps:

Procedure

1. Go to **Reports > Alert History > Search Alerts**.

If you do not see a matching alert, then the alert trigger criteria did not match the transfer.

If you see an alert click the Alert ID and you see the alert detail. This shows whether the JMS request was successful.

Configuring JMS Servers and Transfers

Writing data to a queue, or reading data from a queue is relatively simple. Once you have configured and tested the JMS configuration, you must create two definitions:

- [Server definition](#)
- [Transfer definition](#)

Creating a Server Definition

To create a server definition, perform the following steps:

Procedure

1. Add a server definition.
2. Go to **Partners > Servers > Add Server**.
3. Enter the server name.
4. Set the **Server Type** to JMS.

By default, the "IP Address or fully qualified IP Name" is ignored. It defaults to use the JMS Server connectivity defined in the **Configure JMS Service** page.

5. To use a different JMS Server, configure the "IP Address or fully qualified IP Name" and "IP Port" and select the **Override JMS Service Configuration** checkbox.

Creating a Transfer Definition

To create a transfer definition, perform the following steps:

Procedure

1. Go to **Transfers > Internet Server Transfers > Add Transfers**.
2. Configure the following fields the **Required Transfer Definition** tab:

Field	Instruction
Client File Name	Set to any value; this field is ignored
Server File Name	Set to the queue name. Alternatively, if you want the client software (especially Platform Server) to set the queue name, you can use a token like #(FileName) to define the server file name.

Field	Instruction
Directory Transfer	Set to No.
Description	Enter a text description.
Authorized User Id and/or the Authorized Group Id	Set as needed.
Server Name	Set the server name to the JMS server created in the prior step.
Transfer direction	Set to Upload to Server, Download to Client, or Both.
Virtual Alias	Set to a unique virtual alias for that user or group.

3. Configure the required information in the **JMS Properties** tab.

The JMS Properties fields are different for Uploads and downloads.

Field	Description
Input Selector	Used for downloads to filter JMS messages that are read.
Output JMSType Property	Used for uploads and sets the JMSType property.
Output Property	Allows MFT to set other JMS properties when uploading data to JMS.
Max Message Size	For uploads, allows you to limit the message size. Messages larger than this size will be split into multiple messages.

Field	Description
Write EOF Message	Defines whether a null message is written after the entire file has been uploaded.

4. Click **Add** to add the new transfer definition.

Initiating Transfers to a JMS Queue

This section describes sample Platform Server for UNIX commands that can be used to read or write JMS Queues.

Sending a file to a JMS Queue

```
cfsend n:nftis lf:/data/file.to.send rf:/VirtualAlias/QueueName
```

If the transfer definition server file name has been set to the #(FileName) token, then this uploads data to queue "QueueName".

If the transfer definition server file name has hard-coded the queue name, the queue defined in the transfer definition is used.

Receiving a file from a JMS Queue

```
cfrecv n:nftis lf:/data/file.to.recv rf:/VirtualAlias/QueueName
```

If the transfer definition server file name has been set to the #(FileName) token, then this downloads data from queue "QueueName".

If the transfer definition server file name has hard-coded the queue name, the queue defined in the transfer definition is used.

Configuring Hawk Microagent

MFT includes Hawk Microagent that you can configure to communicate with a Hawk Agent. Hawk collects the following data from [TIBCO® Managed File Transfer Command Center](#):

- Active Transfer Count
- Active Session Count
- Database Connection Count
- Active Users Count by Protocol
- Listener Status
- The day's Transfer Count by Protocol
- The day's Transfer Count by Server Name
- Current Thread Count
- The day's Transfer Count Platform Server
- Server Status
- DNI Status
- Transfers by Host

Setting the Environment Variables

Before configuring Hawk Microagent, set the following environment variables:

1. Set the MFT_CLASSPATH environment variable:

You must set the MFT_CLASSPATH environment variable to the TIBCO Hawk and TIBCO Rendezvous lib directories.

Following are some examples for Windows and UNIX/Linux/zLinux.

Windows

If TIBCO Rendezvous is installed at C:\tibco\tibrv\8.6 and TIBCO Hawk is installed at C:\tibco\hawk\6.2

```
SET MFT_CLASSPATH=C:\tibco\tibrv\8.6\lib\*;C:\tibco\hawk\6.2\lib\*
```

UNIX/Linux/zLinux

If TIBCO Rendezvous is installed at /tibco/tibrv/8.6 and TIBCO Hawk is installed at /tibco/hawk/6.2

```
export MFT_CLASSPATH=/tibco/tibrv/8.6/lib/*:/tibco/hawk/6.2/lib/*
```

Note: The asterisk (*) includes all .jar files in the directory. Windows uses a semicolon (;) as a path separator, whereas Unix and Linux use a colon (:) as a path separator.

2. Set the PATH or LD_LIBRARY_PATH environment variable:

You must set the PATH or LD_LIBRARY_PATH environment variable to the TIBCO Rendezvous C runtime library.

Following are some examples for Windows and UNIX/Linux/zLinux.

Windows

If TIBCO Rendezvous is installed at C:\tibco\tibrv\8.6

```
SET PATH=C:\tibco\tibrv\8.6\bin;%PATH%
```

UNIX/Linux/zLinux

If TIBCO Rendezvous is installed at /tibco/tibrv/8.6

```
export LD_LIBRARY_PATH=/tibco/tibrv/8.6/lib
```

Note: The asterisk (*) must not be included. Windows uses the PATH environment variable and UNIX/Linux use the LD_LIBRARY_PATH environment variable. TIBCO Rendezvous on Windows stores the libraries in the bin directory but on Unix/Linux/zLinux they are located in the lib directory.

3. Edit the java.library.path system property to include the Rendezvous C runtime libraries directory.
 - a. For Unix/Linux/zLinux:

Update the setenv.sh file in the server/bin directory.
 - b. For Windows running as a service:

Update the service.bat file in the server/bin directory.

- c. For Windows not running as a service:

Update the `setenv.bat` file in the `server/bin` directory.

Following is an excerpt from the `setenv.bat` file that has been updated:

```
-Djava.library.path="%CATALINA_BASE%/webapps/cfcc/WEB-INF/license-
files/;D:/tibco/tibrv/8.6/bin/"
```



Note: You need both the license directory and the Rendezvous directories here. On Windows use a semi-colon character to separate the directories. On UNIX/Linux/zLinux use a colon character.

Configuring Hawk Microagent

To configure Hawk Microagent, perform the following steps:

Procedure

1. Go to **Administration > Hawk Microagent > Configuration**.
2. From the dropdown menu, select a Hawk Microagent host.
3. Set the following parameters:

Parameter	Description
Host Name	The name of the Command Center or Internet Server instance.
Enabled	Whether the Hawk Microagent is enabled or disabled.
Display Name	The name of the Microagent displayed on the Hawk Console.
Domain	The Hawk domain the Microagent belongs to. For more information, see the TIBCO Hawk® documentation.
Agent Name	Typically set to none . For more information, see TIBCO Hawk® documentation.
Daemon	The port number the Microagent listens on.

Parameter	Description
Network	A specific network for outbound multicast Rendezvous communications. This field is empty by default. For more information, see the TIBCO Hawk® documentation.
Service	The port number the Microagent listens on.

4. After configuring the Microagent, go to **Administration > Hawk Microagent > Status**.
5. Select a server from the dropdown list and click the **Start Server** button.

Configuring Password Vault

You can configure the [TIBCO® Managed File Transfer Command Center](#) to store secrets in an external password vault. MFT secrets include passwords and private keys stored in the database, excluding user passwords stored as hashes. MFT supports HashiCorp Vault and Azure Key Vault.

Before you begin

You need the URL of the server you want to access.

- HashiCorp Vault: `https://hostname:port/v1/secret/data`
- Azure Key Vault: `https://vaultname.vault.azure.net`

For HashiCorp Vault, you need a token to access the vault. For Azure Key Vault, register your app with Azure and obtain the following information:

- Client Secret
- Client Id
- Tenant Id
- Object Id
- Secret Id

You must also perform the following steps:

- Create a key vault
- Update the firewall under networking to allow access from MFT
- Update the access policies to grant access to the registered application (MFT)
- Create a secret and grant these permissions: Get, List, Set, Delete, Recover, Backup, Restore

You also need an email address to receive notifications when the token or secret is about to expire.

Enabling Password Vault

On the System Configuration page, go to the **Password Vault** tab. Enter the information gathered in the previous steps and enable the password vault feature.

Moving Secrets

Enabling the password vault does not move all secrets to the vault. Any secrets added or updated after enabling the vault are stored in the vault. To move all secrets, click the move button. This moves all secrets currently stored in the database to the vault. To move secrets back to the database, disable the password vault in MFT and then move the secrets to the database.

You can select which MFT components to store in the vault. It's a best practice to store all components in the vault.

Settings

On the System Configuration page, go to the **Password Vault Settings** tab. The following table provides an explanation of the fields on this page.

Field	Value
Enabled	Defines whether the password vault is enabled or disabled. After disabling the Password Vault, MFT can still retrieve passwords from the password vault server.
Vault Software	Select the password vault server type. You can select from the following options: • Azure • HashiCorp
Server URL	Defines the IP Name or IP Address of the target password vault server. For example, • Azure Key Vault: <code>https://vaultname.vault.azure.net</code> • HashiCorp Vault: <code>https://hostname:port/v1/secret/data</code>

Field	Value
Azure Client Secret	(Displayed only when using Azure Key Vault) Defines the client secret used to access the password vault. This field is available on the Azure portal after registering your app with Azure.
Azure Client Id	(Displayed only when using Azure Key Vault) Defines the Azure Client Id used when configuring the Azure Key Vault. This field is available on the Azure portal after registering your app with Azure.
Azure Tenant Id	(Displayed only when using Azure Key Vault) Defines the Azure Tenant Id used when configuring the Azure Key Vault. This field is available on the Azure portal after registering your app with Azure.
Azure Object Id	(Displayed only when using Azure Key Vault) Defines the Azure Object Id used when configuring the Azure Key Vault. This field is available on the Azure portal after registering your app with Azure.
Azure Secret Id	(Displayed only when using Azure Key Vault) Defines the Azure Secret Id used when configuring the Azure Key Vault. This field is available on the Azure portal after registering your app with Azure.
Secret Expiration Email Recipient	(Displayed only when using Azure Key Vault) Defines the email address for notifications when the token or secret key is about to expire. Emails are sent when the key expires in 21 days or less, one email per day.
Trust All Certificates	(Displayed only when using HashiCorp Vault) Defines whether all SSL/TLS certificates or only trusted certificates are allowed. You can select from the following two options:

Field	Value
	• Yes: All certificates are trusted, regardless of whether they use a trusted CA certificate. Use this option only in test environments. • No: Only trusted CA certificates are accepted. This is the default option for production environments.
Token	(Displayed only when using HashiCorp Vault) Defines the token used to access the password vault.
Token Expiration	(Displayed only when using HashiCorp Vault) Defines the email address for notifications when the token or secret key is about to expire.
Email Recipient	Emails are sent when the key expires in 21 days or less, one email per day.
Default Name Prefix	Defines the prefix added to the secret object name. Ensure that this key is unique for each MFT cluster (an MFT cluster includes all MFT Internet Server and Command Center instances connecting to the same database). When connecting MFT Production, QA, and Development clusters to the same Vault server, define a different value for each MFT cluster.

Processing ZIP Files

For uploaded compressed files, run a command on the target servers using transfer definition's Post Processing Actions to extract an uploaded zip file. For example, unzip %LFILE -d targetDirectory.



Note: Users can upload only zip files to this transfer definition. Uploading other file types causes the Post Processing Action command to fail. MFT supports running commands for Post Processing Actions on the following server types: LOCAL, SSH, and Platform Server.

The browser client can download all files in a directory as a single zip file. To do this, right-click the directory and select **Zip Download**.

TIBCO Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, and join TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the [Product Documentation website](#), mainly in HTML and PDF formats.

The [Product Documentation website](#) is updated frequently and is more current than any other documentation included with the product.

Product-Specific Documentation

The documentation for this product is available on the [TIBCO® Managed File Transfer Command Center Documentation](#) page.

How to Contact Support for TIBCO Products

You can contact the Support team in the following ways:

- To access the Support Knowledge Base and getting personalized content about products you are interested in, visit our [product Support website](#).
- To create a Support case, you must have a valid maintenance or support contract with a Cloud Software Group entity. You also need a username and password to log in to the [product Support website](#). If you do not have a username, you can request one by clicking **Register** on the website.
- To submit ideas, suggestions, or feedback for TIBCO products, visit our [product Support website](#).

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to

extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the [TIBCO Ideas Portal](#). For a free registration, go to [TIBCO Community](#).

Legal and Third-Party Notices

SOME CLOUD SOFTWARE GROUP, INC. (“CLOUD SG”) SOFTWARE AND CLOUD SERVICES EMBED, BUNDLE, OR OTHERWISE INCLUDE OTHER SOFTWARE, INCLUDING OTHER CLOUD SG SOFTWARE (COLLECTIVELY, “INCLUDED SOFTWARE”). USE OF INCLUDED SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED CLOUD SG SOFTWARE AND/OR CLOUD SERVICES. THE INCLUDED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER CLOUD SG SOFTWARE AND/OR CLOUD SERVICES OR FOR ANY OTHER PURPOSE.

USE OF CLOUD SG SOFTWARE AND CLOUD SERVICES IS SUBJECT TO THE TERMS AND CONDITIONS OF AN AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER AGREEMENT WHICH IS DISPLAYED WHEN ACCESSING, DOWNLOADING, OR INSTALLING THE SOFTWARE OR CLOUD SERVICES (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH LICENSE AGREEMENT OR CLICKWRAP END USER AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE SAME TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc.

TIBCO, the TIBCO logo, the TIBCO O logo, and Slingshot are either registered trademarks or trademarks of Cloud Software Group, Inc. in the United States and/or other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only. You acknowledge that all rights to these third party marks are the exclusive property of their respective owners. Please refer to Cloud SG’s Third Party Trademark Notices (<https://www.cloud.com/legal>) for more information.

This document includes fonts that are licensed under the SIL Open Font License, Version 1.1, which is available at: <https://scripts.sil.org/OFL>

Copyright (c) Paul D. Hunt, with Reserved Font Name Source Sans Pro and Source Code Pro.

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the “readme” file for the availability of a specific version of Cloud SG software on a specific operating system platform.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. CLOUD SG MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S), THE PROGRAM(S), AND/OR THE SERVICES DESCRIBED IN THIS DOCUMENT AT ANY TIME WITHOUT NOTICE.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "README" FILES.

This and other products of Cloud SG may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at <https://www.cloud.com/legal>.



Copyright © 2003-2026. Cloud Software Group, Inc. All Rights Reserved.