

TIBCO® Enterprise Administrator Agent for TIBCO® Security Server Guide

*Software Release 2.2.0
March 2015*

Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document contains confidential information that 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 TIBCO Software Inc.

TIBCO and Two-Second Advantage are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

Enterprise Java Beans (EJB), Java Platform Enterprise Edition (Java EE), Java 2 Platform Enterprise Edition (J2EE), and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle Corporation in the U.S. and 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.

THIS 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 THIS SOFTWARE VERSION 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. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

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 "READ ME" FILES.

Copyright © 1996-2015 TIBCO Software Inc. ALL RIGHTS RESERVED.

TIBCO Software Inc. Confidential Information

Contents

| | |
|---|-----------|
| Figures | 4 |
| TIBCO Documentation and Support Services | 5 |
| Overview of TIBCO® Security Server | 6 |
| Overview of the Agent for TIBCO Security Server | 7 |
| TIBCO Enterprise Administrator Administrative Interfaces | 8 |
| Starting the TIBCO Security Server Agent | 9 |
| Log in to TIBCO Enterprise Administrator | 10 |
| Agent Management | 11 |
| Registering the TEA Agent for TIBCO Security Server | 11 |
| Reconnecting an Agent | 12 |
| Unregistering an Agent | 12 |
| TIBCO Security Server Management | 13 |
| Credential Authority Service | 13 |
| Registering the Credential Authority Service | 13 |
| Unregistering the Credential Authority service | 13 |
| Configuring the Credential Authority Service | 14 |
| Starting the Credential Authority Service | 16 |
| Stopping the Credential Authority Service | 16 |
| Resource Manager Service | 16 |
| Creating Password Credential Resources | 17 |
| Creating Login Credential Resources | 17 |
| Creating Keystore Credential Resources | 18 |
| Creating Kerberos Authentication Resources | 19 |
| Kerberos Authentication Resources | 20 |
| Creating LDAP Authentication Resources | 23 |
| LDAP Authentication Resources | 24 |
| Creating SiteMinder Authentication Resources | 29 |
| SiteMinder Authentication Resources | 29 |
| Creating WSS Authentication Resources | 32 |
| WSS Authentication Resources | 33 |
| Creating Subject Identity Resources | 34 |
| Creating Trust Identity Resources | 37 |
| Creating Credential Server Resource | 38 |
| Updating a Resource | 39 |
| Deleting a Resource | 40 |

Figures

| | |
|--------------------|----|
| Landing Page | 10 |
|--------------------|----|

TIBCO Documentation and Support Services

All TIBCO documentation is available on the TIBCO Documentation site, which can be found here:

<https://docs.tibco.com>

Product-Specific Documentation

Documentation for TIBCO products is not bundled with the software. Instead, it is available on the TIBCO Documentation site. To directly access documentation for this product, double-click one of the following file depending upon the variant of TIBCO Enterprise Administrator you are using:

For TIBCO Enterprise Administrator SDK use: `TIBCO_HOME\release_notes\TIB_tea-sdk_<version>_docinfo.html`.

For TIBCO Enterprise Administrator use: `TIBCO_HOME\release_notes\TIB_tea_<version>_docinfo.html`

The following documents can be found in the TIBCO Documentation Library for TIBCO® Enterprise Administrator:

- *TIBCO® Enterprise Administrator Release Notes*
- *TIBCO® Enterprise Administrator Installation*
- *TIBCO® Enterprise Administrator User's Guide*
- *TIBCO® Enterprise Administrator Developer's Guide*
- *TIBCO® Enterprise Administrator Agent for TIBCO Enterprise Message Service™ Guide*
- *TIBCO® Enterprise Administrator Agent for TIBCO® Security Server Guide*

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, contact TIBCO Support as follows:

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:

<http://www.tibco.com/services/support>

- If you already have a valid maintenance or support contract, visit this site:

<https://support.tibco.com>

Entry to this site requires a user name and password. If you do not have a user name, you can request one.

How to Join TIBCOCommunity

TIBCOCommunity is an online destination for TIBCO customers, partners, and resident experts. It is a place to share and access the collective experience of the TIBCO community. TIBCOCommunity offers forums, blogs, and access to a variety of resources. To register, go to:

<https://www.tibcommunity.com>

Overview of TIBCO® Security Server

The TIBCO® Security Server hosts all the services that are needed to support a wide range of security requirements regarding authentication, data integrity, data confidentiality, credentials, and single sign on.

The TIBCO Security Server currently hosts two types of services:

1. Credential Authority Service
2. Resource Manager Service

Credential Authority Service

The Credential Authority Service issues credentials needed to establish SSL connections and digitally sign and/or encrypt documents by automatically responding to authorized Certificate Signing Requests (CSR) coming from all products exposed to TIBCO® Enterprise Administrator in an enterprise.

Resource Manager Service

Use the Resource Manager Service to manage the security resource configuration objects needed across the products exposed to TIBCO Enterprise Administrator (TEA) in an enterprise. There are three types of resource configuration objects:

1. Credential configuration
2. Identity configuration
3. Authentication configuration

The Resource Manager Service supports the following activities, among others:

1. Create an SSL server socket
2. Create an SSL client connection
3. Safekeeping login credentials
4. Sign/encrypt a document
5. Authenticate a user's credentials
6. Support SSO between web applications
7. Authenticate SOAP requests

Overview of the Agent for TIBCO Security Server

TIBCO® Enterprise Administrator is shipped with an agent for the TIBCO Security Server. The TIBCO Enterprise Administrator provides a centralized administrative interface to manage and monitor multiple TIBCO products deployed in an enterprise. A product is exposed to TIBCO Enterprise Administrator with the help of an agent. TIBCO Enterprise Administrator agent for TIBCO Security Server, henceforth called the TEA agent in the rest of the document, can be used to administer, manage, and monitor TIBCO Security Server.

The TIBCO Enterprise Administrator server uses the agent to communicate with TIBCO Security Server. In some cases, you need to explicitly register the agent, but this agent autoregisters itself with TIBCO Enterprise Administrator. On registration, the agent exposes the artifacts of TIBCO Security Server to TIBCO Enterprise Administrator. In case, you need to re-register the agent, you can do so by going through the instructions in [Registering the TEA Agent for TIBCO Security Server](#). In addition, you can also reconnect an agent or unregister an agent. For additional details about using TIBCO Enterprise Administrator, see *TIBCO Enterprise Administrator User Guide*.

TIBCO Enterprise Administrator Administrative Interfaces

The agent provides three distinct user interfaces to communicate with the TIBCO Enterprise Administrator server : Web UI, command Line -based Shell interface, and Python Scripting.

1. Web UI: The TIBCO Enterprise Administrator server provides a default UI to manage and monitor products. TIBCO Security Server is shown as a card in the Web UI. You can drill down the product to see the artifacts of the product. You can then administer and monitor the product from the TIBCO Enterprise Administrator Web UI.
2. Shell Interface: The TIBCO Enterprise Administrator server provides a command-line utility called the TIBCO Enterprise Administrator shell. It is a remote shell, based on the SSH protocol. The shell is accessible using any terminal program such as Putty. The scripting language is similar to bash from UNIX, but has important differences. You can use the shell to perform almost all the tasks offered by the server UI.
3. Python Scripting: You can use Python scripting to perform any activity you performed using the Web UI. Python scripting is especially useful when you have to repeat a task for multiple users or use control structures to work through some conditions in your environment. Although you can use the shell utility to use the command-line UI, the shell UI does not support conditional statements and control structures. Python scripting proves to be useful in such cases.

This document discusses the Web UI interface. However, you can use the other two interfaces to achieve just about anything that you achieved using the Web UI. For more information on using these interfaces, refer to *TIBCO Enterprise Administrator User's Guide*.

Starting the TIBCO Security Server Agent

Prerequisites

Ensure that the TIBCO Enterprise Administrator server is running.

Procedure

1. Navigate to `TIBCO_HOME\tea\agents\tss\<version>\bin`.
2. Run `tss-agent.exe`



If you do not start from the `TIBCO_HOME\tea\agents\tss\<version>\bin` folder, then
run `tss-agent --propFile tss-agent.tra`

3. Verify whether or not the agent has started by logging in to TIBCO Enterprise Administrator.
Follow the steps outlined in [Log in to TIBCO Enterprise Administrator](#).

Log in to TIBCO Enterprise Administrator

You can use the Web UI to connect to the TIBCO Enterprise Administrator server.

Prerequisites

You must start the TIBCO Enterprise Administrator server before logging into the Web UI. Open the command prompt and navigate to `<TIBCO_HOME>`. Run `<TIBCO_HOME>\tea\<version>\bin\tea.exe`. You must also start the TIBCO Security Server agent to monitor the product on the Web UI.

Password Policies

Following are the constraints on creating a password:

- The length of the password should be between 1 and 128 characters.
- You cannot reuse the past 5 passwords.
- Your account gets locked after 10 failed attempts. The `admin` account is the only exception to this rule, but the `admin` account experiences a lag of 1 second on every login after 10 failed attempts.
- You must reset a password after a lockout because it cannot be changed.

Procedure

1. Open a browser and navigate to the URL `http://localhost:8777/tea/`, where `localhost` is the default hostname and `8777` is the default port number.



The default port number and other settings can be changed by modifying the settings in `tss.conf` file that is available under `<TIBCO_CONFIG_HOME>\tibco\cfgmgmt\tss\conf`.

2. Enter your Login credentials.

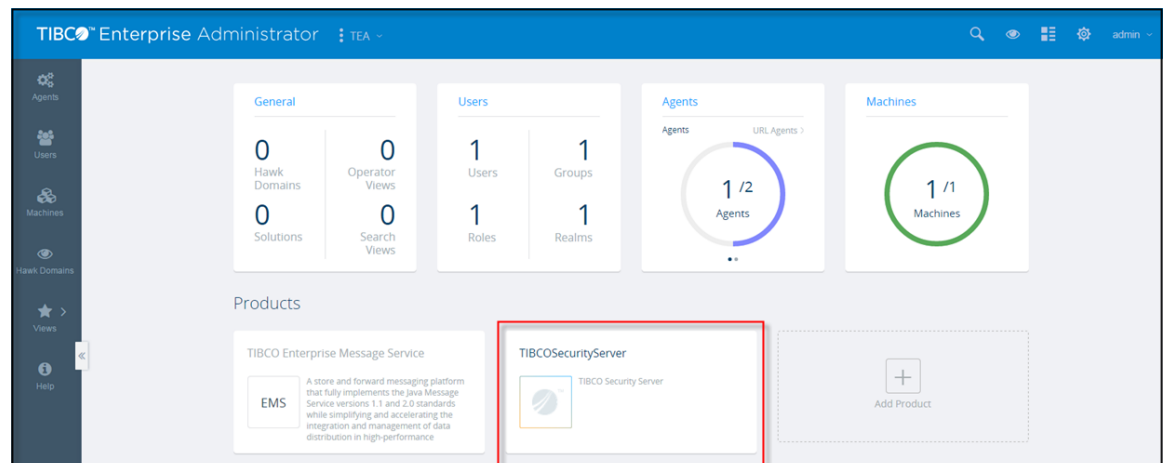
The default username is `admin` and the default password is `admin`.

On successful authentication, the landing page is displayed. The username with which you have logged in is shown as a menu option in the title pane. The landing page displays cards with information on the general details, users, agents, machines, and products exposed to the TIBCO Enterprise Administrator server. The Products pane shows TIBCOSecurityServer as one of the products registered with TIBCO Enterprise Administrator.



The default timeout for a session is 30 minutes.

Landing Page



Agent Management

You can use the Web UI to register TIBCO Enterprise Administrator agents and URL agents. The URL agents are not TIBCO Enterprise Administrator agents; however they might be a web application that you want to port as TIBCO Enterprise Administrator agent . URL agents might be some Every agent that gets added to the TIBCO Enterprise Administrator is displayed on the landing page. You can perform basic administrative tasks collectively on these agents such as reconnecting or unregistering agents.

Registering the TEA Agent for TIBCO Security Server

The TEA agent for TIBCO Security Server auto-registers itself with the TIBCO Enterprise Administrator server using the TIBCO Enterprise Administrator server URL specified in the `tss.conf`. However, there may be circumstances, when you have to explicitly register the agent. For example, if you want to register the TEA agent with any other TIBCO Enterprise Administrator server than the one in `tss.conf`, you must explicitly register agent with the URL `http://<hostname/IPAddress>:<portno>/teaagent/ssa`

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the TEA agent for TIBCO Security Server are running.

Procedure

1. Log in to TIBCO Enterprise Administrator. The username and the password is admin. The landing page is displayed.
2. Click the Agents card on the right. The Agent Management Pane is displayed.
3. Select the TEA agent and click **Register New**.
4. Provide the following details:

- Agent Name
- Agent URL :The default URL is `http://localhost:8080/teaagent/ssa` if TIBCO Enterprise Administrator server and the agent are on same machine. If you have changed the host name and port number in `tss.conf` then the URL is `http://<host_name>:<port_number>/teaagent/ssa`.
- Agent Description



When registering agents, ensure that the agent IDs do not collide.

5. Click **Register**. The TEA agent is visible in the Agent Management pane. The landing page also shows an icon for every registered agent. Your landing page will look different from the one displayed depending on the agents registered.



Watch out for the following:

1. Ensure that you avoid registering two agents with the same ID. The TIBCO Enterprise Administrator server does not validate whether two agents have registered with the same ID.
2. If there are two agents for the same object type, ensure that they have the same operation name and number. This is to ensure that when you invoke an operation, you can select the agent on which you want to execute the operation from the drop-down list.
3. If the agent is not immediately visible in the pane, try refreshing the browser.
4. If the URL used during registration is invalid, the "404-Page not Found" error occurs.
5. Make sure that TEA agent is configured with correct hostname that is reachable from TIBCO Enterprise Administrator server.



The scope of the document limits to a discussion on the TEA agent. This document does not cover the TIBCO Enterprise Administrator Web UI in detail. If you want to learn to use the three interfaces provided by TIBCO Enterprise Administrator, refer to *TIBCO Enterprise Administrator User's Guide*.

Reconnecting an Agent

You can collectively reconnect agents using TIBCO Enterprise Administrator.

Prerequisites

Ensure that the TEA server and the Admin agent are running.

Procedure

1. Click the **Agents** card.
The Agent Management Pane is displayed.
2. From the **Agents** tab, select the agents you want to reconnect. Click **Reconnect**.
A confirmation window is displayed.
3. Click **Reconnect** to confirm.
The agents are reconnected with the server.

Unregistering an Agent

You can collectively unregister agents using the TIBCO Enterprise Administrator.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agents are running.

Procedure

1. Click the Agents card.
The Agent Management pane is displayed.
2. From the Agents tab, select the agents you want to unregister. Click **Unregister**.
A confirmation window is displayed.
3. Click **Unregister** to confirm.
The agents are unregistered from the server.

TIBCO Security Server Management

The TEA agent for TIBCO Security Server provides ways of managing the security server. You can use the Credential Authority services and Resource Manager service to manage various security configurations.

The TIBCOSecurityServer pane provides you option to perform the following tasks:

1. Register a Credential Authority service
2. Unregister a Credential Authority service
3. Configure the Credential Authority Service servers
4. Set various security configurations using Resource Manager

Credential Authority Service

The Credential Authority Services provides a credential server that acts as the certificate server providing certificates to the consumers. The consumers can use the certificate for various reasons such as signing, encrypting, SSL connection and so on. .

You can request certificates by issuing a Certificate Signing Request (CSR) . A CSR contains the consumer's public certificate which needs to be digitally signed by the credential server. After validating the CSR, the server issues the signed certificate to the consumers

Registering the Credential Authority Service

You must register the Credential Authority service with the TIBCO Enterprise Administrator server.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCOSecurityServer** card.
3. Click **register**.
4. Provide the following details:
 1. From the drop-down box, select operation target.
 2. url : On your local machine, the URL is `http://localhost:8080/tcs`.
5. Click **register** to register the Credential Authority service.

Unregistering the Credential Authority service

The Credential Authority service can be explicitly unregistered from the TIBCO Enterprise Administrator server.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCOSecurityServer** card.
3. Click **unregister**.
4. Provide the following details:
 1. From the drop-down box, select operation target.
 2. url
5. Click **unregister** to unregister the Credential Authority service.

Configuring the Credential Authority Service

Before using the services offered by the credential server, it is important that you configure the credential server.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCOSecurityServer** card.
3. Click **Credential Authority Service(s)**
4. Click **setConfiguration**.
5. Provide the following optional basic details:
 1. commonName: Issuer name of the credential server.
 2. orgUnit
 3. org
 4. city
 5. state
 6. country

Additionally, provide the following details:

| Property | Description |
|---------------------------------|---|
| serverCertificateValidityPeriod | Required. The validity period for the Credential Authority Service server's own certificate. Validity period is in Days. |
| clientCertificateValidityPeriod | Required. The validity period for the certificates issues by Credential Authority Service server. Validity period is in Days. |

| Property | Description |
|------------------|--|
| keySize | Required field only if keyStore location is specified. The size of the generated key. The recommended size is 1024. |
| keyAlgo | Required field only if keyStore location is specified. The key algorithm. The recommended algorithm is RSA. |
| keySignAlgo | Required field only if keyStore location is specified. The Signature algorithm used to sign the request. The recommended algorithm is SHA1WithRSA. |
| keyStoreLocation | Optional. Point to the location of the keystore. |
| keyStorePassword | Required field only if keyStore location is specified. |
| keyStoreType | Required field only if keyStore location is specified. Some examples of the keystore Type are JCEKS, JKS, PKCS12. |
| keyStoreProvider | Optional. Some names of the keyStoreProvider are: 1. SunJCE (JCEKS format) 2. SUN (JKS format) 3. IBMJCE (IBM JREs) 4. SunJSSE (PKCS12 format) |
| keyAlias | Required field only if keyStore location is specified. |
| keyPassword | Required field only if keyStore location is specified. |
| hostname | Required. The name of the host. |
| portno | Required. The port number that the host listens at. Make sure that the port is available. |
| Enable SSL | Optional. Select this option to enable SSL. |

6. Click **setConfiguration**.



You can create a credential authority service using the python scripts available under `TIBCO_HOME\tea\agents\tss\<version>\samples\credentialAuthorityService`.

Result

After configuring the credential server, it creates the credential store by itself if no keystore is specified in the configuration property. If not, it uses the keystore specified and stores it in the database.



Re-configuring the credential server changes the credential store. As a result, the previously issued certificates previously get invalidated.

Starting the Credential Authority Service

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server. Ensure that you have configured the credential server before starting it.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCOSecurityServer** card.
3. Click **Credential Authority Service(s)**.
4. From the servers pane, select the **Credential Authority Service**.
5. Click **start**.
6. At the confirmation window, click **start** to start the server.

Stopping the Credential Authority Service

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server. Ensure that the credential authority service has started.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCOSecurityServer** card.
3. Click **Credential Authority Service(s)**.
4. From the servers pane, select the **Credential Authority Service**.
5. Click **stop**.
6. At the confirmation window, click **stop** to stop the server.

Resource Manager Service

The Resource Manager Service provides a set of resources that provide access to various types of security providers: identity, credential, and authentication.

Identity, credential, and authentication providers enable clients and servers to assert and establish identity. Identity provider provides access based on credentials or token. Services such as Subject Identity Provider provides access to a username and password stored in a keystore. The Trust provider maintains the identity of a trusted resource.

The Credential Service provider provides access to private credentials and public certificates. Password Credential Resource provides a mechanism of storing username and passwords. Keystore Credential Resource stores passwords in a keystore. The Login Credential Resource is used to generate key alias and key passwords.

The Authentication providers enable connections to authentication services such as LDAP Authentication, Kerberos Authentication, SiteMinder, and WSS Authentication.

Every resource manager service comes with create, update, and delete options.

Creating Password Credential Resources

Using Password Credential Resources, you can store passwords as secret keys in a keystore. This provider can be configured with a set of user names and passwords, and a secret key.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCOSecurityServer** card.
3. Click **Resource Manager Service**.
4. From the Password Credential Resources pane, click **create**.



Alternatively, click **Password Credential Resources**, and in the following page, click **create**.

5. Provide the following details:

| Property | Description |
|----------------------|---|
| Name | Required. Specify the name. |
| protection parameter | Required. Specify protection parameter to add an extra layer of protection. |
| tokenMap | Required. Specify a token map. |

6. Click **create** to create a password credential resource.



You can create a password credential resource using python scripts available under `TIBCO_HOME\tea\agents\tss\<version>\samples\resourceManagerService\password`.

Creating Login Credential Resources

To keep your username/password secure, you can mask them using a key alias and key password combination. For example, a database administrator might not want to share his credentials with another user while trying to give them access to the database. In such cases, a Login Credential Resource can come handy.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.

2. From the landing page, click **TIBCOSecurityServer** card.
3. Click **Resource Manager Service**.
4. From the Login Credential Resources pane, click **create**.



Alternatively, click **Login Credential Resources**, and in the following page, click **create**.

5. Provide the following details:

| Property | Description |
|-------------|--|
| keyAlias | Required. Name of the key alias. |
| keyPassword | Required. The key password. |
| username | Required. The user name that you want to mask. |
| password | Required. The password that you want to mask. |

6. Click **create** to create a Login Credential resource.



You can create a Login Credential resource using python scripts available under `TIBCO_HOME\tea\agents\tss\<version>\samples\resourceManagerService\login`.

Creating Keystore Credential Resources

By creating a Keystore Credential Resource, you can store encrypted passwords on a keystore. The Keystore Credential Resource provides the ability to specify a keystore (PKCS#12, JKS, JCEKS) containing public certificates, private keys, and secret keys (passwords) for use by the Identity Trust and Identity Subject service providers.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCOSecurityServer** card.
3. Click **Resource Manager Service**.
4. From the Keystore Credential Resources pane, click **create**.



Alternatively, click **Keystore Credential Resources**, and in the following page, click **create**.

5. Provide the following details:

| Property | Description |
|--------------|-----------------------------------|
| Name | Required. Name of the keystore. |
| keyStoreFile | Required. Choose a keystore file. |

| Property | Description |
|-------------------------|---|
| keyStoreLocation | Required. Point to the location of the keystore. |
| keyStorePassword | Required field only if keyStore location is specified. |
| keyStoreType | Required field only if keyStore location is specified. Some examples of the keystore Type are JCEKS, JKS, PKCS12. |
| keyStoreProvider | Optional. Some names of the keyStoreProvider are: <ol style="list-style-type: none"> 1. SunJCE (JCEKS format) 2. SUN (JKS format) 3. IBMJCE (IBM JREs) 4. SunJSSE (PKCS12 format) |
| keystoreRefreshInterval | Required. Time in millisecond to refresh the keystore. |
| keystoreCacheLocation | Optional. Point to the location of the keystore cache. |
| sslIdentityProvider | Optional. Name of the SSL Identity Provider. |

6. Click **create** to create a Keystore Credential resource.



You can create a Keystore Credential resource using python scripts available under `TIBCO_HOME\tea\agents\tss\<version>\samples\resourceManagerService\keystore`.

Creating Kerberos Authentication Resources

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCO Security Server** card.
3. Click **Resource Manager Service**.
4. From the Kerberos Authentication Resources pane, click **create**.



Alternatively, click **Kerberos Authentication Resources**, and in the following page, click **create**.

5. Enter the values for the fields listed in [Kerberos Authentication Resources](#).

6. Click **create** to create a Kerberos Authentication resource.



You can create a Kerberos Authentication resource using python scripts available under `TIBCO_HOME\tea\agents\tss\<version>\samples\resourceManagerService\kerberos.`

Kerberos Authentication Resources

The Kerberos Authentication resource template represents a Kerberos authentication service.

| Property | Required? | Editable? | Accepts SVARs? | Description |
|------------------------------|-----------|-----------|----------------|--|
| Name | | | | |
| enableSecurityTokenAttribute | N | Y | N | Controls if the security token that was authenticated should be included in the AttributeStatement of the issued SAML assertion. Default: Checked. |
| enableSAMLAttributesPurge | | | | Controls if AttributeStatements of the authenticated assertion should be included in the AttributeStatements of the issued SAML assertion. Default: Checked. |
| enableHolderOfKeyAssertion | | | | Controls if Holder-of-Key Subject Confirmation method should be used in the issued SAML assertion. Select one of the following security token types: <ul style="list-style-type: none"> • SAML 1.1 Token 1.1 • SAML 2.0 Token 1.1 • |
| samlValiditySeconds | N | Y | Y | The duration of the validity of the SAML tokens. Default: 600 s. |
| tokenSigningService | N | Y | Y | The name of an Identity Provider resource that identifies the signer of the SAML tokens. |
| clockskew | Y | N | Y | The maximum allowable amount of clock skew before a Kerberos message is assumed to be invalid. Default: 600. |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|----------------|-----------|-----------|----------------|---|
| dnsLookupKdc | Y | N | N | Indicate whether DNS SRV records should be used to locate the KDCs and other servers for a realm, if the KDC is not the default realm. Default: Checked. |
| dnsLookupRealm | Y | N | N | Indicate whether DNS TXT records should be used to determine the Kerberos realm of a host if it is not the default realm. Default: Unchecked. |
| defaultDomain | Y | Y | Y | The default DNS domain to which the Kerberos realm belongs. Default: None. |
| ticketLifeTime | Y | N | Y | The lifetime for initial tickets. Default: 24. |
| renewLifeTime | Y | N | Y | The renewable lifetime for initial tickets. Default: None. |
| noAddresses | Y | N | N | Indicate that initial Kerberos ticket will be addressless. Default: Checked. |
| forwardable | Y | N | N | Indicate that initial Kerberos ticket will be forwardable. Default: Unchecked. |
| proxiable | Y | N | N | Indicate that initial Kerberos ticket will be proxiable. Default: Checked. |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|-----------------------------------|-----------|-----------|----------------|---|
| krb5ConfFileLocationOption | N | Y | N | <p>The method for specifying the location of the Kerberos configuration file. One of:</p> <ul style="list-style-type: none"> System Specific Default Location - Use the system-specific default location. Custom Configuration File - Use a custom configuration file. Enables the Custom Configuration File Name field. Generated - Use a generated configuration file. Enables the Generated Configuration File field and all other fields whose values are used in generating the configuration file. <p>Default: System Specific Default Location.</p> |
| Realm | N | Y | N | <p>The Kerberos realm.</p> <p>Default: None.</p> |
| kdc | N | Y | N | <p>The Kerberos key distribution center.</p> <p>Default: None.</p> |
| krb5ConfFileLocation | Y | Y | Y | <p>The fully-qualified path to the configuration file.</p> <p>Default: None.</p> |
| autoGeneratedKrb5ConfFileLocation | Y | Y | Y | <p>The fully-qualified path to which the generated configuration file is saved.</p> <p>Default: None.</p> |
| storeKey | Y | N | N | <p>Indicate that the principal's key should be stored in the subject's private credentials.</p> <p>Default: Checked.</p> |
| doNotPrompt | | | | |
| refreshKrb5Config | Y | N | N | <p>Indicate that you want the configuration to be refreshed before the login authentication method is invoked.</p> <p>Default: Unchecked.</p> |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|----------------|-----------|---------------------------------|----------------|--|
| renewTGT | Y | N | N | Indicate that you want to renew ticket granting tickets. If checked, the Use Ticket Cache checkbox is checked and the Ticket Cache Name field is enabled. Default: Unchecked. |
| useTicketCache | Y | N | N | Indicate that you want the ticket granting tickets to be obtained from the ticket cache. Default: Unchecked. |
| ticketCache | Y | When useTicketCache is checked. | Y | The name of the ticket cache that contains ticket granting tickets. Default: None. |
| useKeyTab | Y | N | N | Indicate that the principal's key should be obtained from the keytab. When checked, the Keytab Filename field is enabled. If Keytab Filename field is not set, the keytab is obtained from the Kerberos configuration file. Default: Unchecked. |
| keyTab | Y | When useKeyTab is checked. | Y | The file name of the keytab. Default: None. |
| principal | Y | N | Y | The name of the principal. Default: None. |

Creating LDAP Authentication Resources

The LDAP Authentication Resource helps you create an authentication mechanism using LDAP.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCO Security Server** card.
3. Click **Resource Manager Service**.
4. From the LDAP Authentication Resources pane, click **create**.



Alternatively, click **LDAP Authentication Resources**, and in the following page, click **create**.

5. Enter the values for the fields listed in [LDAP Authentication Resources](#).
6. Click **create** to create a LDAP Authentication resource.



You can create a LDAP Authentication resource using python scripts available under `TIBCO_HOME\tea\agents\tss\<version>\samples\resourceManagerService\ldap`.

LDAP Authentication Resources

The LDAP Authentication resource template represents an LDAP server providing authentication services.

LDAP authentication is done in one of the following ways:

- **Bind mode** — The bind mode authenticates (binds) each user's Distinguished Name (DN) and password to the LDAP server. In this case, you can use the DN Template field so that users do not have to provide their whole DN. For example, a DN Template of `uid={0},OU=Department,DC=company,DC=com` allows users to type in only their uid and the RI will use the template to create the DN.
- **Search mode** — In the search mode, a connection binds as the administrative user. It then searches for the given users and authenticates their found DN and passwords with the LDAP server. In this case, you need to provide the credentials of such an administrative user by checking Log in as Administrator.

| Property | Required? | Editable? | Accepts SVARs? | Description |
|------------------------------|-----------|-----------|----------------|---|
| Name | | | | |
| enableSecurityTokenAttribute | N | Y | N | Controls if the security token that was authenticated should be included in the AttributeStatement of the issued SAML assertion. Default: Checked. |
| enableSAMLAttributesPurge | | | | Controls if AttributeStatements of the authenticated assertion should be included in the AttributeStatements of the issued SAML assertion. Default: Checked. |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|----------------------------|-----------|-----------|----------------|--|
| enableHolderOfKeyAssertion | | | | <p>Controls if Holder-of-Key Subject Confirmation method should be used in the issued SAML assertion.</p> <p>Select one of the following security token types:</p> <ul style="list-style-type: none"> • SAML 1.1 Token 1.1 • SAML 2.0 Token 1.1 • |
| samlValiditySeconds | N | Y | Y | <p>The duration of the validity of the SAML tokens.</p> <p>Default: 600 s.</p> |
| tokenSigningService | N | Y | Y | <p>The name of an Identity Provider resource that identifies the signer of the SAML tokens.</p> |
| initialCtxFactory | N | Y | Y | <p>The factory object that provides the starting point for resolution of names within the LDAP server.</p> <p>Default: com.sun.jndi.ldap.LdapCtxFactory.</p> |
| serverURL | Y | Y | Y | <p>A space-separated list of URLs for an LDAP server. To achieve fault tolerance, you can specify URLs. For example, ldap://server1.example.com:686 ldap://server2.example.com:1686.</p> <p>Default: ldap://localhost:389.</p> |
| searchTimeOut | N | Y | Y | <p>The time to wait for a response from the LDAP directory server.</p> <p>Default: -1, which means to wait forever.</p> |
| userAttributeUserName | N | Y | Y | <p>The name of the LDAP attribute from which the user display name can be obtained. Always specify an Attribute Name even though this field is labeled optional.</p> <p>You must use an attribute that is part of the LDAP schema. Otherwise, any attribute not defined by the schema can result in an error.</p> <p>Default: None</p> |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|--------------------------|-----------|-----------|----------------|---|
| userAttributeGroupsName | Y | Y | Y | The name of the attribute in each user object that lists the groups to which the user belongs. Default: None. |
| userAttributesExtra | N | Y | Y | Optional list of user attributes to retrieve from the LDAP directory during authentication. Default: None. |
| groupAttributeGroupsName | Y | Y | Y | The name of the attribute in the group object that contains the name of the group. For example, for OpenLDAP: cn, for ActiveDirectory:sAMAccountName. Default: None. |
| userSearchBaseDN | Y | Y | Y | Base distinguished name from which the search starts. Example: ou=department, dc=company, dc=com. |
| userSearchScopeSubtree | | | | |
| userSearchExpression | N | Y | Y | The expression used for searching a user. An example for this expression is (CN={0}). '{0}' is replaced by the username being searched for. You can define any complex filter like (&(cn={0}) (objectClass=account)). Default: &(objectClass=person)(uid={0}) |
| groupSearchBaseDN | N | Y | Y | Searches for groups beginning at this base distinguished name (DN). Default: None. |
| enableNestedGroupSearch | | | | |
| groupSearchExpression | Y | Y | Y | Search by matching this expression against potential groups. Default: None. |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|----------------------------|-----------|-----------|----------------|--|
| groupSearchScopeSubtree | N | N | N | <p>Search the entire subtree starting at the base DN for groups (default). Otherwise, search only the nodes one level below the base DN.</p> <p>Default: Checked.</p> |
| groupIndication | N | Y | N | <p>Specifies how a user's group memberships are found. Group information is used by Administrator when a user, once authenticated, performs other activities in the system.</p> <p>Options:</p> <ul style="list-style-type: none"> Group has users A list of users that belong to the group. User has groups A list of groups to which the user belongs. User DN has groupsThe DN with a list of groups to which the user belongs. No Group Info Group memberships are not handled. <p>If the selected value is User has groups or User DN has groups, the Users Attribute with Group Names field displays.</p> <p>If the selected value is Group has users, the following fields display:</p> <ul style="list-style-type: none"> Group Search Base DN Group Search Expression Group Attribute with User Names Group Attribute with Group Name Group Attribute with Subgroup Names Group Search Scope Subtree <p>Default: No Group Info.</p> |
| groupAttributeSubgroupName | N | Y | Y | <p>The name of the attribute in the group object that contains its subgroups. For example, for OpenLDAP: uniqueMember, for ActiveDirectory: member.</p> <p>Default: None.</p> |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|-------------------------|-----------|-----------|----------------|--|
| groupAttributeUsersName | Y | Y | Y | The name of the attribute in the group object that contains its users. For example, for OpenLDAP: uniqueMember, for ActiveDirectory: member. Default: None. |
| userDNTemplate | Y | Y | Y | The template by which the User DN, used to bind to the LDAP server, is generated. Because the full DN is always supplied, the template should always contain {0} which gets replaced with the actual username. Default: {0} |
| connectionPools | | | | |
| securityAuthentication | N | Y | Y | Value of Simple Authentication and Security Layer (SASL) authentication protocol to use. Values are implementation-dependent. Some possible values are simple, none, md-5. Default: Blank. |
| followReferrals | N | Y | N | Indicate whether the client should follow referrals returned by the LDAP server. Default: Unchecked. |
| sslIdentityProvider | | | | The name of the Identity Trust provider resource for establishing SSL connection to the LDAP server. |
| credentialProvider | | | | The name of the Credential Keystore or Credential Password provider resource containing the LDAP login credentials. This option requires a keyAlias and keyPassword to also be specified. This option can be used in place of the adminIdentityProvider setting. |
| adminIdentityProvider | | | | The name of the Identity Subject provider resource containing the LDAP login credentials. This option can be used in place of the credentialProvider/keyAlias/keyPassword setting tuple. |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|-------------|-----------|-----------|----------------|--|
| keyPassword | Y | Y | Y | The password protecting the key entry. Default: None |
| keyAlias | Y | Y | Y | Alias of the user's key entry in the keystore managed by the keystore provider. Default: None |

Creating SiteMinder Authentication Resources

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCO Security Server** card.
3. Click **Resource Manager Service**.
4. From the SiteMinder Authentication Resources pane, click **create**.



Alternatively, click **SiteMinder Authentication Resources**, and in the following page, click **create**.

5. Enter the values for the fields listed in [SiteMinder Resources](#).
6. Click **create** to create a SiteMinder Authentication resource.



You can create a SiteMinder Authentication resource using python scripts available under `TIBCO_HOME\tea\agents\tss\<version>\samples\resourceManagerService\siteminder`.

SiteMinder Authentication Resources

The SiteMinder Authentication resource template represents a SiteMinder authentication service.

| Property | Required? | Editable? | Accepts SVARs? | Description |
|----------|-----------|-----------|----------------|-------------|
| Name | | | | |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|--------------------------------|-----------|-----------|----------------|---|
| enableSecurityTokenAttribute | N | Y | N | Controls if the security token that was authenticated should be included in the AttributeStatement of the issued SAML assertion. Default: Checked. |
| enableSAMLAttributesPurge | | | | Controls if AttributeStatements of the authenticated assertion should be included in the AttributeStatements of the issued SAML assertion. Default: Checked. |
| enableHolderOfKeyAssertion | | | | Controls if Holder-of-Key Subject Confirmation method should be used in the issued SAML assertion. Select one of the following security token types: <ul style="list-style-type: none"> • SAML 1.1 Token 1.1 • SAML 2.0 Token 1.1 • |
| samlValiditySeconds | N | Y | Y | The duration of the validity of the SAML tokens. Default: 600 s. |
| tokenSigningService | N | Y | Y | The name of an Identity Provider resource that identifies the signer of the SAML tokens. |
| smHostConfigFileLocationOption | N | Y | N | The method for specifying the location of the SiteMinder configuration file. <ul style="list-style-type: none"> • System Specific Default Location - Use the system-specific default location. • Custom File Location - Use a custom configuration file. Enables the Custom Configuration File Name field. • Generate - Use a generated configuration file. Enables the Generated Configuration File field and all other fields whose values are used in generating the configuration file. Default: System Specific Default Location. |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|-------------------------------------|-----------|-----------|----------------|--|
| smHostConfFileLocation | Y | Y | Y | The path to the configuration file. Default: None. |
| autoGeneratedSmHostConfFileLocation | Y | Y | Y | The path to which the generated configuration file is saved. Default: None. |
| hostName | Y | Y | Y | The name of the host. Default: None. |
| sharedSecret | Y | Y | Y | The host's shared secret. Default: None. |
| sharedSecretTime | Y | N | Y | The validity period for the shared secret. Default: None. |
| hostConfigObject | Y | Y | Y | The host's configuration object name. Default: None. |
| policyServer | Y | Y | Y | The URLs of the SiteMinder Policy Server. Default: None. |
| requestTimeout | Y | N | Y | The request timeout. Default: 60 s. |
| cryptoProvider | Y | N | Y | The name of the crypto provider. Default: None. |
| fipsMode | Y | N | N | <p>The FIPS mode for the crypto provider.</p> <ul style="list-style-type: none"> • FIPS-Compatibility Mode - the environment uses existing SiteMinder algorithms to encrypt sensitive data. • FIPS-Migration Mode - the SiteMinder Policy Server continues to use existing SiteMinder encryption algorithms as you migrate the environment to use only FIPS-compliant algorithms. • FIPS-only Mode - the environment only uses FIPS-compliant algorithms to encrypt sensitive data. <p>Default: None.</p> |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|-----------------|-----------|-----------|----------------|---|
| agentName | Y | Y | Y | The name of the SiteMinder agent that enforces access control policies provided by the Policy Server. Default: None. |
| resource | Y | N | Y | The name should match the corresponding value specified in the policy set or it should be left blank. Default: None. |
| clientIPAddress | Y | N | Y | The IP address of the machine on which the SiteMinder agent is installed. Default: None. |

Creating WSS Authentication Resources

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCO Security Server** card.
3. Click **Resource Manager Service**.
4. From the WSS Authentication Resources pane, click **create**.



Alternatively, click **WSS Authentication Resources**, and in the following page, click **create**.

5. Enter the values for the fields listed in [WSS Authentication Resources](#).
6. Click **create** to create a WSS Authentication resource.



You can create a WSS Authentication resource using python scripts available under `TIBCO_HOME\tea\agents\tss\<version>\samples\resourceManagerService\wss`.

WSS Authentication Resources

A WS-Security ASP resource template enables a connection to Web Services Security authentication services.

| Property | Required? | Editable? | Accepts SVARs? | Description |
|--------------------------------|-----------|-----------|----------------|--|
| Name | | | | |
| enableSecurityTokenAttribute | N | Y | N | Controls if the security token that was authenticated should be included in the AttributeStatement of the issued SAML assertion. Default: Checked. |
| enableSAMLAttributesPurge | | | | Controls if AttributeStatements of the authenticated assertion should be included in the AttributeStatements of the issued SAML assertion. Default: Checked. |
| enableHolderOfKeyAssertion | | | | Controls if Holder-of-Key Subject Confirmation method should be used in the issued SAML assertion. Select one of the following security token types: <ul style="list-style-type: none"> • SAML 1.1 Token 1.1 • SAML 2.0 Token 1.1 |
| samlValiditySeconds | N | Y | Y | The duration of the validity of the SAML tokens. Default: 600 s. |
| tokenSigningService | N | Y | Y | The name of an Identity Provider resource that identifies the signer of the SAML tokens. |
| signatureValidationService | N | Y | N | Indicate whether to verify the signatures. If checked, activates the Trust Provider field. Default: Unchecked. |
| kerberosTokenValidationService | N | N | N | Kerberos is a secure method for authenticating a request for a service in a computer network. |

| Property | Required? | Editable? | Accepts SVARs? | Description |
|--------------------------------|-----------|-----------|----------------|--|
| usernameTokenValidationService | N | N | N | Indicate whether to verify the username. If checked, activates the Authentication Provider field. Default: Unchecked. |
| groupSelectorExpression | | | | |
| wssBspCompliant | | | | |

Creating Subject Identity Resources

The Subject Identity Provider is used for obtaining and using private credentials obtained from a credential store.

The Subject Identity Provider needs the following:

- Requires a trust store for SSL client connections and signature verification.
- Requires a credential store for SSL server and SSL mutual authentication and for creating digital signature.
- Requires a private keystore for creating digital signature.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server. Ensure that the Keystore Credential Resource and Login Credential resource is configured.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCO Security Server** card.
3. Click **Resource Manager Service**.
4. From the Subject Identity Resources pane, click **create**.



Alternatively, click **Subject Identity Resources**, and in the following page, click **create**.

5. Provide the following details:

| Property | Description |
|-------------|---|
| name | Required. Name of the Subject Identity Provider. |
| sslProtocol | Optional. The name of the SSL Protocol such as TLSv1. |
| sslProvider | Optional. The name of the SSL Provider. |

| Property | Description |
|------------------------------|--|
| sslCipherStrength | Optional. The cipher strength is the number of bits in the key used to encrypt data. The greater the number of bits in the key (cipher strength), the more possible key combinations and the longer it would take to break the encryption. The cipher strength should be at least 128 bits. |
| sslExplicitCiphers | Optional. Explicit Ciphers are enabled when SSL Cipher Class is set to Explicit Ciphers. |
| sslHostName | Optional. Name of the SSL Host. |
| sslVerifyHost | Optional. Select this option to verify SSL Host. |
| sslVendor | Optional. Name of the SSL vendor. |
| kerberosServiceProvider | Optional. Name of the Kerberos Service Provider. |
| kerberosServicePrincipalName | Optional. The name of a Kerberos client principal . Specify this information to gain access to the private key of the client principal. |
| wssEncryptionAlgorithm | Optional. The WSS encryption algorithm. By default it is AES_128. |
| wsskeyEncryptionAlgorithm | Optional. The WSS key encryption algorithm. By default it is RSAOEP. |
| wssBspCompliant | Optional. Select this option to make the resource wssbsp compliant. |
| wssStrictTimestamp | Optional. Select this option to enable WSS strict timestamp. |
| wssTimeStampTimeToLive | Optional. The time to live in seconds. |
| wssTimeStampFutureTimeToLive | Optional. The future time to live in seconds. |
| wssCertificateRevocationURL | Optional. The WSS Certificate revocation URL. |
| trustStoreServiceProvider | Required. The name of the keystore credential resource. |
| enableTrustStoreAccess | Required. By default this option is enabled. |
| sslExplicitlyTrustAllCAs | Optional. By default this option is enabled. |
| sslCertificateRevocationURL | Optional. The SSL Certificate revocation URL. |

| Property | Description |
|--|--|
| sslCertificateRevocationReloadInterval | Optional the reload interval for revoking the SSL certificate. |
| IdentityServiceProvider | Required. The name of the keystore credential resource. |
| keyAlias | Required. Name of the key alias. You can use the information captured by the Login Credential Resource. |
| keyPassword | Required. The Key password. You can use the information captured by the Login Credential Resource. |
| enableCredentialStoreAccess | Optional. Select this option to enable credential store access. |
| sslClientAuth | Optional. The SSL client authentication. |
| wssEnableProtectToken | Optional. This option enables protected tokens. By default, this option is selected. |
| kerberosPrincipal | Optional. The name of a Kerberos client principal. Specify this information to gain access to the private key of the client principal. |
| kerberosPrincipalPassword | Optional. The principal password for Kerberos. |
| wssSignatureAlgorithm | Optional. The WSS signature algorithm. By default, it is RSA_SHA256. |
| wssDigestAlgorithm | Optional. The WSS digest algorithm. By default, it is SHA256. |
| wssCanonAlgorithm | Optional. The WSS canon algorithm. By default, it is XML_EXC_C14N. |
| wssTimetoLive | Optional. The time to live in seconds. |

6. Click **create** to create a Subject Identity resource.



You can create a Subject Identity resource using python scripts available under `TIBCO_HOME\tea\agents\tss\<version>\samples\resourceManagerService\subject`.

Creating Trust Identity Resources

The Trust Identity Provider is used for obtaining certificates needed for performing trust operations from a credential store. This resource requires a trust store for SSL client and signature verification.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCO Security Server** card.
3. Click **Resource Manager Service**.
4. From the Trust Identity Resources pane, click **create**.



Alternatively, click **Trust Identity Resources**, and in the following page, click **create**.

5. Provide the following details:

| Property | Description |
|------------------------------|--|
| name | Required. Name of the Trust Identity Provider. |
| sslProtocol | Optional. The name of the SSL Protocol such as TLSv1. |
| sslProvider | Optional. The name of the SSL Provider. |
| sslCipherStrength | Optional. The cipher strength is the number of bits in the key used to encrypt data. The greater the number of bits in the key (cipher strength), the more possible key combinations and the longer it would take to break the encryption. The cipher strength should be at least 128 bits. |
| sslExplicitCiphers | Optional. Explicit Ciphers are enabled when SSL Cipher Class is set to Explicit Ciphers. |
| sslHostName | Optional. Name of the SSL Host. |
| sslVerifyHost | Optional. Select this option to verify SSL Host. |
| sslVendor | Optional. Name of the SSL vendor. |
| kerberosServiceProvider | Optional. Name of the Kerberos Service Provider. |
| kerberosServicePrincipalName | Optional. The name of a Kerberos client principal . Specify this information to gain access to the private key of the client principal. |

| Property | Description |
|--|--|
| wssEncryptionAlgorithm | Optional. The WSS encryption algorithm. By default it is AES_128. |
| wsskeyEncryptionAlgorithm | Optional. The WSS key encryption algorithm. By default it is RSAOEP. |
| wssBspCompliant | Optional. Select this option to make the resource wssbsp compliant. |
| wssStrictTimestamp | Optional. Select this option to enable WSS strict timestamp. |
| wssTimeStampTimeToLive | Optional. The time to live in seconds. |
| wssTimeStampFutureTimeToLive | Optional. The future time to live in seconds. |
| wssEnableSignatureConfirmation | Optional. Select this option to enable signature confirmation. |
| wssKeyType | Optional. The WSS key type. By default, the value is set toSKI_KEY_IDENTIFIER. |
| wssCertificateRevocationURL | Optional. The WSS Certificate revocation URL. |
| wssCertificateRevocationReloadInterval | Optional the reload interval for revoking the WSS certificate. |
| trustStoreServiceProvider | Required. The name of the keystore credential resource. |
| enableTrustStoreAccess | Required. By default this option is enabled. |
| sslExplicitlyTrustAllCAs | Optional. By default this option is enabled. |
| sslCertificateRevocationURL | Optional. The SSL Certificate revocation URL. |
| sslCertificateRevocationReloadInterval | Optional the reload interval for revoking the SSL certificate. |

6. Click **create** to create a Trust Identity resource.



You can create a Trust Identity resource using python scripts available under *TIBCO_HOME* \tea\agents\tss\<version>\samples\resourceManagerService\trust.

Creating Credential Server Resource

The Credential Server Resource contains configuration details of Credential Authority Server.

Prerequisites

Ensure that the TEA server and the TEA agent for TIBCO Security Server are running. Ensure that you have started the Credential Authority Server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCO Security Server** card.
3. Click **Resource Manager Service**.
4. From the Credential Server Resources pane, click **create**.



Alternatively, click **Credential Server Resources**, and in the following page, click **create**.

5. Provide the following details:

| Property | Description |
|----------------------------|---|
| Name | Required. Name of the keystore. |
| credentialServerHost | Required. Hostname of the credential server. |
| credentialServerPort | Required. The port on which the credential server listens. Make sure it is available. |
| credentialServerURLs | Required. URLs of the credential server. |
| credentialProvider | Optional. Name of the credential provider. |
| keyAlias | Optional. The key alias name. |
| keysPassword | Optional. The key password. |
| credentialServerThumbprint | Optional. The thumb print of the credential server. |
| enableSSL | Optional. Select this option to enable SSL. |
| commonName | Optional. Common Name of the credential server. |

6. Click **create** to create a Credential Server resource.



You can create a Credential Server resource using python scripts available under `TIBCO_HOME\tea\agents\tss\<version>\samples\resourceManagerService\credentialServer`.

Updating a Resource

The procedure to update a resource is the same. The instructions here are for updating any resource under Resource Manager Service.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCO Security Server** card.

3. Click **Resource Manager Service**.
4. Select a resource of your choice.
5. From the members list, select a resource that you created.
6. Click **update** to modify the resource.
7. Make the necessary changes and click **update** to save the changes.

Deleting a Resource

The procedure to delete a resource is the same. The instructions here are for deleting any resource under Resource Manager Service.

Prerequisites

Ensure that the TIBCO Enterprise Administrator server and the agent are running. Ensure that the agent is registered with the server.

Procedure

1. Log in to TIBCO Enterprise Administrator.
2. From the landing page, click **TIBCO Security Server** card.
3. Click **Resource Manager Service**.
4. Select a resource of your choice.
5. From the members list, select a resource that you created.
6. Click **delete** to modify the resource.
7. At the confirmation window, click **delete** to delete the resource.