TIBCO BusinessConnect™ Container Edition

Security Guidelines

Version 1.3.0 | October 2023

Contents

Introduction

This document describes guidelines to ensure security within the various components of TIBCO BusinessConnect™ Container Edition. It also provides additional security-related guidance and recommendations for other aspects of internal and external communication. In particular, this document provides details of product connectivity and configuration of security options.

For information about how to upgrade third-party components and postinstallation activities, see the *TIBCO BusinessConnect Container Edition Installation and Deployment*.

Product Connectivity

The inbound and outbound data flow of TIBCO BusinessConnect Container Edition is as follows:

Inbound Data Flow

The gateway services such as HTTP/HTTPS/HTTPCA are hosted to receive B2B communications directly from the Internet with security features such as SSL and SSH. The firewall between the Gateway Server and the rest of your system protects against the threat of malicious communications. TIBCO BusinessConnect Container Edition Gateway Server is a standalone Java executable that is not dependent on TIBCO ActiveMatrix BusinessWorks. However, it still needs TIBCO Enterprise Message Service to communicate with the Interior Server.

TIBCO Enterprise Message Service is also used for transferring small size messages such as HTTPs service keys and so on between the Gateway Server and Interior Server.

For other FTP/FTPS/SSHFTP inbound pollers, Interior Server directly connects to the FTP server.

Outbound Data Flow

Interior Server directly sends out the requests to the Gateway Server.

This server is located inside the company's firewall and handles all messaging level activities, such as message packaging and unpacking, encryption and decryption, signature and verification, and so on, according to numerous transport and vertical business standards.

For detailed information on the data flow, see the *TIBCO BusinessConnect Container Edition Concepts*.
If you use secure transport to exchange data with the trading partner, there are many security configurations in TIBCO BusinessConnect Container Edition that can guarantee the secure data exchange. For more information on security configuration, see Security.

How to upgrade third-party components?

You need not to upgrade the third-party components that are packaged in the TIBCO BusinessConnect™ Container Edition except for the following JAR files:

- hibernate-c3p0-5.4.32.Final.jar

- hibernate-core-5.6.15.Final.jar

- hibernate-commons-annotations-5.1.2.Final.jar

- mysql-connector-j/8.0.32

- ojdbc8-21.1.0.0.jar

- fesi-1.1.5.jar

You can directly download the first three jars: hibernate-c3p0-5.4.12.Final.jar, hibernate-core-5.6.15.Final.jar, and hibernate-commons-annotations-5.1.2.Final.jar from the *TIBCO hosts* site. There are three HTML files referring to the download URLs.

The mysql-connector-java-8.0.32.jar or mysql-connector-java-8.0.32+.jar is required to be downloaded to use.

The ojdbc8-21.1.0.0.jar is the Oracle JDBC connector JAR file.

Hence, you are not required to upgrade the new versions of the third-party components.

The third-party components with the versions currently used in the TIBCO BusinessConnect Container Edition have very fewer vulnerability issues.

Post-Install Activities

During installation of the Kubernetes cluster and deploying the TIBCO BusinessConnect™ Container Edition, a master node and the worker nodes should not be accessed by any other user except the admin.

Only the admin user can access the Kubernetes cluster and have the read/write access to the Docker registry and the database tables.

Since the TIBCO BusinessConnect Container Edition services are deployed and run on the Kubernetes cluster, it should be secured to protect the possible security threats.

Where to Configure Security Options

When you configure the http service of the Gateway server, you can upload the credential for HTTPS/HTTPCA service.

Thus, your trading partner can use a secure channel to send you the request.

When you configure inbound FTPS/SSHFTP/Email poller, you can also specify credentials for secure transport.

Also, when you create outbound transport for a Partner, you can create secure transports like HTTPS, AS2_HTTPS, and FTPS and further use them to send out data to your partner.

Security

This topic provides you an overview of the security methods used in TIBCO BusinessConnect Container Edition to protect your business data and communications.

For more details on security methods, see the following:

- Public and Private Keys
- Digital Certificates
- Shadow Credentials
- Digital Signatures
- Encryption
- Digest Algorithms
- Supported SSHFTP Ciphers
- Cipher Suites
- SSHFTP Support in TIBCO BusinessConnect Container Edition

Public and Private Keys

TIBCO BusinessConnect Container Edition uses both PKI and PGP encryption methods to validate private and public keys. Both in PKI and in PGP method, each key pair has a public and a private part and messages are encrypted with the public part of the key and can only be decrypted with the associated private part of the key. This is done to ensure that only the intended recipient of the message can actually read it.

For creating and verifying signatures on messages, the holder of the private part of the key pair uses the private key to sign the message. Entities, which have the public part of the key pair are then able to verify that the signature on the message was created by the holder of the private part of the key pair and therefore be assured that the message was sent by the holder of the private part of the key pair.

The following keys are supported in TIBCO BusinessConnect Container Edition:

Type of Key	Description
Public Keys	These keys are provided to the trading partners so that they can encrypt data and verify signatures. • For PKI: PKCS#7 public key identity format, which comes in the following file formats: .p7b and .p7c. .Storing of individual X509 certificates in PEM (base64) and DER (ASN.1 Distinguished Encoding) formats is also supported. • For PGP: Supported types are DSA/ElGamal and RSA public key
Private Keys	Used to decrypt data and sign messages. The extension of the private key file name is most commonly referenced as .p12, but it may be anything else as long as the data in the file is compliant with the PKCS#12 specification. Supported types for PGP are DSA/ElGamal and RSA private keys.
SSH Private and Public Keys	Used to support the SSHFTP transport in TIBCO BusinessConnect Container Edition.

Digital Certificates

Digital certificates are data strings that a Certificate Authority (CA) creates after the CA verifies the identity of an entity that has submitted a CSR (Certificate Signing Request).

When the CA signs and issues a certificate to a user, the CA's signature on the certificate verifies the authenticity of the link between the user's public key and the user's actual identity. A user can then use its certificate, as contained in its certificate file, to identify itself during e-commerce. The three basic items in a certificate are the CA's signature, the user's identity, and the user's public key. A certificate is like a driver's license in that both are issued by a recognized authority (a CA or a governmental agency, respectively) and both identify the holder. Certificates are specified by the X.509 standard, such as X.509v3.

Digital certificates perform these functions:

• Certify the identity of the holder of the certificate

- Allow for non-repudiation of transactions

- Encrypt email messages

- Sign a mobile code that can be downloaded by a web server

These certificates contain both a private key for the certificate holder and a public key for distribution to partners. They expire on a pre-determined date. Digital certificates are based on the trust that both trading partners hold in the certificate authority. Some CAs are themselves authenticated using a certificate by a higher-level CA, which may in turn be authenticated by a certificate from an even higher-level CA. This results in a certificate chain.

In a transaction, when digital certificates are being exchanged between the two parties, Party A uses the other party's public key to encrypt the transaction data whereas Party B uses its own private key to decrypt the data.

Using Certificates with TIBCO BusinessConnect Container Edition

A large number of certificate authorities (CAs) is in the business of providing digital certificates (also called SSL certificates) to authenticate the identity of the certificate holder.

Obtaining a Certificate

You can obtain an SSL certificate from the following websites of any authorized certificate authority (CA):

- VeriSign: https://www.verisign.com/

- GeoTrust: https://www.geotrust.com/

> ℹ **Note:** TIBCO BusinessConnect Container Edition supports X509 certificates versions 1, 2, and 3. All digital certificates used in TIBCO BusinessConnect Container Edition must be compliant with the PKIX standard RFC #3280, which is described on the https://www.ietf.org/ website.

The following are the three kinds of certificates that you use while working with TIBCO BusinessConnect Container Edition and the PKI validation method:

Certificates Authority (CA)

This is a trusted third party that validates identities and issues X.509 certificates by signing the certificate with its signature. Any client or server software that supports certificates has a collection of trusted CA certificates, which determine the certificate issuers that the software can trust. The root CA's certificate is unique as it is a self-signed certificate. It is signed by the root CA itself. The CAs that are directly subordinate to the root CA in the CA hierarchy have CA certificates that were verified and signed by the root CA.

Certificate Chain

Root certificates

The certificate issued by the highest level certificate authority (CA) is called the root certificate.

> **ⓘ Note:** You can add CA certificates directly to the certificate store outside of the partner configuration process.

Leaf certificates

These certificates are issued to you directly from a CA. They are also called identity certificates. You acquire a leaf certificate from a CA by sending a Certificate Signing Request (CSR), which is associated with the private key of your server.

You can acquire a leaf certificate from a CA by sending a Certificate Signing Request (CSR), which is associated with the private key of your server.

Intermediate certificates

The certificates in the chain that lead up to the highest-level CA are called subordinate or intermediate certificates.

> **ⓘ Note:** TIBCO BusinessConnect supports X509 certificates versions 1, 2, and 3. All digital certificates used in TIBCO BusinessConnect must be compliant with the PKIX standardRFC #3280, which is described on: https://www.ietf.org/ website.

Certificates File

A file that contains the private key's certificate chain. Unlike a key identity, it contains no private key and is not protected with a password. Trading partners exchange certificate files during the setup phase of their relationship. Each trading partner then installs the other partner's certificate file. For a host to verify the validity of a trading partner's certificate, the host must trust each CA's certificate in the certificate chain within the trading partner's certificate file. The certificate file defines how each trading partner should expect the other to identify itself in e-commerce transactions. The supported format is PKCS#7 certificates only, which can have file extensions like .p7b and .p7c. When setting up an installation for e-commerce, the key identity file relates to the trading host and one or more certificate files related to one or more trading partners that the host has.

Storing Certificates

A certificate exists in a system file. To exchange business documents with a trading partner you must store the certificates as part of that participant. Hosts require a private key in addition to a public key certificate; partners only include public key certificates. TIBCO BusinessConnect Container Edition stores all certificates, including root, leaf, and intermediate certificates, in a central location: the credential store.

Shadow Credentials

Shadow credentials stand ready to take over for credentials that expire. You define when the shadow credential takes effect.

You can assign a shadow credential to any private key or certificate if all the following standards are met:

- The valid time period for the shadow and base credentials overlap

- Shadow and base credentials are both valid at the time you assign the shadow

- Both credentials are still valid at the time when the shadow credential is to take effect

> ℹ **Note:** You cannot assign a shadow credential to another shadow credential. After the shadow credential takes effect, it is still a shadow credential. You have to remove or update the original credential and remove or promote the shadow credential. A shadow credential is used during overlay and shadow credential period for HTTPS and HTTPSCA transport level handshake of SSL/TLS and for client authentication.

TIBCO BusinessConnect Container Edition supports shadow credentials to be on standby whenever the primary configured credential is about to expire. The activation of shadow credential can be set at the participant level, and it takes effect on the date that is specified.

The following terms and definitions are used to describe when shadow credential gets picked for different usages:

Original credential period

  The period between the date when the original credential was uploaded to the date before the activation date was set for the shadow credential.

Overlay period

  This definition is applicable only when the shadow credential is associated with the original credential. It is defined as the period between the activation date of the shadow credential and the end of the original credential's expiration time.

Shadow credential period

  This period starts when the original certificate expires and lasts until the shadow credential expires.

To understand which credentials get picked for different operations, see the following table:

Usage Description	Message Flow Direction	Type of Credential Used During Different Periods		
		Original Credential Period	Overlay Period	Shadow Credential Period
Message signing, encryption	Outbound to Partner	Original credential used	Shadow credential used only	Shadow credential only
Message authentication and decryption	Inbound message from partner	Original credential used	Shadow credential used first, if it fails the original credential is tried	Shadow credential only

This behavior is valid for protocols that support plain Email/AS1/AS2 SMIME messaging. Check the appropriate protocol documentation for behavior of SMIME message processing other than plain Email/AS1/AS2.

Digital Signatures

Authentication using digital signatures is done using S/MIME authentication. It involves adding a digital signature to the outgoing message. Digital signatures are verifiable transformations made on a piece of data by the private key, which can be verified by using the corresponding public key. They bind a document to the possessor of a particular key. Digital signatures are used to bind information to the identity of its originator. They can be used to provide data origin authentication and data integrity.

A digital signature includes the following parts:

- A certificate authority's distinguished the name of the signer

- A sender's public key (optional)

- The serial number of the signer's certificate

Encryption

Encryption is available through the following security mechanisms:

- SSL This protocol uses public and private keys to enable encryption of the transport protocol on which an encrypted or unencrypted message travels.

- S/MIME This message packaging and signing protocol uses public and private keys to enable encryption and decryption of a message.

- SSH SSH keys are used to support the SSHFTP transport in TIBCO BusinessConnect Container Edition.

This protocol provides transport layer security with both server and client authentication by establishing a secured channel through key negotiation and strong encryption algorithms.

For more information about SSH, see SSHFTP Support in TIBCO BusinessConnect Container Edition.

Digest Algorithms

Digest algorithms utilized in digital signatures provide help in detecting changes in the signed payload since the signature has been generated on the content. The procedure of verifying that no unauthorized changes were made on the signed content is called the verification of the digital signature. If verification is successful, parties can be certain that the document has been created by the signing party and that it was unaltered since its signing.

TIBCO BusinessConnect Container Edition offers the following digest algorithms to verify digital signatures:

- SHA1

- SHA-256

- SHA-384

- SHA-512

Encryption Algorithms

Encryption algorithms are used in two different contexts:

- Transport layer

- Business layer

In the transport layer, the encryption takes place on the transport connection, which considers the data moved through it opaque. The negotiation of the symmetric keys takes place by an asymmetric algorithm utilized and defined in SSL/TLS or SSH. In the business layer, business documents' payloads are encrypted as per the specification of the given business protocol. These options can be used independently from each other. There are multiple encryption algorithms available for use. You and your business partner must use the same encryption algorithm; otherwise, decryption is not possible. The number included in the name of the algorithm is the number of bits. This is independent of the bit size of the certificate. The larger the algorithm bit size, the more secure the encryption.

TIBCO BusinessConnect Container Edition supports the following encryption algorithms for the encryption of transport layer and business layer:

- DES3

- AES-128

- AES-192

- AES-256

Supported SSHFTP Ciphers

The following ciphers are supported for SSHFTP protocol:

Key Exchange Algorithms

```
DIFFIE-HELLMAN-GROUP1-SHA1
DIFFIE-HELLMAN-GROUP14-SHA1

DIFFIE-HELLMAN-GROUP-EXCHANGE-SHA1
DIFFIE-HELLMAN-GROUP-EXCHANGE-SHA256
ECDH-SHA2-NISTP256
ECDH-SHA2-NISTP384
ECDH-SHA2-NISTP521
CURVE25519-SHA256@LIBSSH.ORG
CURVE25519-SHA256
RSA2048-SHA256
RSA1024-SHA1
DIFFIE-HELLMAN-GROUP14-SHA256
DIFFIE-HELLMAN-GROUP15-SHA512
DIFFIE-HELLMAN-GROUP16-SHA512
DIFFIE-HELLMAN-GROUP17-SHA512
DIFFIE-HELLMAN-GROUP18-SHA512
```

Encryption

```
AES128_CBC
AES192_CBC
AES256_CBC
AES128_CTR
AES192_CTR
AES256_CTR
ARCFOUR
ARCFOUR_128
ARCFOUR_256
AES_GCM_128
AES_GCM_256
BLOWFISH_CBC
CHACHA20_POLY1305
TRIPLEDES_CBC
TRIPLEDES_CTR
```

Server host keys

```
RSA-SHA2-512
RSA-SHA2-256
SSH-RSA
SSH-DSA
```

Macs

```
HMAC_SHA1
HMAC_SHA1_ETM
HMAC_SHA1_96
HMAC_MD5
HMAC_MD5_ETM
HMAC_MD5_96
HMAC_SHA256
HMAC_SHA256_ETM
HMAC_SHA256_96
HMAC_SHA512
HMAC_SHA512_ETM
HMAC_SHA512_96
HMAC_RIPEMD160
HMAC_RIPEMD160_ETM
```

Compression

```
NONE
ZLIB
ZLIB@OPENSSH.COM
```

Cipher Suites

The following cipher suites are supported for BusinessConnect Container Edition:

> **Note:** The ciphers for all SSL/TLS-based secure protocols are implemented by the underlying Java platform. For the latest information about supported ciphers, see the official Java documentation.

Displayed in the Gateway Server Logs

```
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
TLS_RSA_WITH_AES_256_CBC_SHA256
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384
```

```
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
TLS_DHE_DSS_WITH_AES_256_CBC_SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA
TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA
TLS_ECDH_RSA_WITH_AES_256_CBC_SHA
TLS_DHE_RSA_WITH_AES_256_CBC_SHA
TLS_DHE_DSS_WITH_AES_256_CBC_SHA
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
TLS_RSA_WITH_AES_128_CBC_SHA256
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
TLS_DHE_DSS_WITH_AES_128_CBC_SHA256
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_128_CBC_SHA
TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA
TLS_ECDH_RSA_WITH_AES_128_CBC_SHA
TLS_DHE_RSA_WITH_AES_128_CBC_SHA
TLS_DHE_DSS_WITH_AES_128_CBC_SHA
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
TLS_RSA_WITH_AES_256_GCM_SHA384
TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384
TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
TLS_DHE_DSS_WITH_AES_256_GCM_SHA384
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
TLS_RSA_WITH_AES_128_GCM_SHA256
TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256
TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
TLS_DHE_DSS_WITH_AES_128_GCM_SHA256
TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA
TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA
TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA
TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA
```

Export and Stronger

```
TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
TLS_RSA_WITH_RC4_128_SHA (0x005)
TLS_RSA_WITH_RC4_128_MD5 (0x0004)
TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x00a)
TLS_RSA_WITH_DES_CBC_SHA (0x0009
TLS_RSA_EXPORT_WITH_RC4_40_MD5 (0x0003)
TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5 (0x0006)
TLS_RSA_EXPORT_WITH_DES40_CBC_SHA (0x0008)
TLS_DHE_DSS_WITH_AES_128_CBC_SHA (0x0032)
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA (0x0013)
TLS_DHE_DSS_WITH DES_CBC_SHA (0x0012)
TLS_DHE_DSS_EXPORT_WITH_DES40_CBC-SHA (0x0011)
TLS_DHE_RSA_WITH_AES_128_CBC_SHA (0x0033)
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (0x0016)
TLS_DHE_RSA_WITH_DES_CBC_SHA (0x0015)
TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA (0x0014)
TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
TLS_DHE_DSS_WITH_AES_256_CBC_SHA (0x0038)
TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)
```

Stronger than Export

```
TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
TLS_DHE_DSS_WITH_AES_256_CBC_SHA (0x0038)
TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)
TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x00a)
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA (0x0013)
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (0x0016)
TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
TLS_RSA_WITH_RC4_128_SHA (0x005)
TLS_DHE_DSS_WITH_AES_128_CBC_SHA (0x0032)
TLS_DHE_RSA_WITH_AES_128_CBC_SHA (0x0033)
TLS_RSA_WITH_RC4_128_MD5 (0x0004)
TLS_RSA_WITH_DES_CBC_SHA (0x0009
TLS_DHE_DSS_WITH DES_CBC_SHA (0x0012)
TLS_DHE_RSA_WITH_DES_CBC_SHA (0x0015)
```

128 Bit and Stronger

```
TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
TLS_DHE_DSS_WITH_AES_256_CBC_SHA (0x0038)
TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)
TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x00a)
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA (0x0013)
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (0x0016)
TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
TLS_RSA_WITH_RC4_128_SHA (0x005)
TLS_DHE_DSS_WITH_AES_128_CBC_SHA (0x0032)
TLS_DHE_RSA_WITH_AES_128_CBC_SHA (0x0033)
TLS_RSA_WITH_RC4_128_MD5 (0x0004)
```

Stronger than 128 Bit

```
TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
TLS_DHE_DSS_WITH_AES_256_CBC_SHA (0x0038)
TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)
TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x00a)
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA (0x0013)
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (0x0016)
```

256 Bit and Stronger

```
TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
TLS_DHE_DSS_WITH_AES_256_CBC_SHA (0x0038)
TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)
```

SSHFTP Support in TIBCO BusinessConnect Container Edition

To support the SSHFTP transport in TIBCO BusinessConnect Container Edition, the following types of keys, certificates, and algorithms are available:

- Key types: RSA, DSA

- Public key formats: OpenSSH PEM and Ssh.Com* (with import and export)

- Private key formats: OpenSSH PEM and Ssh.Com** (with import); OpenSSH PEM (with export)

- Host signature algorithms: SSH-RSA and SSH-DSS

- Server public key algorithms: DSA and RSA.

Authentication Methods for SSHFTP

The supported authentication methods are password, public key, and a combination of password and public key. The client is always identified by a username, whether the authentication takes place by password, public key or both. The SSH server drives the authentication (requests the preferred authentication methods) and the SSH client obeys by submitting the credentials, which are specific to the requested/agreed-upon method or methods.

Password

The configured password is used to complete the user authentication phase with the SSH server.

Public key

The configured public key (retrieved from the user's SSH private key) is used to complete the user authentication phase with the SSH server.

Public key and Password

TIBCO BusinessConnect Container Edition is allowed to authenticate using password, public key, or both password and public key. If the SSH server indicates both options, TIBCO BusinessConnect Container Edition starts using the 'public key' method. If it is successful and the server requires no further authentication steps to be run, the negotiation is successful and the tunnel is established.

If the server rejects the authentication attempt, TIBCO BusinessConnect Container Edition moves to password mode, in which case the outcome depends on the success of this attempt. If the password fails, the transport creation fails and the framework sends the corresponding error message to the business protocol.

When either 'Public Key' or 'Public Key and Password' is selected, the sending participant must be configured with an SSH private key since the transport assumes that this credential is made available to (and may be requested by) the SSH server. The client's private key for any inbound or outbound SSHFTP transport is configured through the field

'Client Authentication Identity for SSHFTP' on the corresponding business agreement of the sending and receiving participants.

SSH Server Public Key Retriever

As an administrator, you may face problems finding, installing, and configuring the public keys of SSH servers of the trading partners while setting up and configuring inbound and outbound SSH transports in TIBCO BusinessConnect Container Edition. Sometimes, it is a priority to be able to set-up a working connection quickly, instead of taking enough time to ensure that the identities of the peer trading partners' SSH servers be trusted by retrieving the servers' credentials only from verified/trusted sources. The SSH Server Public Key Retriever was added to facilitate speedy setup of a working connection, and to help establish a trusted connection.

Selecting Algorithms and Methods during Tunnel Negotiation

Supported MACs for SSHFTP

```
HMAC-SHA2-256*
HMAC-SHA2-512*
HMAC-SHA256@SSH.COM*
HMAC-SHA512@SSH.COM*
HMAC-SHA1
HMAC-SHA1-96*
HMAC-MD5-96
```

* Macs that cannot be selected from the TIBCO BusinessConnect Container Edition GUI.

If configured to ANY, then any of the supported MACs can be selected by the server.

Supported Compression Algorithms for SSHFTP

```
Zlib
Zlib@openssh.com
```

If NONE is selected, no compression is enforced by the client. This assumes that the SSH server also considers 'NONE' to be a valid option.

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 following documentation for TIBCO BusinessConnect™ Container Edition is available on the TIBCO BusinessConnect™ Container Edition page.

- *TIBCO BusinessConnect™ Container Edition Release Notes*

- *TIBCO BusinessConnect™ Container Edition Installation and Deployment*

- *TIBCO BusinessConnect™ Container Edition Concepts*

- *TIBCO BusinessConnect™ Container Edition Trading Partner Management*

- *TIBCO BusinessConnect™ Container Edition Administration*

- *TIBCO BusinessConnect™ Container Edition Security Guidelines*

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.

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

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.tibco.com/patents.