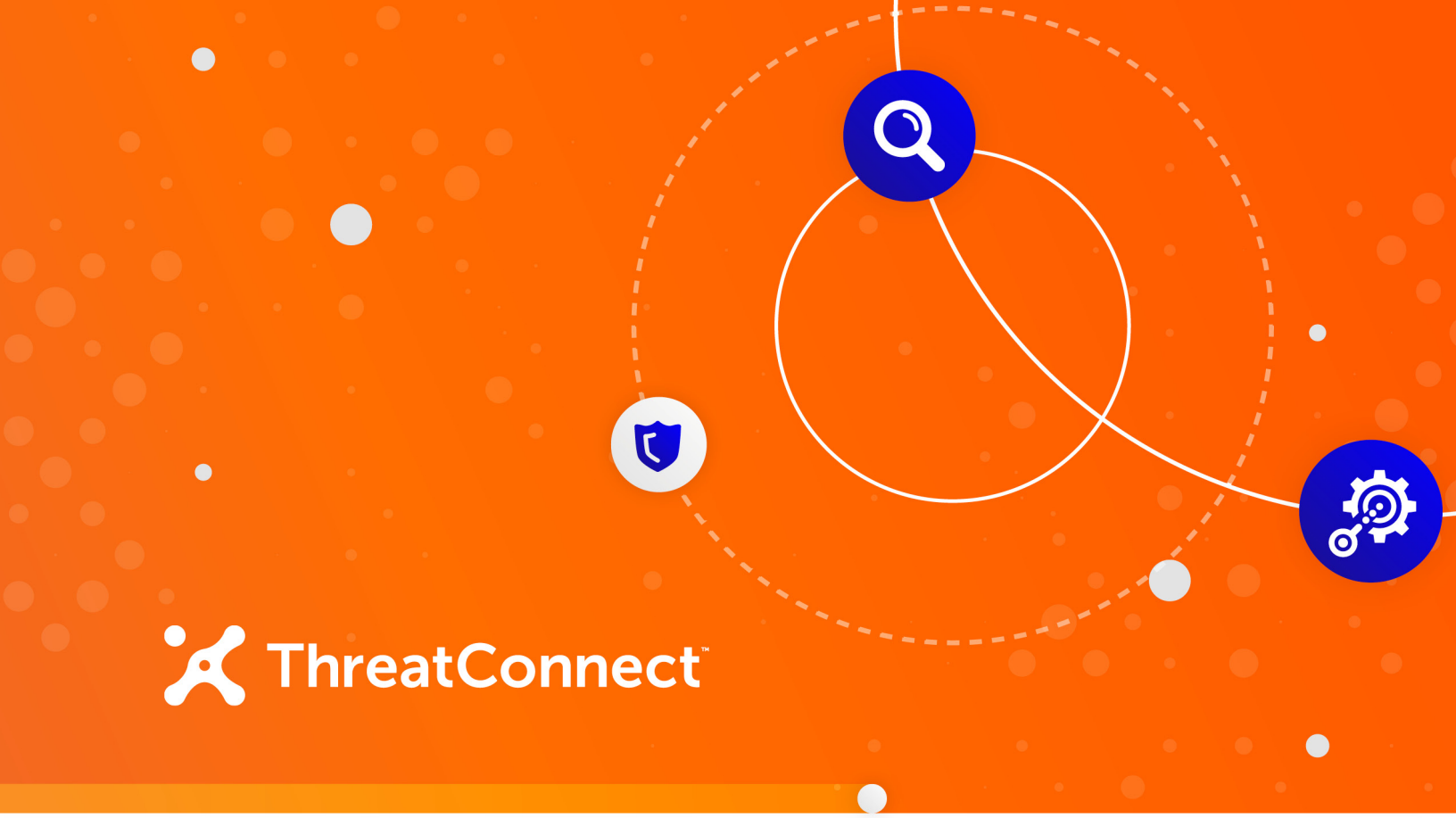

ThreatConnect™

SAML

Installation and Configuration Guide

DOCUMENT VERSION 2.0

MARCH 19, 2020

10017-03 EN Rev. A

ThreatConnect, Inc.
3865 Wilson Blvd., Suite 550, Arlington, VA 22203

P: 1.800.965.2708 | F: 1.703.229.4489
www.ThreatConnect.com

Table of Contents

INTRODUCTION

Security Assertion Markup Language (SAML) is used in a Single Sign-On (SSO) environment. It is a widely accepted standard for transporting authentication and authorization data between multiple systems. The central system is the Identity Provider (IDP), which contains the identity profile of users within an organization. The profile data can consist of job title, organizational role, email address, etc. This information is shared when requested by other systems within the enterprise. These requesters, or clients, are applications called Service Providers (SP).

When a user attempts to log into an SP by providing his or her credentials, the SP issues a request to the IDP for the user's profile data. If the IDP confirms the user's credentials and discovers the associated profile, then the IDP will fulfill the request by responding with a SAML assertion containing the user's organizational information. Once the SAML assertion is received, the SP will interpret the assertion and grant the user access to the application with the affiliated role.

Multiple vendors implement and sell the IDP system. These vendors include CA Single Sign-On (aka SiteMinder), PingFederate®, and others. These products leverage the SAML standard to exchange authentication and authorization data with an SP.

ThreatConnect® can be configured as an SP, and this guide instructs users how to install the platform with SSO configurations, in order to allow ThreatConnect to participate in an SSO enterprise as an SP requesting identity data from the designated IDP.

The ThreatConnect SSO feature has been thoroughly tested with CA Single Sign-On, and it has also been proven to work with other vendors who utilize the SAML standard. Thus, any IDP vendor will work with ThreatConnect if that vendor issues SAML assertions and strictly follows the SAML standard.

The SSO feature currently supports signed assertions, but it does not support encrypted assertions. For signature support, carefully follow the certificate installation instructions.

NOTE: User authentication is supported by any integration that supports SAML 2.0.

INSTALLATION PREREQUISITES

SSL Certificates

Install and configure the Secure Sockets Layer (SSL) certificates on the ThreatConnect instance **before** implementing Security Assertion Markup Language (SAML). Refer to the *ThreatConnect Installation Guides* for detailed operating system instructions to install SSL certificates.

NOTE: When importing a customer's Identity Provider (IDP) public certificate into ThreatConnect's local keystore, use the IDP hostname as the alias for the certificate.

Required Information

From the Deployment Engineer involved in the SAML installation and configuration, obtain a *SAML-IDP Specification Chart*. Fill out this spreadsheet **before** the scheduled SAML installation, and return it to the engineer, so that he or she may review and ensure that all needed data parts have been filled out.

INSTALLATION

Enabling the SAML Option

Unless all prerequisites have been accomplished before the initial configuration of ThreatConnect, some options will need to be adjusted in order to access the SAML configuration. These changes will reset all ThreatConnect installation when running the SAML configuration.

To enable the SAML configuration, stop the ThreatConnect service and edit /<install location for ThreatConnect>/threatconnect/config/install.properties file:

```
# service threatconnect stop
# vim /opt/threatconnect/config/install.properties
```

Change setupcompleted=true to setupcompleted=false. Save the file and quit. Log in as the threatconnect user and run setup.sh:

```
# su - threatconnect
# cd /<install location for ThreatConnect>/threatconnect/app
# ./setup.sh
```

Run through the initial setup configuration for ThreatConnect again. Enter **true** when asked about SAML configuration. Input the items pulled from the completed SAML spreadsheet into the appropriate fields listed below:

i. Enter IDP URL <value of IDP URL location>.
ii. Enter Service Provider (SP) URL <value of SP URL location>.
iii. Enter IDP SLO URL <value of IDP Single Logout URL>.
iv. Set IDP post binding (true or false).
v. Enter SP binding type (POST | REDIRECT) <value of SP binding method>.
vi. Enter SP relay address <value of SP relay URL (must be encoded)>.
vii. Enter SAML2's skew milliseconds.
viii. Please enter name ID format: <value should be **urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified**, unless directed by the authentication team to input something else>
ix. Verify SAML's signatures (true or false) <enable signature verification>.
x. Enter IDP host for public certificate <value of IDP host>.
xi. Enter SP signing key alias <value of the signing key alias for SP>.
xii. Enter keystore absolute path <value of keystore location>.
xiii. Enter keystore password.
xiv. Enter keystore password again.
xv. Enter private key password.
xvi. Enter private key password again.
xvii. Enable SHA256 SAML signature algorithm [true\false]: (false will use SHA1).
xviii. Proceed with SAML changes in server configuration (yes or no).

NOTE: When importing a customer's IDP public certificate into ThreatConnect's local keystore (xi), use the value entered for "Enter IDP host for public certificate" (ix) as the alias for the certificate.

Log out as the threatconnect user, add the admin account, and restart the threatconnect service to access the Web UI:

```
# logout
# service threatconnect start
```

For SAML to work on user accounts, the admin account will need to be set up before the user creation can occur. Connect to the MySQL database that is being used by the ThreatConnect instance and run the strings below. Note that the value for [UserName] appears multiple times and needs to be entered in all places.

```
# mysql -u tcuser -p threatconnect

INSERT INTO UserTable (id_Organization, id_TimeZone, id_Role, userName,
password, salt, firstName, locked, resetRequired, failedAttempts, lastLogin,
summaryEmailHour, deleted, disabled, pseudonym, profileConfigured,
profileEditable, receiveReplyNotification, tracked, termsAccepted,
logoutInterval, mailBounceCount) VALUES ('1', '24', '2', '[UserName]', '',
'', '[FirstName]', '0', '0', '0', NULL, '5', '0', '0', '[UserPseudonym]',
'1', '1', '0', '0', '0', '1800', '0');

INSERT INTO CommunityMembership (id_Organization, id_User, id_OwnerRole)
SELECT u.id_Organization, u.id, o.id FROM UserTable u, OwnerRole o WHERE
u.userName IN ('[UserName]') AND o.name = 'Organization Administrator' AND
NOT EXISTS (SELECT id FROM CommunityMembership c WHERE u.id_Organization  =
c.id_Organization AND u.id = c.id_User);

INSERT INTO UserNotificationSettings (id_User, severity, id_NotificationType,
actionImmediate, actionSummary, actionRealTime) SELECT u.id, 'HIGH', NULL, 1,
0, 1 FROM User u where u.username = '[UserName]';

INSERT INTO UserNotificationSettings (id_User, severity, id_notificationtype,
actionImmediate, actionSummary, actionRealTime) SELECT u.id, 'MEDIUM', NULL,
1, 0, 0 FROM User u where u.username = '[UserName]';

INSERT INTO UserNotificationSettings (id_User, severity, id_notificationtype,
actionImmediate, actionSummary, actionRealTime) SELECT u.id, 'LOW', NULL, 0,
1, 0 FROM User u where u.username = '[UserName]';
```

Change these values based on the setup of the client's admin account:

```
[UserName] = admin@<client's email domain> (email address of admin account in
IDP LDAP)
[FirstName] - first name of user
[UserPseudonym] - pseudonym value for user
```

Once the admin is able to log in via SAML, the additional user account can be created within the ThreatConnect Web UI.

ADDITIONAL NOTES

Metadata Requested by the IDP Administrator

The client's Single Sign-on (SSO) team may request the ThreatConnect instance metadata in order to set up their SAML configuration. In order to pull this data, the SSL certificates must be installed and operational **before** the full XML file can be given to the client's SSO team.

Template Metadata file:

```
<?xml version="1.0" encoding="UTF-8"?>
<md:EntityDescriptor xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
entityID="https://<FQDN-of-ThreatConnect-server>/auth/">
      <md:SPSSODescriptor AuthnRequestsSigned="true"
WantAssertionsSigned="true"
protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
            <md:KeyDescriptor use="signing">
                  <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
                        <ds:X509Data>
<ds:X509Certificate>MIIC0TCCAbkCAQAwXDELMAkGA1UEBhMCSU4xCzAJBCgKCAQEAmEwfAFLj
gDOBgNVBAoTCXJicHJpdmF0ZTELMAkGA1UECxMCUUExETAPBgNVBAMTCHJhamt1bWFyMIIBIjANBg
kqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAmEwfAFLjgDOEZk1AYPhX7dbYMXqkk4rF3uyYZeoMnnX
PLs463GzGvVPnRgjTdIzm+1QOnkTx3BBu7kxlhtze2Sr7rtHLs1FYbzXREs5aVgIPnpkfuKdR9QND
aJJ1byxStnF+zI4feSYmHXsVWfHm24+FK0kCk3tSnw2/noXyW5xc2UbrGLYqaezpPSlf5WJ3isKFl
Qr2k+HKXh4Rid4TUmEaoZXPAcB7QtkBYnIxzzmBoFCWSSsVldPRkaw=</ds:X509Certificate>
                        </ds:X509Data>
                  </ds:KeyInfo>
            </md:KeyDescriptor>
      <md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified</md:NameIDFormat>
            <md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
Location="https://<FQDN-of-ThreatConnect-server>/auth/" index="0"
isDefault="true" />
      </md:SPSSODescriptor>
</md:EntityDescriptor>
```

Adjust the following sections based on the setup:

SAML:2.0:bindings: - Needs to be HTTP-POST or HTTP-REDIRECT, based on the spreadsheet documentation for the client

ds:X509Certificate: X509 data will be pulled from the ThreatConnect server certificate that is installed in the keystore.jks file on the ThreatConnect instance

Location=: "https://<FQDN-of-ThreatConnect-server>/auth/"

entityID: Is usually same as location, but can be defined by the IDP admin