Thredd Identity and Access Management Migration Guide

Version: 1.0

05 October 2026

Publication number: TIAMMG-1.0-10/5/2026

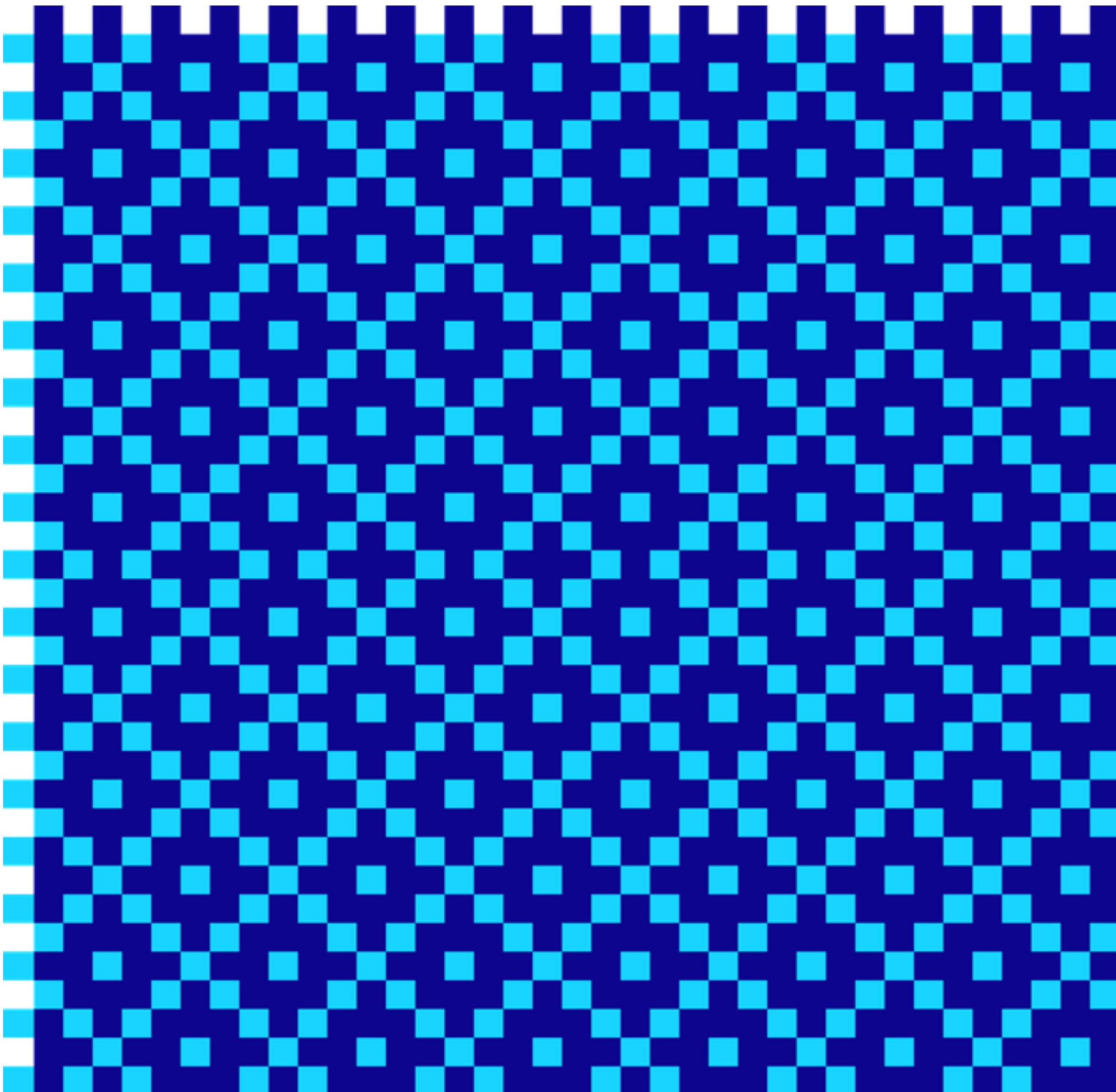
For the latest technical documentation, see the Documentation Portal.

Thredd, 33 Kingsway, London, WC2B 6UF, UK

Support Email: occ@thredd.com

Support Phone: +44 (0) 203 740 9682

© Thredd 2026



Copyright

© Thredd 2026

The material contained in this guide is copyrighted and owned by Thredd Ltd together with any other intellectual property in such material.

Except for personal and non-commercial use, no part of this guide may be copied, republished, performed in public, broadcast, uploaded, transmitted, distributed, modified or dealt with in any manner at all, without the prior written permission of Thredd Ltd., and, then, only in such a way that the source and intellectual property rights are acknowledged.

To the maximum extent permitted by law, Thredd Ltd shall not be liable to any person or organisation, in any manner whatsoever from the use, construction or interpretation of, or the reliance upon, all or any of the information or materials contained in this guide.

The information in these materials is subject to change without notice and Thredd Ltd. assumes no responsibility for any errors.

About this guide

This guide provides information about how to migrate your organisation, client applications, users, connectivity and identity and access management to Thredd Portal, REST APIs via API Hub, and Thredd's Secure Framework.

Thredd's scalable security architecture roadmap is aligned with emerging standards and regulatory requirements. In the context of this work, Thredd has enhanced its identity and management provision, connectivity services, Certificate Authority (CA), and developer tooling within its Secure Framework.

As a result, Thredd is phasing out support for Raidiam Connect for issuing certificates and Cloudentity as an identity provider and OAuth OpenID Provider.

If you are currently using Cloudentity and Raidiam Connect, you can migrate to Thredd Portal and the new identity and access management platform once you receive confirmation from Thredd that you can begin this migration process.

Target audience

This guide is aimed at developers and system integrators who need to migrate and configure secure connections to Thredd services as an existing client. It provides information for security consultants and Chief Information Security Officers (CISOs) who need to assess Thredd's security infrastructure.

What's changed?

If you want to find out what's changed since the previous release, see the Document History section.

How to use this guide

This guide covers several different migration pathways, which differ primarily based on the secure connectivity method available to clients to connect to Thredd. The migration process that you must follow depends on your existing configuration with Thredd and your target configuration state.

When you are ready, refer to the main Migration Overview Index page to identify and follow the Migration Overview that best matches your existing and target configurations.

Each overview summarises the different stages of the typical migration pathway. Use the Migration Overview as your guide to work through the different stages, and use the links within it to the instructions that you must complete for each stage.

To ensure that you follow the correct process, refer to the Migration Overview and sections that match your target state for connecting to Thredd. You should also familiarise yourself with the Thredd Secure Framework and Thredd Portal overview.

> **Tip:** For the latest technical documentation, see the Documentation Portal.

Section 1: Getting Started

This guide covers several different migration pathways, which differ primarily based on the secure connectivity method available to clients to connect to Thredd.

Read this section to identify the migration process that you must follow – this depends on your existing configuration with Thredd and your target configuration state.

You can also familiarise yourself with the Thredd Secure Framework before you begin the process of migrating and managing access for your applications and users via the Admin features of Thredd Portal.

Topics covered in this section:

- About this guide
- Thredd Secure Framework

Tip: To find out what has changed, see the Document History.

Thredd Secure Framework

Thredd's Secure Framework is the combination of several components which enables clients to securely access Thredd's resources over mTLS (Mutual Transport Level Security). *Thredd Portal* sits at the centre of this, enabling Admins to manage their organisation's applications, users, and credentials to access Thredd services through a single user interface and dashboard.

Thredd supports industry standards for authentication, identity and access management, and secure network and API-level connectivity.

Key features

Identity and Access Management through Thredd Portal

- Standards-based authentication – support for OpenID Connect (OIDC) and SAML.
- Single-Sign On (SSO), self-service onboarding, and token lifecycle management.
- Secure Access Control – role-based access (RBAC) and user management.

Connectivity Services

- Secure network and API-level connectivity via VPN, TLS and mTLS.
- Certificate lifecycle management – request, revoke and renew certificates issued by Thredd's Certificate Authority through Thredd Portal.
- mTLS Termination for mTLS connections in an EHI mTLS setup.

Developer Tooling

- Securely access Thredd REST APIs and API Hub.
- Test connectivity to Thredd REST APIs and API Hub using the Postman Collection in a UAT environment.

Identity and Access Management through Thredd Portal

Thredd provides a Software as a Service (SaaS) capability that acts as the *Identity Provider (IDP)* to authenticate user access to Thredd's interfaces through logging in to Thredd Portal. It also acts as an *OAuth OpenID Provider (OP)* for the registration and management of customer applications, generation and validation of access tokens, and for the enforcement of access control policies.

Thredd supports using a Client Secret (which requires an access token from Thredd) for Client Authentication when connecting to the REST APIs.

Thredd Certificate Authority through Thredd Portal

Thredd adopts a self-service approach that enables Admin users to independently request and manage certificates for their applications from Thredd's Certificate Authority (CA) through Thredd Portal.

Client applications require a transport certificate to connect to services, depending on the configuration you choose. These are required for mTLS connections for establishing connections between resources.

Thredd Application	Certificates Required
REST API over mTLS	Transport Certificate for the mTLS connection, which you obtain from Thredd Portal.
SOAP API	Transport Certificate, which you obtain from Thredd.
External Host Interface (EHI) over mTLS	Server Certificate (which you obtain separately), Thredd Trust Chain (that you download from Thredd), and a Transport Certificate (which Thredd presents).
Thredd Portal	Certificates that are pre-installed by Thredd.

Mutual TLS explained

Mutual Transport Layer Security (Mutual TLS, mTLS) is an enhanced security protocol that requires a Client and Server to verify each other's identity before establishing a secure communication channel. This digital handshake requires both parties to provide their official identity.

What is Transport Layer Security (TLS) and how is mTLS different?

Transport Layer Security (TLS) is the standard internet protocol that ensures private, secure communication. For example, it enables HTTPS, where HTTP essentially runs over TLS, to secure the websites that people visit daily. TLS uses one-way authentication, where only the Server presents a certificate to the Client to prove its identity.

With *Mutual TLS* (mTLS), this process is reciprocal, as the 'Mutual' part of its name implies. mTLS requires two-way authentication, which means that both the Client and the Server must present a valid digital certificate that allows them to verify each other's identities.

Feature	TLS (one-way)	Mutual TLS (two-way)
Server Authentication	Required (the Server proves its identity to the Client)	Required
Client Authentication	Optional (usually none)	Required (the Client proves its identity to the Server)
Verification method	The Server uses a Server Certificate	The Client and the Server use Certificates

How mTLS works using a two-way verification process

The mTLS process involves four key steps:

1. Client request: A Client (your application) attempts to connect to the Server.
2. Server challenge and authentication: The Server sends its digital certificate to the Client and simultaneously requests the Client's Transport Certificate.
3. Client certificate presentation: The Client sends its unique Transport Certificate and a digital signature back to the Server.
4. Mutual verification:
    1. The Client verifies the Server's certificate.
    2. The Server verifies the Client's Transport Certificate, checking its validity, issuing authority, and integrity.
    3. If both checks pass, the Server and Client establish a TLS connection. If a check fails, the connection is immediately terminated.

Key benefits

By implementing mTLS, Thredd provides a significantly higher level of assurance and security for clients, and their data and services:

- Zero Trust Architecture (ZTA): mTLS is foundational to a Zero Trust security model. It ensures that no device or application is trusted by default, regardless of its location (inside or outside the network).
- Stronger authentication: It eliminates reliance on API keys alone. Your Transport Certificate is a unique, cryptographically-secured identity that is much harder to forge or steal.
- Protection against Man-In-The-Middle (MITM) attacks: The mutual verification process makes it virtually impossible for an unauthorised third party to impersonate either the client or the server to intercept data.

Transport Certificates to connect to REST APIs via mTLS

For Thredd's systems to securely authenticate your application, you must provide a valid Transport Certificate when using mTLS. This certificate is issued by Thredd's internal Certificate Authority and acts as the digital passport for your client application.

The Transport Certificate is mandatory for mTLS and is used solely for establishing the secure connection. Without a valid, signed Transport Certificate, your application is unable to connect to Thredd's secure endpoints. An Admin user can request a Transport Certificate for your client application through Thredd Portal.

This guide includes a tutorial for completing this task when you reach the relevant stage in the migration process, and additional certificate lifecycle information.

Using the EHI with mTLS

In an EHIm (External Host Interface and mTLS) setup, your role of your organisation is reversed. Instead, your organisation acts as the *Server* and Thredd is the *Client*. An EHIm setup within Thredd's Secure Framework requires you to install Thredd's Trust Chain on your EHI server.

To migrate your EHIm configuration, you must switch from using the Raidiam Connect trust chain to trusting the Thredd EHI Trust Chain instead. During this process you will coordinate with Thredd, confirming when you are ready to begin and when you have completed the key steps, providing the necessary information to complete the migration. The requirement to have a Server certificate remains, which you obtain from a Certificate of Authority vendor, such as Verizon or Digicert or Amazon Web Services.

To learn more and follow the EHIm migration process, see Migrating to Thredd-issued certificates for EHI with mTLS.

Section 2: Migration overview

This guide covers several different migration pathways, which differ based on the encryption method that a client can use to connect to Thredd. The migration process that you must follow depends on your existing configuration with Thredd and your target configuration state.

There is a Migration Overview for each typical migration pathway, which summarises the different stages of the migration process. Use the Migration Overview as your guide to work through the different stages, and use the links within it to the instructions you must complete for each stage.

To ensure that you follow the correct process, refer to the Migration Overview that matches your target state for connecting to Thredd. You should also familiarise yourself with the Thredd Secure Framework and Thredd Portal overview.

Note that in the following table:

- Existing setups: Raidiam Connect (Certificate Authority) and Cloudentity are responsible for Identity and Access Management (IAM).
- Target setups: Thredd Portal is responsible for Identity and Access Management (IAM).

Tip: For the latest technical documentation, see the Documentation Portal.

Existing setup	Target setup	Relevant Migration Overview and additional pages
Mutual TLS with: • Raidiam Connect and Cloudentity for IAM • Thredd Portal for cards and transaction management • External Host Interface over mTLS (EHIm) • REST API and/or API Hub	Mutual TLS with: • Thredd Portal for IAM • Thredd Portal for cards and transaction Management • External Host Interface over mTLS (EHIm) • REST API and/or API Hub	Migrating from mTLS to a new mTLS connection to Thredd, REST APIs and API Hub
Mutual TLS with: • Raidiam Connect and Cloudentity for IAM • External Host Interface over mTLS (EHIm) • REST API and/or API Hub • Smart Client (cards and transaction management)	Mutual TLS with: • Thredd Portal for IAM • External Host Interface over mTLS (EHIm) • REST API and/or API Hub • Smart Client (cards and transaction management)	Migrating from mTLS to a new mTLS connection to Thredd, REST APIs and API Hub and Smart Client Migration
Mutual TLS with: • Raidiam Connect and Cloudentity for IAM • External Host Interface over mTLS (EHIm) • REST API and/or API Hub • SOAP Web Services • Smart Client (cards and transaction management)	Mutual TLS with: • Thredd Portal for IAM • External Host Interface over mTLS (EHIm) • REST API and/or API Hub • SOAP Web Services • Smart Client (cards and transaction management)	Migrating from mTLS to a new mTLS connection to Thredd, REST APIs, API Hub and SOAP Web Services and Smart Client Migration
Mutual TLS with: • Raidiam Connect and Cloudentity for IAM • Thredd Portal for cards and transaction management • External Host Interface over mTLS (EHIm) • SOAP Web Services	Mutual TLS with: • Thredd Portal for IAM • Thredd Portal for cards and transaction management • External Host Interface over mTLS (EHIm) • SOAP Web Services	Migrating from mTLS to a new mTLS connection to Thredd and SOAP Web Services only

Existing setup	Target setup	Relevant Migration Overview and additional pages
Mutual TLS with: • Raidiam Connect and Cloudentity for IAM • Smart Client (cards and transaction management) • External Host Interface over mTLS (EHIm) • SOAP Web Services	Mutual TLS with: • Thredd Portal for IAM • Smart Client (cards and transaction management) • External Host Interface over mTLS (EHIm) • SOAP Web Services	Migrating from mTLS to a new mTLS connection to Thredd and SOAP Web Services only and Smart Client Migration

Migrating from mTLS to a new mTLS connection to Thredd and API Hub

Thredd's scalable security architecture roadmap is aligned with emerging standards and regulatory requirements. In the context of this work, Thredd has enhanced its *Secure Framework*. As a result, Thredd is phasing out support for Raidiam certificates and Cloudentity. You will have used Cloudentity as the identity provider for Thredd interfaces and OAuth OpenID provider for application, access token and access policy management.

Thredd's Secure Framework is the combination of several components which enables clients to access Thredd's resources securely over mTLS (Mutual Transport Level Security). Thredd Portal sits at the centre of this, enabling Admin users to manage their organisation's applications, users, and credentials for accessing Thredd services through a single user interface and dashboard.

Migrating your connectivity to Thredd's enhanced Secure Framework and managing your organisation and access through Thredd Portal offers a higher level of assurance and security for your data and services. To learn more, see Thredd Secure Framework.

Who should read this guidance

This guidance applies to you if you:

- currently connect to Thredd applications and services using mTLS (Mutual Transport Layer Security)
- currently use Transport Certificates issued by Raidiam Connect
- currently use Cloudentity as an identity provider and OAuth OpenID provider
- will continue to use mTLS – you will change the your Certificate Authority by using certificates issued by Thredd through Thredd Portal

This migration process also applies to you if you use any of the following applications and services (you do not need to be currently using all of these services):

- Thredd REST APIs
- Thredd API Hub
- Thredd Portal
- Thredd Smart Client
- Thredd PAN Finder
- Webhooks/Event Delivery System (EDS)
- Thredd's External Host Interface (EHI) application

Migration end state

When you have completed the migration process, your connectivity to Thredd will comprise of mTLS with client credentials.

Thredd has prepared updated guidance to help walk you through the migration process. It comprises several stages, depending on the different services that you use.

It will guide you through your migration so that you can use the following services at a minimum:

- Thredd Portal
- Thredd REST APIs and API Hub using a Mutual TLS connection
- EHI with Mutual TLS (EHIm)
- Smart Client – existing users only (requires download of the latest version)
- PAN Finder – existing users only (requires download of the latest version)

You can begin the process of migrating to the Thredd Secure Framework once you receive confirmation from Thredd.

Thredd Portal overview

Thredd Portal is the main entry point to Thredd services and is the main component that is related to the API Hub. Before you can use the API Hub, your organisation must be set up on Thredd Portal and the Thredd Secure Framework. These combine several components that enable secure access to Thredd's resources, using a common identity store.

This enables you to use the latest version of Thredd Portal for:

- Managing your organisation, applications, OAuth Client, users and user roles, by your registered Organisation Admin users (all clients).
- Card and transaction management by users who are assigned an appropriate role (for clients already using Thredd Portal or clients that Thredd has advised can use this functionality).

Note: If you currently use Smart Client to manage cards and transactions, you can continue to do by downloading the latest version. For more information, see Smart Client migration.

This migration process will onboard you to the latest version of Thredd Portal – this includes existing Thredd Portal users and Smart Client users. To learn more, see Thredd Portal overview.

Preparation and testing considerations

This migration process comprises several steps and requires you to update the configuration of your system and how you connect to the Thredd platform. You should prepare for this by reading the migration guidance, so that you can ensure a smooth transition.

For best practice, Thredd recommends that you first set up your organisation's access in the UAT (User Acceptance Test) environment. This enables you to test and verify that your organisation can use Thredd applications and API endpoints successfully.

A successful migration to UAT involves replicating production environment settings, but updating environment-specific configurations such as API keys, and passing your testing. Key tasks during the testing phase include verifying that your organisation can access Thredd applications, such as Thredd Portal, and use API endpoints successfully.

Once you have verified that the migration to the UAT environment is satisfactory, you can coordinate with Thredd and prepare for migrating to Production. You will need to repeat all of these migration steps for Production, ensuring that environment-specification configurations are correct. This includes requesting new certificates and credentials for your client application and configuring the latest Postman Collection.

Overview of the key migration stages

You must complete the following steps:

1. An Organisation Admin user must set up their access to Thredd Portal and then review details for their organisation's application and OAuth Client. They can also review user access details, depending on whether your organisation's users access Thredd Portal and other services via Single Sign-On (SSO) or use passwords to log in.
    - If you will use Single Sign-On to access Thredd applications, see Migrating Single Sign-On access and users to the Thredd platform.
    - If you will use password log in to access Thredd applications, see Migrating password access and users to the Thredd platform.
2. Create a new Application and OAuth Client Thredd Portal. Before you can access Thredd's REST API, you must make sure that you have an application and an OAuth Client in Thredd Portal. You can then request client credentials for your application and configure your REST interface to use them to request an access token to include with your API calls. This allows Thredd to verify your identity and grant access. See Creating an Application and OAuth Client.
3. Request the certificates and create credentials that your application requires to obtain and access token and access Thredd's REST APIs and API Hub. See Connecting to the REST APIs.
4. Download and configure the latest Postman Collection to test your connection to Thredd's API Hub to verify that it is working as you expect. This updated collection contains all of the endpoints that you require. See Using the Postman Collection to call REST APIs over mTLS.

    Note: If you have a previous version that you configured with credentials from Cloudentity and Raidiam Connect then you must not use it. The latest Postman Collection is specifically designed for use with the latest Thredd Secure Framework and credentials associated with your client application in Thredd Portal.
5. Review and set up access to Thredd Portal and other applications – only if you did not complete this in step 1. An Organisation Admin must set up access to the Thredd platform for their organisation's users.
    - For the SSO method, this involves inviting users to log in to Thredd Portal via your organisation's identity provider. Once users have logged in for the first time, you can assign the appropriate roles to each user to ensure that in they have the appropriate access to Thredd Portal and applications. See Migrating Single Sign-On access and users to the Thredd platform.
    - For the password login method, this involves manually adding users in Thredd Portal. You can assign the appropriate roles to them, and use the built-in invitation feature to send an invite to the user to log in to Thredd Portal. See Migrating password access and users to the Thredd platform.
6. Migrate other Thredd applications. At a minimum, you must migrate your EHI (EHI with mTLS) configuration.

Migrating Smart Client (Smart Client users only)

Clients that use and will continue to use the Smart Client application to manage cards and transactions must update to the latest version of Smart Client.

For more information, see Smart Client Migration.

Note: Do not follow this guidance if you use Thredd Portal for your cards and transaction management.

Migrating PAN Finder (PAN Finder users only)

Clients that use and will continue to use the PAN Finder application must update to the latest version of PAN Finder. You must already have updated to the latest version of the Smart Client application before installing PAN Finder.

For more information, see PAN Finder Migration.

Migrating your EHIm configuration

Update your EHIm configuration to use the new Thredd EHI Trust Chain. You need to download the new Thredd EHI Trust Chain, configure the changes on your Server to trust it, and ask Thredd to switch the EHI Client Certificate once you are ready.

For more information, see Migrating to Thredd-issued certificates for EHI with mTLS.

Migrating to Production after successful UAT testing

Once you have verified that the migration to the UAT environment is satisfactory, you can coordinate with Thredd and prepare for migrating to Production. You will need to repeat all of these migration steps for Production, ensuring that environment-specification configurations are correct. This includes requesting new certificates and configuring the latest Postman Collection.

1 Migrating your webhook integration

Thredd is moving webhook payload signing to an AWS-based signing service.

If you already receive Thredd webhooks, sometimes referred to as the Thredd Event Delivery System or EDS, update your webhook receiver to verify AWS-signed payloads before changing your webhook configuration. When your receiver is ready, update the webhook registration to set CertificateAuthority to AWS and verify incoming payload signatures using the Thredd Portal JWKS URL.

Note: Your API integration through API Hub does not change as part of this Webhook migration. However, you will need to update your REST API authentication configuration separately as part of the wider migration effort.

Prerequisites

Before migrating:

- Identify all webhook registrations that you need to migrate.
- Know which Thredd environments each webhook uses.
- Have the JWKS URL for each environment.
- Can update and test your webhook signature verification logic.

Note: Do not change your webhook to use AWS signing until your webhook receiver can successfully verify AWS-signed payloads. Otherwise, your system might reject valid webhook deliveries.

How AWS webhook signing works

When AWS signing is enabled, Thredd signs each webhook payload using an AWS-managed signing key.

Your webhook receiver:

1. Receives the signed webhook.
2. Reads the key identifier (kid) from the signature information.
3. Retrieves the available public keys from the Thredd JWKS endpoint.
4. Selects the public key that matches the kid.
5. Verifies the webhook signature.
6. Processes the webhook only if signature verification succeeds.

Your implementation should also support key caching and key rotation.

Migrate your webhooks

Identify your webhook registrations

Identify all existing Thredd webhook registrations and the environments in which they are used.

Migrate and test each environment separately.

Update your signature verification

Update your webhook receiver so that it can:

- Retrieve public signing keys from the appropriate Thredd JWKS endpoint.
- Identify the required public key using the kid supplied with the webhook.
- Verify the webhook payload signature before processing the event.
- Refresh the JWKS data when the required key is not available in your cache.
- Handle verification failures without processing unverified webhook data.

Reject webhook payloads that are unsigned, incorrectly signed, expired, or have been modified.

Retrieve your JWKS endpoint

Thredd's Public Key Infrastructure (PKI) provides a JSON Web Key Set (JWKS) endpoint that you can use to validate signatures.

An Organisation Admin must log in to the Thredd Portal and obtain the JWKS endpoint details.

To view your JWKS endpoint in Thredd Portal:

1. Log in to Thredd Portal.
2. Navigate to the System Admin menu, select Organisation, and then select EHI Configuration.
3. The EHI Configuration page opens. Locate the required JWKS endpoint and copy its URL.

The JWKS endpoint depends on the environment sending the webhook:

- Production: https://jwks.threddid.com/{thredd_org_id}/jwks.json
- UAT: https://jwks.cloud-uat.threddid.com/{thredd_org_id}/jwks.json

Note: Use the JWKS URL that matches the webhook environment. Using a JWKS URL from a different environment will cause signature verification to fail.

Test AWS signing

Test the integration in a non-production environment before migrating production.

Confirm that your webhook receiver:

- Accepts correctly signed webhook payloads.
- Rejects payloads with an invalid signature.
- Rejects payloads that have been modified.
- Can retrieve and cache the signing keys.
- Can select the correct key using the kid.

Enable AWS signing

After your webhook receiver has been updated and tested, update the webhook registration and set:

```
{
    "config": {
        "url": "string",
        "customHeaders": null,
        "certificateAuthority": "AWS"
        }
}
```

Setting certificateAuthority to AWS tells Thredd to use AWS signing for webhook payloads.

Update webhook registration using the Update Webhook Details endpoint

You can update a webhook's registration details by making a PUT request to the Update Webhook Details endpoint, including the webhookId in the URL. For example:

```
PUT https://api.thredd.com/eds/api/v1/Webhooks/{webhookId}
```

Note: When updating the webhook, you must include all fields in the request body even if they're not being updated. The full webhook details should be noted down using the Get Webhook Details endpoint before updating the webhook.

Example Update Webhook Details request:

```
{
    "productId": "123",
    "events": "[101, 102]",
    "config": {
        "url": "https://client_domain.com/webhook",
        "certificateAuthority": "AWS",
        "customHeaders": "{\"key1\": \"value1\", \"key2\": \"value2\", \"key3\": \"value3\"}"
        }
}
```

If successful, a 200 response is returned with the updated details of the webhook.

For more information, see:

- Update Webhook Details - Field Descriptions
- Update Webhook Details - API Explorer

Validate webhook delivery

After enabling AWS signing, confirm that:

- Webhooks continue to reach your endpoint.
- Signature verification succeeds.
- Your endpoint returns the expected HTTP response.
- There is no unexpected increase in failed deliveries or retries.

Migrate your remaining webhooks

You can migrate webhook registrations individually. Existing and AWS-based signing can coexist during the migration period, so you do not need to move every webhook at the same time.

After all of your webhook registrations have been successfully migrated, you can remove any dependencies in your application that are used only for the previous signing method.

Migration checklist

Before completing your migration, check that you have:

- Identified all webhook registrations and environments.
- Obtained the correct Thredd JWKS URL.
- Added JWKS key retrieval to your webhook receiver.
- Added key selection using `kid`.
- Implemented webhook signature verification.
- Implemented key caching and refresh behaviour.
- Tested signature verification in a non-production environment.
- Confirmed that invalid or modified payloads are rejected.
- Set `certificateAuthority` to `AWS`.
- Validated webhook delivery after the change.
- Repeated the migration for each required environment.

For more information on the Webhook authentication process, see the Connecting to Thredd guide.

Section 3: Migrating user access

A Organisation Admin user can manage user access to Thredd Portal and other Thredd interfaces through Thredd Portal. This includes the log in method and individual user profiles and their Thredd Portal Roles.

This guidance focuses on a Organisation Admin setting up their own registration to Thredd Portal and confirming the log in method for users within your organisation.

Topics covered in this section:

- Thredd Portal overview
- Migrating Single Sign-On access and users to the Thredd platform
- Setting up password login to Thredd Portal
- Understanding Roles
- Understanding Scopes

Thredd Portal overview

Thredd Portal is the main entry point to Thredd services and is the main component that is related to the API Hub; before you can use the API Hub, your organisation must be set up on Thredd Portal and Thredd Secure Framework. These combine several components that enable secure access to Thredd's resources, using a common identity store.

Thredd Portal provides a number of features, including:

- Organisation and application management – Organisation Admin users. Manage access for your organisation's applications, edit/add SSO details, create and manage certificates, generate and validate access tokens and certificates, and more.
- User management – Organisation Admin users. Invite users to onboard to Thredd and manage user roles, permissions, and access, including the option to deactivate and reactivate users.
- Access to Webhook management in Thredd Portal – for users with the Developer role.
- Cards and transaction management, and other related functionality – for users with the appropriate roles.

Note: If you currently use Smart Client to manage cards and transactions, then you can continue to use it for these purposes after migrating your organisation and user access management to Thredd Portal.

Identity and Access Management

Thredd Portal provides identity and access management functionality at an organisation, application, and user level. Thredd Portal functions as a *Confidential Client*, where Thredd's own application infrastructure undertakes authentication and authorisation activity on behalf of the user.

Once a user is registered and has logged in to Thredd Portal, they can access other Thredd services. Thredd Portal also operates behind-the-scenes as an authentication server, enabling a single sign-on journey and access to Thredd's REST APIs.

Thredd Portal also provides access to Thredd's Certificate Authority for setting up and managing certificates to connect to Thredd services, including the REST APIs via API Hub.

To learn more, see Thredd Secure Framework.

Connecting to Thredd Portal

Thredd Portal provides support for multiple roles, with Admin access available to an Organisation Admin. An Organisation Admin user can manage the settings for your organisation's applications, certificates, SSO configuration, and users through Thredd Portal.

This guidance focuses on an Organisation Admin completing the migration to Thredd Portal, and checking the settings for your organisation and users.

Thredd Portal supports the following methods for users to log in to Thredd services:

- Single Sign-On (SSO) – if your organisation already uses SSO, then Thredd migrates your settings to Thredd Portal.
- Email address and password – this involves manually adding details for each user in Thredd Portal and sending an invite using its built-in invitation feature.

About Single Sign-On access (optional)

Note: SSO configurations and users and will not be automatically migrated over as part of the migration process. SSO configurations and users will need to be re-added.

Thredd's Secure Framework allows you to optionally use SSO, using *SAML* (Security Assertion Markup Language) or *OIDC* (OpenID Connect), to access Thredd services, for example Thredd Portal. This not mandatory but Thredd recommends that your users log in using SSO instead of using a password because it offers the following benefits:

- An enhanced user experience for users as it removes the hassle of remembering passwords.
- Companies to save time on maintenance.
- Reductions in overheads when managing accounts.
- Enables you to automate user onboarding, where your users can visit the Thredd Portal URL and automatically create their account upon their first successful login.

This enables a Single Sign-On journey by linking your IdP with Thredd's own provision. If you do not use an IdP, Thredd can act as the IdP.

This Single Sign-On journey is involved with:

- Accessing the organisation and user management features of Thredd Portal for Organisation Admin users, such as creating certificates with Thredd's CA, and the user management functionality to set up access for other users.
- Accessing the card and transaction management features of Thredd Portal.
- Managing access behind-the-scenes to the REST API.

Setting up access to Thredd Portal

An Organisation Admin user at your organisation must first register and log in to Thredd Portal before other users within your organisation can access Thredd Portal and other services.

The set up guidance outlines the full process of setting up the Organisation Admin for the first time and setting up additional users. However, an Organisation Admin does not have to invite/add users to register with Thredd Portal straight away. Instead, the Admin can choose to first check the organisation's SSO details (if using SSO), check or create credentials for your client applications, and test access to Thredd REST APIs and API Hub. An Organisation Admin can later return to this step to check access for the organisation's users.

Summary of the steps

The following is a summary of tasks that Thredd completes:

1. Thredd registers your organisation.
2. Thredd registers an Organisation Admin user for your organisation and sends an email inviting them to log in to Thredd Portal.

   Thredd registers an Organisation Admin user for your organisation and sends an email inviting them to log in to Thredd Portal.
3. Organisation Admin recreates required SSO configuration settings in the Thredd Portal.
4. Organisation Admin recreates required application and OAuth Client settings in the Thredd Portal.

An Organisation Admin user at your organisation can then complete the following tasks:

1. Log in to Thredd Portal.
2. Manage the SSO settings for your organisation (necessary only if your organisation uses SSO to log in to Thredd services).
3. Manage user access:
   - If using SSO – invite users via your SSO/identity provider to sign in to Thredd Portal.
   - If not using SSO – manually invite users to sign in using the **Invite User** feature in Thredd Portal via **System Admin** > **User Management**.
4. Check the roles assigned to users within Thredd Portal, making any changes if necessary.

Set up guides

To learn more about configuring access for your organisation, applications and users in Thredd Portal, see:

- Migrating Single Sign-On access and users to the Thredd platform
- Setting up password login to Thredd Portal
- Understanding Roles
- Understanding Scopes

If your organisation will use Single Sign-On to access Thredd services, make sure that you check your SSO provider settings and make any changes as required.

Tip: You can learn more about the cards and transaction management features of Thredd Portal in Thredd Portal Guide: Getting Started.

Migrating Single Sign-On access and users to the Thredd platform

Thredd's Secure Framework supports industry standard identity and access management and protocols for authentication, Single Sign-On (SSO) with token lifecycle management, secure role-based access control, and secure network and API-level connectivity. Thredd's scalable security architecture roadmap is aligned with emerging standards and regulatory requirements.

In the context of this work, Thredd has enhanced the Identity Provider (IdP) capability for Thredd's interfaces, including Thredd Portal and its Certificate Authority.

These enhancements enable users with an Organisation Admin role to manage settings and access for their organisation, users, including SSO configuration, and applications through Thredd Portal.

Using Single Sign-On (SSO) enables you to automate user onboarding; you can invite users to log in to Thredd Portal URL and automatically create their accounts. This eliminates the need to manually add and invite users in Thredd Portal.

Migrating from Cloudentity for existing clients

As part of the enhancements to the Thredd platform, Thredd is phasing out support for the Cloudentity identity provider.

If your organisation currently uses Cloudentity, you can migrate to Thredd Portal and the current identity and access management platform after you receive confirmation from Thredd that your migration is ready.

Note: For best practice, Thredd recommends that you complete and verify the migration in the UAT environment before completing the equivalent setup in Production.

Before you begin

- You must have the Organisation Admin role for your organisation.
- You must have received confirmation from Thredd that your organisation is ready to migrate.
- You must have access to your organisation's identity provider configuration so that you can verify the migrated SSO settings.

Step 1: Log in to Thredd Portal for the first time

Once Thredd has completed the initial step of registering your organisation and Organisation Admin user in the platform, it sends an invitation to the Organisation Admin containing a link to log in to the Thredd Portal and set up your access as the Organisation Admin.

You need to refer to the email to log in to the Thredd Portal for the first time.

1. In Thredd Portal, navigate to the User Management area within System Admin, then select **Add SSO Provider**.
2. Enter the details for the identity provider, including its name and a description.
3. Select the protocol used by your identity provider: **SAML 2.0** or **OpenID Connect (OIDC)**.
   - **SAML 2.0** – provide the SAML metadata using one of the following methods:
     - **Metadata URL** – enter the metadata URL provided by your identity provider.
     - **Metadata XML** – provide the XML metadata supplied by your identity provider.
   - **OpenID Connect (OIDC)** – provide the required configuration details from your identity provider. Depending on your configuration, these can include:
     - Authorization Endpoint
     - Client ID
     - Client Secret
     - Discovery URL
     - Issuer
     - Response Mode
     - Response Type
     - Scopes

- Token Endpoint
- User Info Endpoint

Complete all fields marked as required in Thredd Portal. Other fields depend on your identity provider configuration.

4. Configure the attribute mappings between your identity provider and Thredd. The default external attribute is email. Select **Add Attribute** mapping if you need additional mappings.
5. Select **Next**. The Review & Submit page opens.
6. Review the details and select **Save Configuration**.
7. When you save the configuration, Thredd Portal runs a health check to validate it. If validation succeeds, the identity provider is added as an active SSO configuration. If validation fails, review the reported errors and correct the configuration.

Note: If you want to use more than one SSO configuration, you can create a new configuration via the **User Admin** > **SSO Configuration** area of Thredd Portal. SSO is used to sign in to Thredd Portal and supported services accessed through it. It is not used to authenticate connections to Thredd REST APIs.

Testing API Hub access using the latest Postman Collection

At this point, you can either choose to:

- Continue with setting up access for your organisation's users (as per step 3 of this guide). After this, test making API calls using the Postman Collection.
- Test access to the API Hub first. This involves downloading and configuring the latest Postman Collection to make API calls. You can then return to this guide later to set up access for your organisation's users.

For information about connecting to Thredd REST APIs and using the latest Postman Collection, see Connecting to the REST APIs.

Step 3: Configure user access to the Thredd platform

To ensure that the users for your organisation can access the Thredd platform, you must invite users via your SSO provider. When a new user tries to log in to Thredd Portal, it registers them and assigns the **Read-Only** role by default.

An Organisation Admin can view the users that are registered for their organisation in Thredd Portal, and complete the following tasks:

- Manage user access, deactivate or reactivate a user's access to the platform.
- Assign the appropriate roles and permissions to a user.
- Edit user details, such as the email address that a user logs in with.
- Resend invitations to users.

Note: You must ensure that you configure access for any users who require it, but you can complete this task when you are ready to do so. For example, if you want to first test your organisation's connection to Thredd, such as the REST APIs, and then return to this step.

To view and manage users:

1. Navigate to **System Admin** and select **User Management**.
2. The **User Management** screen appears, and displays a directory of users by default.

   Note: To add more users, you must invite them via your SSO provider.

3. The **Users** screen displays a summary of information about each user in the following columns:
   - **User ID**
   - **First Name**
   - **Last Name**
   - **Email Address**
   - **Role** – displays all roles assigned to a user, which can be more than one role.
   - **Status** – indicates if a user has logged in to Thredd Portal and is active after receiving an invitation and changing their password.
   - **Last logged in**
   - **IP Address**
   - **Actions** – the action menu for each user, which you can use to review or edit the details and role of a user.

4. Configure the attribute mappings between your identity provider and Thredd. The default external attribute is **email**. Select **Add Attribute Mapping** if you need additional mappings.
5. Select **Next**. The Review & Submit page opens.
6. Review the details and select **Save Configuration**.
7. When you save the configuration, Thredd Portal runs a health check to validate it. If validation succeeds, the identity provider is added as an active SSO configuration. If validation fails, review the reported errors and correct the configuration.

Note: SSO is used to sign in to Thredd Portal and supported services accessed through it. It is not used to authenticate connections to Thredd REST APIs.

Step 3: Review user access

After you have confirmed the SSO configuration, review the users registered for your organisation and make sure that they have the appropriate access.

An Organisation Admin can:

- View users registered for the organisation.
- Manage user access, including deactivating or reactivating access.
- Assign the appropriate roles and permissions.
- Edit user details.

If your organisation uses SSO, invite new users through your identity provider. When a new user signs in successfully, Thredd Portal registers the user and you can assign the appropriate roles and permissions.

Organisation Admins can manage users from the User Management area within the System Admin menu in Thredd Portal.

Note: You can review user access after validating your SSO configuration. You do not need to complete user administration before testing other parts of your UAT setup.

Step 4: Continue testing your connection

After you have reviewed your SSO configuration, you can continue testing the services included in your migration.

For information about connecting to Thredd REST APIs and using the latest Postman Collection, see Connecting to the REST APIs.

Example SSO integrations are available in the SSO Sample Integration section of the Connecting to Thredd Guide.

Setting up password login to Thredd Portal

Clients who do not have their own Single Sign-On (SSO)-compliant solution (such as Entra, Google, or Okta) can manually add users and set up their password-based access to Thredd Portal and applications.

An Organisation Admin can set up and manage the settings for your organisation's applications, certificates, and users through Thredd Portal. This guidance assumes that you are the Organisation Admin and need to log in to Thredd Portal for the first time. You can manually add users and set up password-based access when Thredd sends an email to the Organisation Admin user notifying them that it is ready for your organisation to do this.

Note: For best practice, Thredd recommends that you first set up your organisation's access in the test (UAT) environment, so that you can verify that your organisation can use Thredd applications and API endpoints successfully. You can review the user access settings later.

About password (non-SSO) user access

Adding new users who will use a password to access the Thredd platform is a manual process. It requires an Organisation Admin to manually create and invite each individual user via Thredd Portal, which sends an email to users inviting them to log in. Offboarding users is also a manual process, requiring an Organisation Admin users to revoke access for an individual user via Thredd Portal.

Tip: As an optional alternative, Thredd recommends that you configure Single Sign-On (SSO) access to Thredd. This enables you to automate user onboarding by using Just-in-Time (JIT) provisioning, where your users can visit the Thredd Portal URL and automatically create their account upon their first successful login. This eliminates the need to manually create and invite users.

Prerequisites

- You must have the Organisation Admin role for your organisation.
- You must have received confirmation from Thredd that you can start setting up access for you and your organisation's users.

Step 1. Log in to Thredd Portal for the first time

Once Thredd has completed the initial step of registering your organisation and Organisation Admin user in the platform, it sends an email invitation to the Organisation Admin containing a link to log in to Thredd Portal and set up your access as the Organisation Admin.

You need to refer to the email to log in to Thredd Portal for the first time.

1. Click on the link in the email for **Log in to Thredd Portal**. This directs you to the initial access screen.
2. On the next screen, you must create a new password. The password must be at least eight characters in length. Enter and confirm the new password in the separate fields, and then select the **Save and Continue to Thredd Portal** button.

When you log in to Thredd Portal, you can set up access for your organisation's users when you are ready.

Note: If the link does not work or you need support, use the link at the bottom of the email to contact Thredd. Do not reply to the email. It is system-generated and sent from a notification-only address that cannot accept incoming emails.

Testing API Hub access using the latest Postman Collection

At this point, you can either choose to:

- Continue with setting up access for your organisation's users (as per step 2 of this guide). After this, test making API calls using the Postman Collection.
- Test access to the API Hub first. This involves downloading and configuring the latest Postman Collection to make API calls. You can then return to this guide later to set up access for your organisation's users.

For information about connecting to Thredd REST APIs and using the latest Postman Collection, see Connecting to the REST APIs.

Step 2. Check users and configure roles in the Thredd platform

As a first step, an Organisation Admin should ensure users at your organisation are already registered, and add users if necessary. This ensures that they have access to the Thredd platform.

An Organisation Admin can view the users that are registered for their organisation in Thredd Portal, and complete the following tasks:

- Manage user access, deactivate or reactivate a user's access to the platform.
- Assign the appropriate roles and permissions to a user.
- Edit user details, such as the email address that a user logs in with.
- Resend invitations to users

Note: While you must manually add and invite users who require access, you can complete this task when you are ready to do so. For example, if you want to first test your organisation's connection to the Thredd platform, such as to REST APIs, and then return to this step.

To view and add users:

1. Navigate to **System Admin** and select **User Management**.
2. The **User Management** screen appears, and displays a directory of users by default. If you need to add users, you can do so here.
3. The **Users** tab displays a summary of information about each user in the following columns:
    - **User ID**
    - **First Name**
    - **Last Name**
    - **Email Address**
    - **Role** - displays all roles assigned to a user, which can be more than one role
    - **Status** - indicates whether a user is Active or Inactive
    - **Last logged in**
    - **IP Address**

The **Action** column contains the action menu for each user, which you can use to edit the role and permissions for a user.

Adding a new user via User Management

An Organisation Admin user can add, edit and remove users. If only an Organisation Admin user exists then first add an additional Admin user if your organisation requires one. You can either add additional users at the same time or add other users later.

To add a new user:

1. Select the button to create/add a new user.
2. Enter the user's first and last names, email address, and assign a role. You can view a list of the roles available to a user under **Roles**.

    The platform assigns all users a **Read-Only** role by default which allows view-only access to cards and transactions. You can assign a different role to this if required, and you can also assign more than one role to a user.
3. Select the role that is most appropriate for the user based on the description:
    - **Admin Roles**: Admin, Sensitive Information Manager
    - **Developer Roles**: Developer
    - **Cards & Transactions Roles**: Cards Operations Specialist, Card Config Manager, Card Balance Manager
    - **Read-Only Roles**: Read-Only

    Tip: To learn more, see Understanding Roles.
4. When you have entered the user details, review the details and select the option to continue.
5. The **Review and Finish** screen appears. If all user details are correct, select **Confirm and finish** and a message appears to confirm that you have saved the details. To make a change before saving, select **Previous step** to return to the previous screen.
6. When you return to the **User Management** screen, you can check if the new user details have been added successfully.

Repeat the process if you want to add additional users.

Tip: Users who will use the cards and transaction management functionality can find information in the Thredd Portal: Cards and Transaction Management Guide.

Editing an existing user profile

An Organisation Admin can add, edit and remove users. To edit the details of a user:

1. Select the action menu (three dots) for an individual user.
2. Select **Edit User**.

3. The **Edit User** screen appears and displays the user's name, email address, and the role assigned to the user. You can view a list of the roles available to a user under **Roles**. The platform assigns all users a **Read-Only** role by default which allows view-only access to cards and transactions. You can assign a different role to this if required, and you can also assign more than one role to a user.
4. To change the assigned role, select the role that is most appropriate for the user based on its description:
    - **Admin Roles**: Admin, Sensitive Information Manager
    - **Developer Roles**: Developer
    - **Cards & Transactions Roles**: Cards Operations Specialist, Card Config Manager, Card Balance Manager
    - **Read-Only Roles**: Read-Only
5. Select **Next**.
6. The **Review User Details** screen appears. If the role is correct, select **Save**. To make a change before saving, select **Back** to return to the **Edit User** screen.
7. When you are ready to save the user details, select **Save**. A message appears to confirm that the user details have been updated successfully.

Warning: Deactivating a user will revoke their access to the Thredd platform. If you need to restore access, you can reactivate a user.

Understanding Roles

When an Organisation Admin user configures an individual user's profile in Thredd Portal, they can assign one or multiple roles to the user.

Roles are associated with *Users* – they provide the permissions to each individual user within your organisation based on the given *Role*. Thredd Portal Roles are preconfigured; for example, Developer, Manager, and Admin. Organisation Admin users can assign, view and update the Thredd Portal Roles for an individual user in Thredd Portal via **System Admin** > **User Management**.

Available Thredd Portal Roles

The following table describes what each Thredd Portal Role allows a user to do in Thredd Portal.

Thredd Portal Role	Description
Manager	• Transaction Search & View • Remove auth • Card Search & View • Balance adjustment • Card Load/Unload • Change card status • PIN & CVC2 services • Edit cardholder details • Edit card configurations • Extend Thredd Expiry Date • Activate a Card • Balance Transfer
Read Only	View-only access to cards and transactions information. • Card Search & View • Transaction Search & View
Card Operations Specialist	Card lifecycle management and transaction visibility. • Card Search & View • Transaction Search & View • Change card status • Activate a Card • Extend Thredd Expiry Date • PIN & CVC2 services • Remove auth
Card Configuration Manager	Manages card configurations. • Card Search & View • Transaction Search & View • Edit card configurations
Card Balance Manager	Manages balance adjustments and balance transfers. • Card Search & View • Transaction Search & View • Balance Transfer

Thredd Portal Role	Description
	• Balance adjustment
Developer	Configures and manages webhooks for system integrations.
Sensitive Information Manager	Restricted access to sensitive card data, including PAN Finder operations.
Organisation Admin	Full platform control, including configurations and all operational settings.

Assigning Thredd Portal Roles to a User in Thredd Portal

Users are not able to update their own roles within Thredd Portal.

An Organisation Admin user can add a Thredd Portal Roles to a User via their user profile in Thredd Portal.

To view and edit the Roles for a User:

1. Navigate to **System Admin**, then **User Management**.
2. In the list under **User**, locate the user that you want to check and select the **Actions** menu (three dots) next to the specific user, and select **Edit User**.
3. The **Edit User** screen appears and displays the user's Basic Information. Select **Next** to view the Groups and Thredd Portal Roles.
4. Select the field under **Thredd Portal Roles** to display a dropdown menu of the different roles that are available for the user. Assign the appropriate portal roles that will provide access to the specific features and card operations that a user requires. You can also select a group if a group is available. Select **Next**.
5. Review the user's details and select **Update User** to save the settings. If you need to make any changes, select **Back** and repeat the previous steps before updating the user details.

Tip: If the user is logged in to Thredd Portal at the time that the Admin makes any changes, the user might need to log out and then log in again to ensure that the changes take effect.

Understanding Scopes

Thredd assigns products (and scopes) to your organisation during the implementation stage, according to the Thredd services that you will use.

Applications are associated with *Scopes* – these relate to the products and services that your organisation will use in its agreement with Thredd.

When you register an Application for your organisation in Thredd Portal, it is assigned the appropriate scopes. Organisation Admin users can view the available Scopes for their Application in Thredd Portal via **System Admin** > **Applications**.

Available Scopes

The following is an example of some of the scopes that are available from Thredd. These relate to the API endpoints that your organisation has access to. Thredd assigns scopes to your organisation according to the Thredd services that your organisation will use.

Scope Name	Description
cards.read	Enables you to use GET endpoints to return information on cards.
cards.write	Enables you to use the POST, UPDATE and PATCH endpoints to create and update card information.
pin.read	Enables you to use the Retrieve PIN endpoint.
pin.write	Enables you to use the Set PIN and Unblock PIN endpoint.
cvv.read	Enables you to use the Retrieve CVV endpoint.
cvv.write	Enables you to use the Unblock CVV endpoint.
bulkcard.read	Enables you to use the Get Bulk Card Progress endpoint.
bulkcard.write	Enables you to use the Create Bulk Card endpoint.
cards.encrypted	Enables you to use the Get Encrypted Data endpoint.
cards.sensitive	Enables you to use the Get Full PAN endpoint.
3ds.read	Enables you to use the Get 3DS Configuration endpoint.
ads.read	Enables you to use the GET endpoints for Visa Alias Directory.
ads.write	Enables you to use the POST, DELETE and PUT endpoints for Visa Alias Directory.
adxapi	Enables you to use endpoints for the ADX (Cardinal) 3D Secure provider integration.
apata.read	Enables you to use GET endpoints to retrieve Apata 3D Secure configuration/authentication data.
apata.write	Enables you to use POST/PATCH endpoints to create or update Apata 3D Secure configuration.
credit.read	Enables you to use GET endpoints for credit related account or product data.
credit.write	Enables you to use POST/PATCH endpoints for credit-related account or product operations.
cts.read	Enables you to use GET endpoints for Card Transaction Services (CTS) data.

Scope Name	Description
cts.write	Enables you to use POST endpoints for Card Transaction Services (CTS) operations.
issuer.write	Enables you to use POST endpoints for issuer-level configuration or operations.
limits.read	Enables you to use GET endpoints to retrieve card/account limit details (e.g. spend, velocity limits).
limits.write	Enables you to use POST endpoints to set or update card/account limits.
rdxapi	Enables you to use endpoints for the RDX 3D Secure provider integration (alternative to Apata).
scamdetect	Enables you to use endpoints related to scam/fraud detection features (e.g. flagging suspicious transactions).
ws	Enables access to Thredd webhooks/Event Delivery System (EDS).

Smart Client Migration

Smart Client is a cards and transaction management application that Thredd made available to clients before Thredd Portal.

It does not offer identity and access management functionality for your organisation, applications or users. Instead, a Organisation Admin user at your organisation must use Thredd Portal to manage the access and roles for users of Smart Client and other Thredd applications and services.

As a result of Thredd's enhancements to its platform, a new version of Smart Client is available to clients.

To continue using Smart Client, you must download and install the latest version of Smart Client from Thredd.

Note: If you do not use Smart Client, and instead use Thredd Portal for your cards and transaction management, do not follow these steps to download and install Smart Client.

System Requirements

To install the Thredd Smart Client application, you require a computer running Windows 7 or later.

Before you download and install Smart Client, you need to install the prerequisite software:

- Microsoft .Net Framework 5 (x64)
- Microsoft .Net Framework 4.8 (x64)
- Microsoft Visual Studio 2019 (version 16.11.31)

To install the prerequisite software, use Microsoft Edge to click on one of the following links that is relevant to your environment.

Note: For best practice, Thredd recommends that you first set up your organisation's access in the UAT (User Acceptance Testing) environment, so that you can verify that your organisation can use Thredd applications and API endpoints successfully.

Environment	URL
UAT	https://sc-uat.thredd.net/
Production PRDZ over mTLS	https://pz-smartclient.thredd.net/publish.htm
Production PRD1 over mTLS	https://p1-smartclient.thredd.net/publish.htm
Production PRD2 over mTLS	https://p2-smartclient.thredd.net/publish.htm

Migrating Smart Client

To update to the latest version of Smart Client in a UAT (User Acceptance Testing) environment:

1. Check if you already have an existing version of Smart Client installed, and remove any previous version.
2. Download the latest UAT version of Smart Client.
3. Select **Download prerequisites** and save the archive to your local drive.
4. Extract the archive and run/open the file *ReportViewer.exe*, and make sure that the file is not blocked by antivirus or any other software.
5. Once you have installed the prerequisites, select **Launch** and follow the online instructions to access Smart Client.
6. Test the latest UAT version of Smart Client and verify that it works as you expect.

When you are satisfied that you can successfully use the new version of Smart Client, you should inform Thredd and coordinate preparing for moving to your Production environment.

To update to the latest version of Smart Client in a Production environment:

1. Download and install the latest Production version of Smart Client, but otherwise following the previous instructions.
2. Test the latest version of Smart Client and verify that it works as you expect.

3. When you are satisfied that you can successfully use the new version of Smart Client, remove the previous version of Smart Client.

Tip: You can learn more about the cards and transaction management features of Smart Client in the Smart Client Guide.

PAN Finder Migration

PAN Finder is an application that Thredd made available to clients before Thredd Portal.

It enables clients to retrieve the full Primary Account Number (PAN) of a card or the linked Thredd public token.

As a result of Thredd's enhancements to its platform, a new version of PAN Finder is available to clients.

To continue using PAN Finder, you must download and install the latest version of PAN Finder from Thredd.

Note: If you do not use Smart Client or PAN Finder, and instead use Thredd Portal for your cards and transaction management, do not download and install Smart Client and PAN Finder.

System Requirements

To install the Thredd PAN Finder application, you must first download and install the latest version of Smart Client, following the instructions in Smart Client Migration. This requires a computer running Windows 7 or later that meets the same requirements as the Smart Client application.

To install the prerequisite software, use Microsoft Edge to click on one of the following links that is relevant to your environment.

Note: For best practice, Thredd recommends that you first set up your organisation's access in the UAT (User Acceptance Testing) environment, so that you can verify that your organisation can use Thredd applications and API endpoints successfully.

Environment	URL
UAT	https://pf-uat.thredd.net/
Production PRD0	https://panfinder-prd0.thredd.net:3134/PANFinder/publish.htm
Production PRD1	https://panfinder-prd1.thredd.net:3134/PANFinder/publish.htm
Production PRD2	https://panfinder-prd2.thredd.net:3134/PANFinder/publish.htm

Migrating PAN Finder

To update to the latest version of PAN Finder in a UAT (User Acceptance Testing) environment:

1. Check if you already have an existing version of PAN Finder installed, and remove any previous version. If you have previously used a cloud-based version of PAN Finder, remove any bookmarks you may have for it.
2. Download the latest version of PAN Finder.
3. Select **Install** to download and install the app to your local drive.
4. Once you have installed the app, locate the PAN Finder shortcut icon on your desktop and open the application.
5. Test the latest version of PAN FInder and verify that it works as you expect.

When you are satisfied that you can successfully use the new version of PAN Finder, you should inform Thredd and coordinate preparing for moving to your Production environment.

To update to the latest version of PAN Finder in a Production environment, repeat the steps to download and install the application.

Tip: You can learn more about using PAN Finder in the PAN Finder Guide.

Section 4: Migrating access to API Hub

A Organisation Admin user can manage client applications and OAuth Client settings in Thredd Portal. The most common use case is to manage the applications and their credentials that they require to access Thredd's API Hub and REST APIs. To understand the credentials that you need to access API Hub, start by referring to Connecting to the REST APIs.

Topics covered in this section:

- Connecting to the REST APIs
- Requesting certificates for applications using mTLS
- Using the Postman Collection to call REST APIs over mTLS

Creating an Application and OAuth Client

Before you can access Thredd's REST API, you must create an application and an OAuth Client in Thredd Portal to register it and receive a Client ID.

Once you have created an application and OAuth Client, you can request client credentials, and configure your application and a REST interface (such as Postman) to provide credentials when making API calls. Think of the Client ID as the passport for your OAuth Client, and therefore your application. Combined with the credentials you need for your chosen Client Authentication method, it allows Thredd to verify your identity and grant access.

Summary of the steps

The steps for creating a new application and OAuth Client are as follows:

1. Log in to Thredd Portal.
2. Create a new client application, adding a name and description.
3. Create a new OAuth Client for the application.

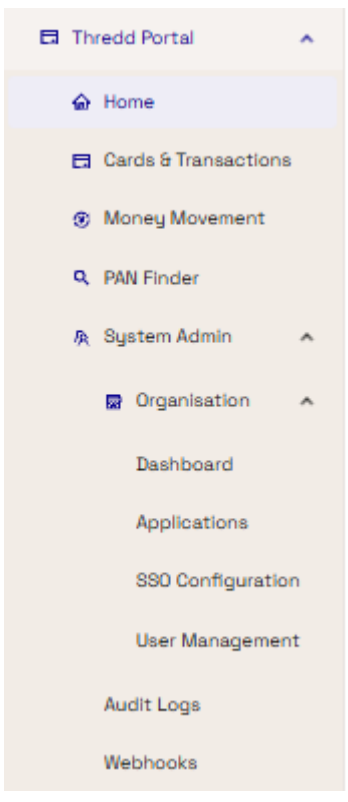
Note: The PMid is automatically added to your OAuth Client if you only have one PM. If you have multiple PMids, then you can assign the PMid in Thredd Portal.

Prerequisites

1. You must have access to Thredd Portal.
2. You must be an Admin for your organisation.
3. You must have installed OpenSSL on the machine (the Server) that will make requests to Thredd platform (the Client).

Step 1: Log in to Thredd Portal

Log in to Thredd Portal and select **System Admin**. In **System Admin** menu, select **Organisation**, then **Applications**.



Step 2: Create a new client application

The **Applications** screen lists any applications that are registered in Thredd Portal.

Note: If an existing application is already present (and you do not need to create a new one), then check whether an OAuth Client already exists for it. Select the **Actions** menu and select **Review Application**. Check the **OAuth Clients** screen; if an OAuth Client is not present, see step 3.

To create a new application:

1. Select the **Add Application** button.
2. The **Create New Application** screen appears. Enter the basic information, such as a name and description, for the application.

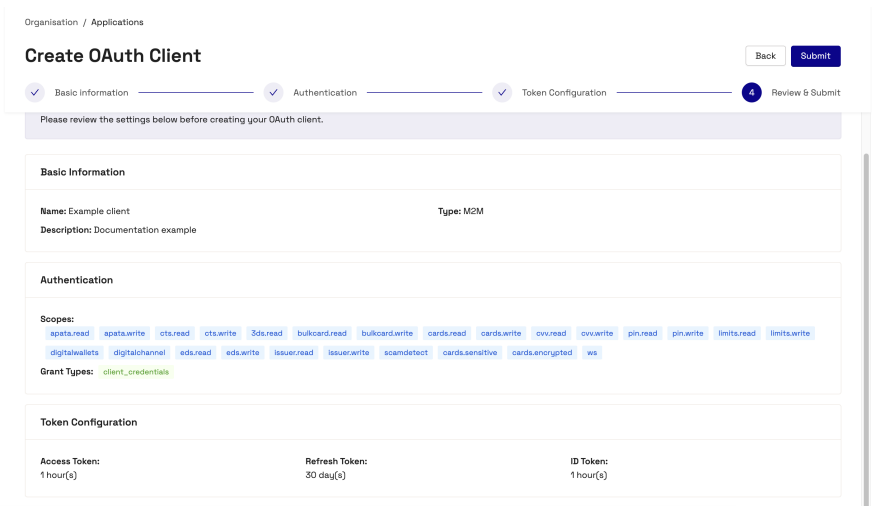
   Tip: Thredd recommends that you use an application name that you can easily identify for any application management activities. For example, to distinguish it from other applications when you need to obtain credentials, such as a Client ID, access token or certificate.
3. Select **Next** and review the new application summary; select **Back** to make changes or select **Create Application** to save it.
4. Your new application now appears in the **Applications** screen. Select the menu (three dots) under **Actions** and select **Review Application**. The application summary opens on the **OAuth Clients** tab by default. See *Step 3. Create a new OAuth Client*.

Step 3: Create a new OAuth Client

The application details has one tab; **OAuth Clients**. MTLS users will have an additional **Certificates** tab. The steps to create an OAuth Client are as follows:

1. In the **OAuth Clients** tab, select **Create Client**.
2. The **Create OAuth Client** screen appears, starting with the **Basic information** stage. Enter the basic information, such as the **Client Name** and **Description**. Select **Next**.
3. The **Authentication** stage summarises the recommended **Security Configuration**. To continue, select **Next**. To change the information, select **Back**.
4. Optional, if you have more than one PMID, select the required PMID. Users with only one PMID will have it auto-selected.
5. The **Review** stage allows you to review your configuration, based on the information that you entered in Basic Information, Authentication, and Token Configuration. You can view the list of available **Scopes** for your application in the **Authentication** section. To proceed, select **Submit**; to make changes select **Back** and then submit the details to finish creating the OAuth Client.

You have now registered a new OAuth Client, with a unique Client ID for your application. You can view it in the **OAuth Clients** screen of your application's details in Thredd Portal.



Next steps

You must make sure that you request and provide the necessary credentials from Thredd to access Thredd's REST APIs.

If you are using an mTLS connection, your application must also present a Transport Certificate from Thredd. Thredd adopts a self-service approach, which allows an Admin user to independently request and manage certificates via Thredd Portal.

See Connecting to the REST APIs.

Connecting to the REST APIs

There are two main areas in which you will need to set up your connection to access Thredd REST APIs:

1. Thredd Portal: Log in to Thredd Portal and obtain the credentials for your existing application and OAuth Client. This is the application that currently connects to Thredd REST APIs via the API Hub. If you need to create a new application and OAuth Client, you can do so here.
2. A REST tool: Configure your REST interface, such as Postman, to interact with the Thredd platform, so that you can access the API end-points.

For best practice, Thredd recommends that an Organisation Admin first sets up your organisation's access in the UAT (User Acceptance Test) environment. This enables you to test and verify that your organisation can use the API endpoints successfully. Once an Admin has verified that your testing in the UAT environment is satisfactory, you can prepare for migrating to a Production environment.

Note: For your Production environment, you will need to request new credentials and ensure that the environment-specific configuration is correct. Make sure that old credentials are not cached, such as a certificate or token, and that you send updated credentials to Thredd.

Overview of set up steps

You need to complete the following steps to migrate your connection to Thredd.

1. Get credentials for your client application via Thredd Portal.
2. Configure your REST tool to interact with Thredd to make calls to Thredd API that include an access token. You must obtain an access token using your client credentials. You can also download Thredd's Postman Collection and configure your client credentials as variables in Postman to test making API calls in a UAT environment.
3. Get an access token from Thredd's OAuth token endpoint to include in the Authorization header of your API request.
4. Make a call to Thredd's REST API to test your setup.

Step 1: Get credentials for your Application via Thredd Portal

To access any Thredd API endpoint you must exchange your client credentials for a short-lived Bearer token (OAuth token).

When you have obtained your credentials, you must request an OAuth token and include it in the header of your requests to the REST API endpoints. To get an OAuth token, you can either use the endpoint within the Postman Collection (when testing in a UAT environment) or call the OAuth Token endpoint directly when using a Production environment (see step 3).

Complete the following steps to obtain the information you need via Thredd Portal, based on your Client Authentication method.

Note: You must have created an application and OAuth client in Thredd Portal. If you have not already done this, see Creating an Application and OAuth Client.

Client Secret authentication

You need:

- Client ID of your application
- Access token from Thredd's OAuth Token endpoint
- If you use mTLS, you also require a Transport Certificate from Thredd

Get the Client ID:

1. In Thredd Portal, navigate to **System Admin**, and select **Applications**. Select the **Actions** menu (icon of three dots) and select **Review Application**. A screen appears with tabs for the **OAuth Clients** and **Certificates** for your Application.
2. On the **OAuth Clients** tab, locate the OAuth Client that you will use to connect to the Thredd REST APIs. Select the **Actions** menu (icon of three dots) and select **View Details**. The **Overview** screen for the OAuth Client appears.
3. On the **Overview** tab, locate the **Client ID** and **Client Secret** under Client Configuration. You can copy these values from here when you need to.

Get a Transport Certificate – for mTLS connections:

If you are using an mTLS connection, your application requires a Transport Certificate for establishing mutual connections between your Client and Thredd. An Admin user can obtain a Transport Certificate from Thredd Portal. See Requesting certificates for applications using mTLS.

Step 2: Configure your REST tool to make calls to Thredd API using your client credentials

You can access Thredd's API Hub and REST APIs over mTLS. You must make sure that you configure your REST tool with the appropriate settings, for example:

- Base URL – this depends on whether you are using the UAT or Production environment.
- Headers – you must set the correct Authorization, Content-Type, and X-Region headers.

Base URL

- Connectivity for the UAT environment: uat-api.thredd.com
- Connectivity for the Production environment: api.thredd.com

Configuring headers

Whether you are using your own tool or the Postman Collection, you must make sure that you configure the following headers with the correct information in each API request. These headers are mandatory for all to the API Hub.

- Authorization header – include your OAuth token in the authorization header as appropriate for your application's authentication method.
- X-Region header – this is mandatory and determines the environment you are trying to connect to. Specify one of the following values in the X-Region header:
  - Use 0 for the Prd 0 (Production 0) environment
  - Use 1 for the Prd 1 (Production 1) environment
  - Use 2 for the Prd 2 (Production 2) environment

  Note: If you are not sure which environment to use, contact your Thredd Account Manager or Implementation team. By default, the Postman Collection is set to use the Production 0 environment (Prd0), but you can change this in the variables section.
- Content-Type header – specify the media type of the requested resource in the content-type header. Example value: application/x-www-form-urlencoded

For some endpoints, you can pass pagination values. Where this is relevant, this is stated in the documentation for a given endpoint in API Reference on the API Hub.

Thredd provides Postman Collections that enable you to use Postman to test using the Thredd REST APIs in a UAT environment. To use the collection, you need to configure Postman with the appropriate environment variables and credentials.

Note: If your integration currently uses an environment-specific API URL, such as https://coreapi-prdX.threddpay.net, update it to use the current production API Hub endpoint: https://api.thredd.com/core/, and ensure the X-Region header is set to the correct value for your required environment. For UAT, use https://uat-api.thredd.com/core/ instead.

Postman Collection

Download the Postman Collection from the API Hub and follow the guide:

- For mTLS connections: Using the Postman Collection to call REST APIs over mTLS
- To download the Postman Collection, see Accessing the API Hub.

Step 3: Get an access token using your client credentials

You must request an access token from Thredd and include it in the Authorization header of your requests to the REST API endpoints. You can get an access token using the following methods:

- Use the endpoint within the Postman Collection (when testing in a UAT environment).
- Call the OAuth Token endpoint directly when using a Production environment.

Access tokens are valid for fifteen minutes and include the scopes that are associated with your application.

Follow the advice that match your Client Authentication method.

Client Secret authentication

You must provide your:

- Client ID
- Client secret

Endpoint:

```
POST https://threddid.com/oauth2/token
```

Example request body (URL-encoded):

```
curl --location 'https://threddid.com/oauth2/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'grant_type=client_credentials' \
--data-urlencode 'client_id=YOUR_CLIENT_ID'
--data-urlencode 'client_secret=YOUR_CLIENT_SECRET'
```

If you use an mTLS connection, you must also include your Transport Certificate in your request to the OAuth token endpoint. The Authorisation Server will reject all requests without this.

Example OAuth token response

```
{
  "access_token": "eyJhbGciOi...",
  "token_type": "Bearer",
  "expires_in": 3600
}
```

Access tokens are valid for one hour and include the scopes that are associated with your application.

You must configure your application to refresh the token automatically when it expires.

Step 4: Make a call to Thredd's REST APIs

Once you have the `access_token`, you must include it in the Authorization header of every request to Thredd's REST APIs.

The next step is to test making an API call. Refer to the documentation for the base URLs and API Reference for endpoints at the API Hub: https://cardsapidocs.thredd.com/v2.0/docs

Remember to include the following in your requests:

- Base URL – this depends on whether you are using the UAT or Production environment.
- Mandatory information in your request headers.
- Correct HTTP method for the endpoint.

If you change the environment or region that you want to make API calls to, you must update the Base URL and headers that you include in your requests.

Example request header (mTLS)

```
GET /core/api/v1/products HTTP/1.1
Host: api.thredd.com
Authorization: Bearer eyJhbGciOi...
Content-Type: application/json
```

Note: For mTLS, when making your request to `uat-api.thredd.com` (UAT environment) or `api.thredd.com` (production environment) you must also include your Transport Certificate in the request. Thredd's APIs will reject all requests that do not include this.

Requesting certificates for applications using mTLS

Client applications require a Transport Certificate in order for you to access Thredd's REST APIs over an mTLS connection.

Whether you want to generate a new certificate, or renew or revoke a certificate, you can manage certificates through Thredd Portal.

Note: You must use certificates from Thredd to connect to Thredd APIs. Using self-signed or third-party certificates, such as from Raidiam Connect, may result in Thredd refusing connections.

Purpose of Transport Certificates

A *Transport Certificate* is mandatory for mTLS and is used solely for establishing the secure mutual connection between the Server and the Client. Without a valid, signed Transport Certificate, your application is unable to connect to Thredd's secure API endpoints.

Certificate Lifecycle Management

Regularly rotating (refreshing) your certificates is essential to maintain secure communication between your applications and the Thredd platform. New certificates are valid for 350 days; Thredd sends an email to you when a certificate is approaching its expiry date, notifying you 60 days in advance. If you want to renew a certificate, you should request a new Transport or Signing Certificate and rotate it before the date that it expires. This ensures that your application can continue to access Thredd applications.

You can also revoke a certificate at any time by logging in to Thredd Portal and choosing the option to revoke it in the **Certificates** tab of your Application. Fulfilling a revocation request can take up to one hour for Transport Certificates and is instant for Signing Certificates.

Summary of the steps

The steps for obtaining a certificate are as follows:

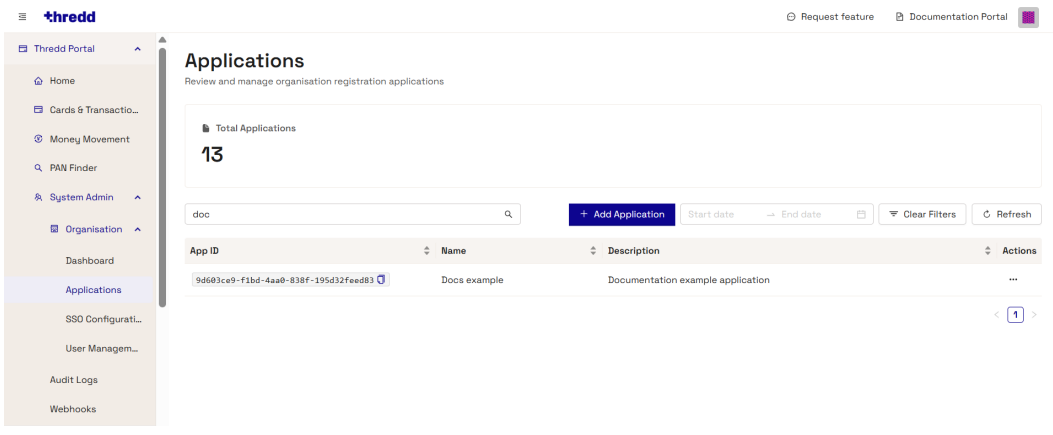
1. Log in to Thredd Portal and select or create the application that requires a certificate.
2. Create a private key and Certificate Signing Request for a new Transport Certificate.
3. Upload the Certificate Signing Request for your application.
4. Download the certificate for your application.
5. Store the private key on your client and ensure it can use the Transport Certificate.
6. Configure your Client to use the Transport Certificate in your requests.

Prerequisites

1. You must have access to Thredd Portal.
2. You must have an Admin role in Thredd Portal.
3. You must have installed OpenSSL on the machine (the Server) that will make requests to Thredd platform (the Client).

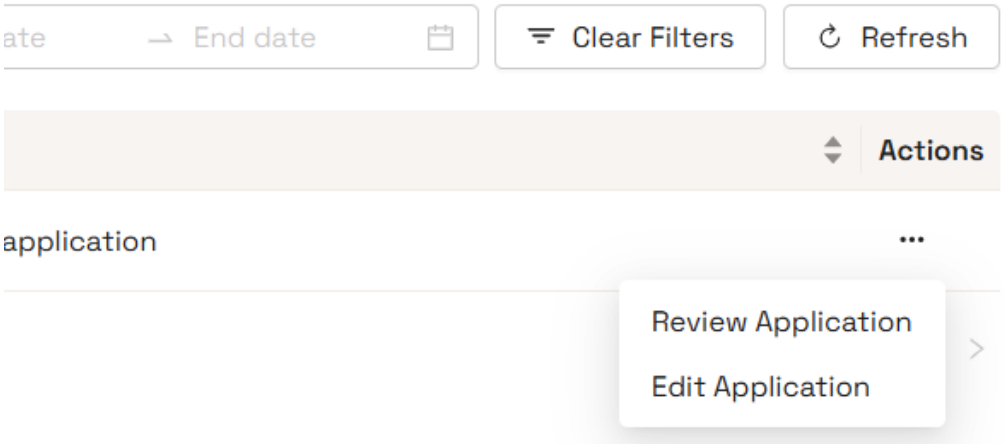
Step 1: Log in to Thredd Portal and select or create the application that requires a certificate

Log in to Thredd Portal and select **System Admin**. From the **System Admin** menu, select **Applications**. You will have access to at least one organisation for your parent company. The **Applications** screen lists any applications that are registered in Thredd Portal.



If an existing application is already present and you want to create a signing certificate for it, do the following:

1. Select the **Actions** menu (three dots) and select **Review Application**.



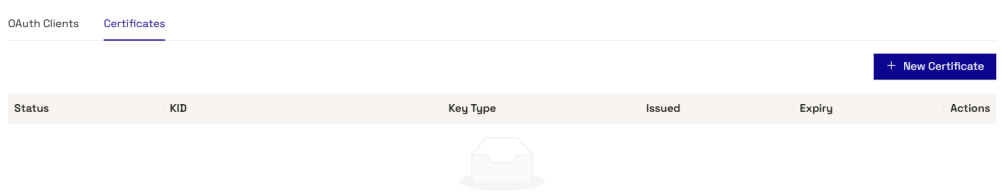
2. The application summary screen appears and displays the **OAuth Clients** screen. Select the **Certificates** tab. See step 2 to create a certificate.

Tip: If you need to create a new application, you can do so by selecting the **Add Application** button. Follow the on-screen prompts to create the Application, and then create an OAuth Client for it. For a full guide, including an explanation of the different Application Types and their associated Security Configurations, see Creating an Application and OAuth Client.

Step 2: Review the application details and create a new Transport Certificate

The **Applications** screen lists an entry for your application once it is registered in Thredd Portal. To request a Transport Certificate:

1. Select the Action menu (three dots) under **Actions** and select **Review Application**.
2. The **OAuth Clients** screen appears; it displays a summary of your application details and permissions. Check that these details are correct, and only update any details if you need to. Specific configuration details are under **Client Configuration**, such as the Client ID. The **Permissions** tab lists the available **Scopes** for your organisation.
3. Next, select the **Certificates** tab and select **New Certificate** button. The **New Certificate** screen appears.



Step 3: Create a private key and Certificate Signing Request for a Transport Certificate

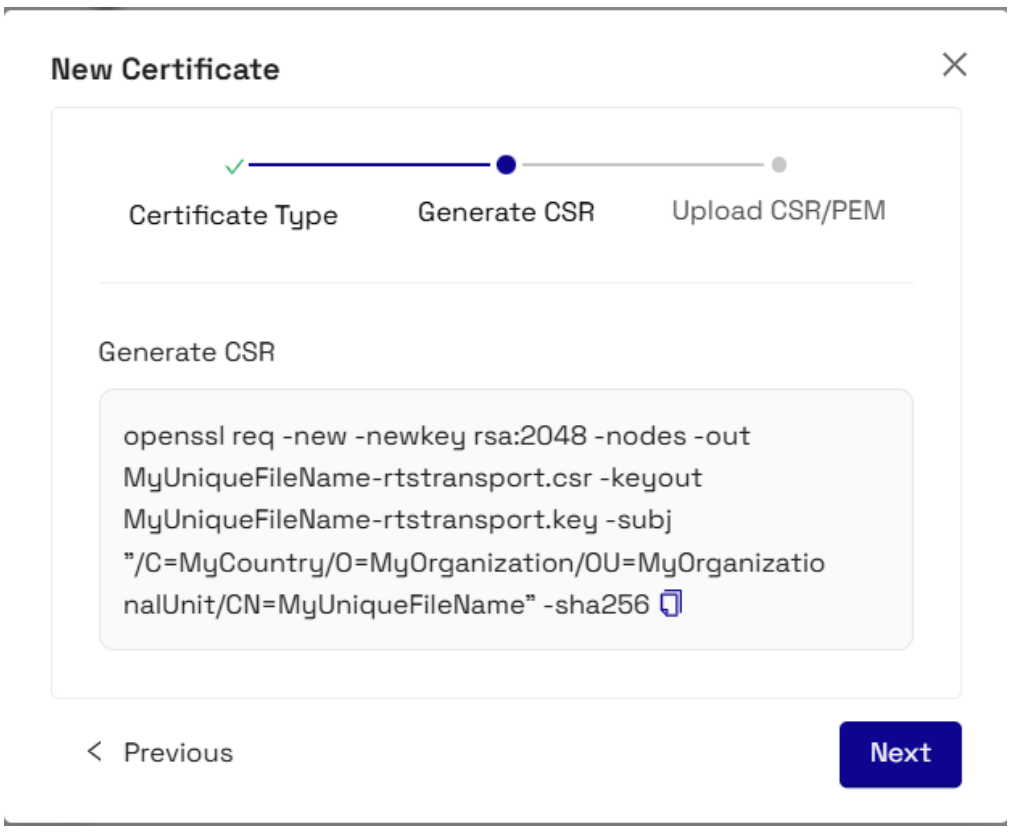
1. Select **Transport** from the **Certificate Type** dropdown menu, and then select **Next**.
2. Copy the OpenSSL command from the clipboard and run the command on the machine that you have installed OpenSSL on. This command creates a private key and a Certificate Signing Request (CSR). This command is unique to generate a Transport Certificate with the unique `-rtstransport.csr` and `-rtstransport.key` values.
3. You must store the private key securely. Once you have successfully created the private key (.key) and CSR (.csr) files, select **Next**.

   Example:

   ```
   openssl req -new -newkey rsa:2048 -nodes -out MyUniqueFileName-rtstransport.csr -keyout MyUniqueFileName-rtstransport.key -subj "/C=MyCountry/O=MyOrganization/OU=MyOrganizationalUnit/CN=MyUniqueFileName" -sha256
   ```

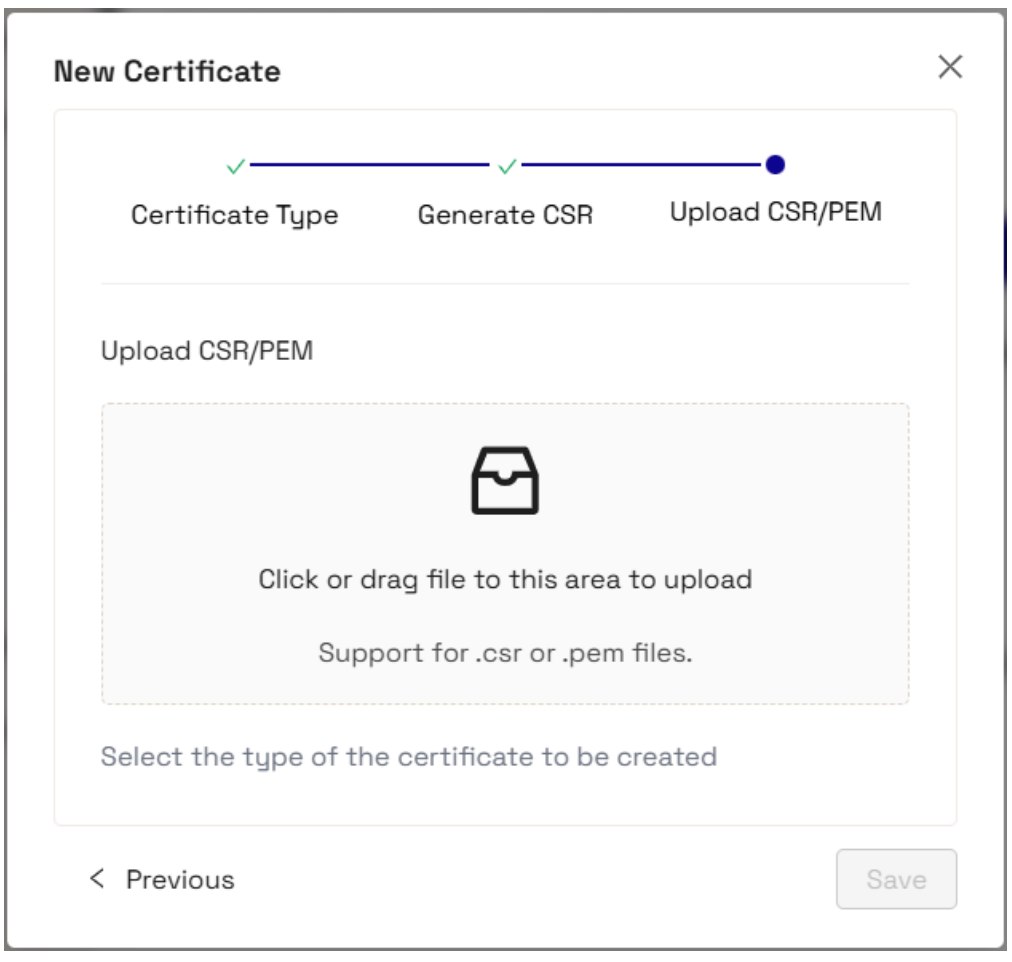
   Note: The following example contains placeholder values, but the command that you copy contains your unique file name for .csr (-out)

and .key (-keyout), and organisation information in the certificate subject (-subj).



Step 4: Upload the Certificate Signing Request for your application

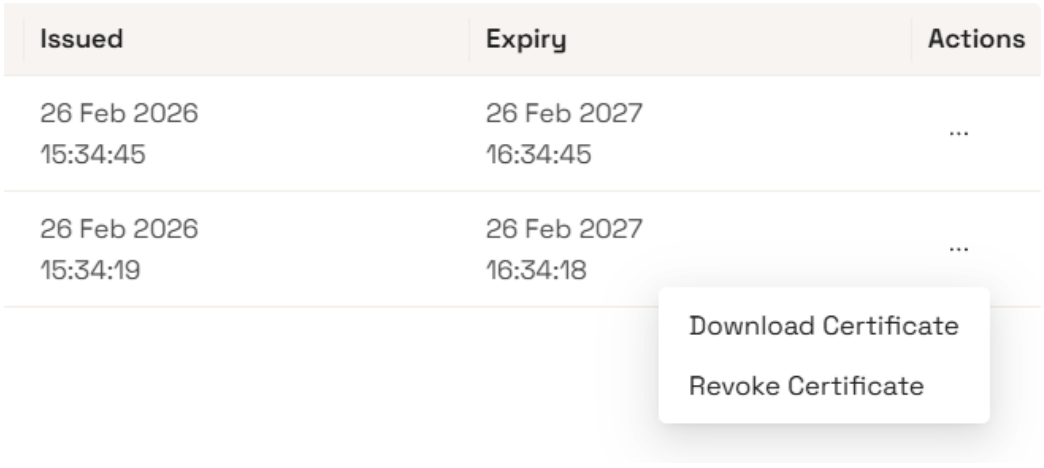
1. Select the **Upload CSR** button and select your Certificate Signing Request file. This uploads your Certificate Signing request onto the Thredd platform, so that it can use the file to request the Transport Certificate from the Thredd Certificate Authority (CA).
2. Select **Save**.



If this is successful, Thredd's system adds the newly-issued Transport Certificate to your application. You can verify this in the **App Certificates** screen of the **Certificates** tab for your application.

Step 5: Download the certificate for your application

Next, download the certificate from the **App Certificates** screen. Select the **Actions** menu (three dots) next to the TRANSPORT certificate, and select **Download Certificate**.



Step 6: Configure your Client to use the Transport Certificate in your requests

You must configure your Client to ensure that it includes the Transport Certificate in all calls that it makes to the Thredd platform. To understand how to do this in your specific case, refer to your Client library documentation.

The domain for mTLS connections is: api.thredd.com

Next steps

Once you have obtained the necessary credentials for your Client application, you can begin setting up your connection to the API Hub using the latest Postman Collection. See Using the Postman Collection to call REST APIs over mTLS.

Using the Postman Collection to call REST APIs over mTLS

Follow these steps to configure Postman with your client credentials so that you can obtain an access token. You must include an access token in all requests to Thredd's REST APIs.

You must make sure that you request and provide the Client Secret from Thredd to access Thredd's REST APIs. The Client Secret is selected as the Client Authentication method when setting up your organisation in Thredd Portal.

Additionally, because you are using an mTLS connection, your application must also present a Transport Certificate, which an Admin user can obtain from Thredd Portal.

Prerequisites

You must have the following before setting up the IAM Postman Collection:

- Administrator access to Thredd Portal
- Postman
- The Thredd IAM Postman Collection downloaded from here
- The Thredd IAM Postman Environment file from here
- OpenSSL installed on your machine

The set up for using mTLS with our Postman Collection differs slightly from the TLS setup. Before starting, ensure you have installed OpenSSL on your machine as this is required to generate a Transport certificate.

To set up Postman for mTLS, see the following steps.

Step One: Create Application and Auth Client in Thredd Portal

You can create an application and auth client in Thredd Portal. Successfully creating these produces a Client Secret and Client ID that you can enter into the Postman Collection.

Details on how to create an Application and associated Auth Client can be found in the Applications page of the Thredd Portal guide.

Step Two: Create Transport Certificate in Thredd Portal

To use mTLS for the Postman Collection you must first create a Transport Certificate, which is done in Thredd Portal.

To create a Transport Certificate in Thredd Portal:

1. Open the System Admin menu and select Applications.
2. Click Actions for the application you want to create a certificate for and click Review Application.
3. Open the Certificates tab and click New Certificate. The New Certificate window opens.
4. Select Transport in the Certificate Type field and click Next.
5. Copy the generated CSR.
6. Open a command prompt on your device.
7. Paste the CSR into the command prompt. Two files are created - a CSR file and a KEY file.
8. Return to Thredd Portal and click Next.
9. Click on the window and navigate to and select the CSR file created in the previous step.
10. Click Save. The window closes and the Certificates tab updates with the new Transport certificate.
11. Click Actions on the certificate and click Download Certificate.

Step Three: Add Transport Certificate and Key to Postman

To add your certificate and key to Postman:

1. Click File from the menu, then select Settings.
2. Click Certificates.
3. Click Add Certificate.
4. Enter uat-api.thredd.com in the Host field.
5. Click Select File under the CRT file heading. A window opens.
6. Navigate to the where you saved your Transport certificate and select the PEM file. Click Open. The window closes and the certificate is added to Postman.
7. Click Select File under the KEY file heading. A window opens.
8. Navigate to the where you saved your Transport key and select the KEY file. Click Open. The window closes and the key is added to Postman.
9. Click Add.

The Transport Certificate is now added to Postman.

Note: If the certificate is not included in Postman, the API requests still work but they are downgraded to a TLS connection. You can confirm that your connection is mTLS by viewing the Postman console and ensuring there is a client_certificate section included. This indicates the request has used a mTLS connection.

Step Four: Import Postman Collection into Postman

You can download the Postman Collection from the API Explorer page. Ensure you download the Thredd IAM API Collection.

When you have successfully downloaded the collection, open Postman and import the collection.

1. Click File from the menu, then select Import.
2. Drag the downloaded collection into the window.

The collection displays and opens in Postman.

Step Five: Import Postman Environment File

The Postman Environment file includes the variables for the Postman Collection to work, and is where you enter your Client Secret and Client ID into the collection. You can download the Postman Environment from the API Explorer page.

To import the environment file:

1. Click File from the menu, then select Import.
2. Drag the downloaded file into the window.

The environment variables are added to Postman.

Step Six: Enter Client Secret and Client ID into Postman

Note: Before starting this step, ensure you have your applications Client Secret and Client ID ready.

To add the Client Secret and Client ID to the Collection:

1. Use the dropdown in the top-right of Postman to select the UAT Thredd environment.
2. Click the Variables button in the top-right corner of Postman. All variables for the environment display.
3. Enter your Client Secret from Thredd Portal into the client_secret field.
4. Enter your Client ID from Thredd Portal into the client_id field.

Step Seven: Generate Authentication Token

Before you can use the API endpoints, you must create an authentication token. To create the token:

1. Open the Authorisation folder in the Collection and select the Get Access Token endpoint.
2. Click Send.

An authentication token is created and automatically added to your variables. This token lasts for 60 minutes. When the token expires you will need to generate a new token to continue using the APIs.

Step Eight: Check your settings and finish setting up

Make sure that you have configured the following settings for your API calls:

- You have set the correct Base URLs for the API Hub.
- You have obtained an access token.
- Including the access token in the authorisation header.
- Adding an X-Region header – this is mandatory and determines the environment that you want to connect to; select one of the following options:
    - Use `0` for the Prd 0 (Production 0) environment
    - Use `1` for the Prd 1 (Production 1) environment
    - Use `2` for the Prd 2 (Production 2) environment

Once you have completed these steps, you are ready to make your first API call. For more information, see the Thredd API Hub.

Note: Thredd previously used environment-specific production core API URLs, such as `https://coreapi-prdX.threddpay.net`. Thredd now uses a single production core API Hub endpoint: `https://api.thredd.com/core/`, with the `X-Region` header used to specify the target environment. Ensure you use the correct production core API Hub URL and `X-Region` value for your environment. For UAT, use `https://uat-api.thredd.com/core/` instead.

Preparing your Production environment

Once an Admin has verified that your testing in the UAT environment is satisfactory, you can prepare for migrating to a Production environment. You will need to repeat all of these steps for Production, such as requesting new certificates or access tokens, and ensuring that environment-specification configurations are correct. See Updating Postman.

Using the Postman Collection to call REST APIs over TLS

Follow these steps to configure Postman with your client credentials so that you can obtain an access token. You must include an access token in all requests to Thredd's REST APIs.

You must make sure that you request and provide the Client Secret from Thredd to access Thredd's REST APIs. The Client Secret is selected as the Client Authentication method when setting up your organisation in Thredd Portal.

Prerequisites

You must have the following before setting up the IAM Postman Collection:

- Administrator access to Thredd Portal
- Postman
- The Thredd IAM Postman Collection downloaded from here
- The Thredd IAM Postman Environment file from here

To set up Postman for TLS, see the following steps.

Step One: Create Application and Auth Client in Thredd Portal

You can create an application and auth client in Thredd Portal. Successfully creating these produces a Client Secret and Client ID that you can enter into the Postman Collection.

Details on how to create an Application and associated Auth Client can be found in the Applications page of the Thredd Portal guide.

Step Two: Import Postman Collection into Postman

You can download the Postman Collection from the API Explorer page. Ensure you download the Thredd IAM API Collection.

When you have successfully downloaded the collection, open Postman and import the collection.

1. Click File from the menu, then select Import.
2. Drag the downloaded collection into the window.

The collection displays and opens in Postman.

Step Three: Import Postman Environment File

The Postman Environment file includes the variables for the Postman Collection to work, and is where you enter your Client Secret and Client ID into the collection. You can download the Postman Environment from the API Explorer page.

To import the environment file:

1. Click File from the menu, then select Import.
2. Drag the downloaded file into the window.

The environment variables are added to Postman.

Step Four: Enter Client Secret and Client ID into Postman

Note: Before starting this step, ensure you have your applications Client Secret and Client ID ready.

To add the Client Secret and Client ID to the Collection:

1. Use the dropdown in the top-right of Postman to select the UAT Thredd environment.
2. Click the Variables button in the top-right corner of Postman. All variables for the environment display.
3. Enter your Client Secret from Thredd Portal into the client_secret field.
4. Enter your Client ID from Thredd Portal into the client_id field.

Step Five: Generate Authentication Token

Before you can use the API endpoints, you must create an authentication token. To create the token:

1. Open the Authorisation folder in the Collection and select the Get Access Token endpoint.
2. Click Send.

An authentication token is created and automatically added to your variables. This token lasts for 60 minutes. When the token expires you will need to generate a new token to continue using the APIs.

Step Six: Check your settings and finish setting up

Make sure that you have configured the following settings for your API calls:

- You have set the correct Base URLs for the API Hub.
- You have obtained an access token.
- Including the access token in the authorisation header.
- Adding an X-Region header – this is mandatory and determines the environment that you want to connect to; select one of the following options:
    - Use `0` for the Prd 0 (Production 0) environment
    - Use `1` for the Prd 1 (Production 1) environment
    - Use `2` for the Prd 2 (Production 2) environment

When you have completed these steps, you are ready to make your first API call. For more information, see the Thredd API Hub.

Note: Thredd previously used environment-specific production core API URLs, such as `https://coreapi-prdX.threddpay.net`. Thredd now uses a single production core API Hub endpoint: `https://api.thredd.com/core/`, with the `X-Region` header used to specify the target environment. Ensure you use the correct production core API Hub URL and `X-Region` value for your environment. For UAT, use `https://uat-api.thredd.com/core/` instead.

Preparing your Production environment

Once an Admin has verified that your testing in the UAT environment is satisfactory, you can prepare for migrating to a Production environment. You will need to repeat all of these steps for Production, such as requesting new certificates or access tokens, and ensuring that environment-specification configurations are correct. See Updating Postman.

Section 5: Migrating EHI connectivity

Learn how to configure the External Host Interface (EHI) application for the Thredd Secure Framework .

Topics covered in this section:

- Migrating to Thredd-issued certificates for EHI with mTLS

Migrating to Thredd-issued certificates for EHI with mTLS

Thredd has enhanced its Certificate Authority service for clients who want to optionally use an EHIm setup, whereby your organisation uses the External Host Interface (EHI) application to communicate with Thredd using mTLS.

An EHI with mTLS setup (EHIm) comprises an integration with an external payments network that itself is authenticated using Mutual TLS (mTLS). mTLS relies on a system of digitally-signed certificates issued by a trusted third party called a Certificate Authority (CA). The CA proves the authenticity of the public key and the identity of the Server presenting the key.

In the EHIm flow, your application is the *Server* and Thredd is the *Client*. Both parties must prove their identities to each other for authentication by presenting their respective certificates to each other, and verifying this during the mTLS handshake.

If you are currently validating certificates from Raidiam Connect, you must change to validate client certificates from Thredd.

Note: Thredd will only migrate clients that are currently using mTLS with Raidiam certificates to mTLS using Thredd certificates.

Who should read this guide

This guidance applies only to existing clients of Thredd who have an existing EHIm setup that is validating certificates from Raidiam Connect.

Note: If you are setting up EHI with mTLS for the first time, see *Setting up EHI for mTLS connections* in the Connecting to Thredd Guide.

Why you need to switch to using Thredd-issued certificates

Thredd identity and access platform supports industry standard protocols for authentication, Single Sign-On (SSO), secure role-based access control, token and certificate lifecycle management, and secure network and API-level connectivity. Thredd's scalable security architecture roadmap is aligned with emerging standards and regulatory requirements.

In the context of this work, Thredd has enhanced its Certificate Authority service for clients who want to optionally use mTLS (Mutual Transport Layer Security) for the External Host Interface (EHI) application. As a result, Thredd is phasing out support for Raidiam certificates.

What is mTLS?

Mutual Transport Layer Security (Mutual TLS or mTLS) is an enhanced security protocol that ensures that both the Client and the Server verify each other's identity before establishing a secure communication channel. This is a digital handshake where both parties must provide their official identity. To learn more about mTLS, see Mutual TLS explained.

MTLS in EHI

In the EHIm (EHI + mTLS) communication flow, your application is the *Server* and Thredd is the *Client*. Both parties must prove their identities to each other for authentication by presenting their respective certificates to each other, and verifying this during the mTLS handshake. Once both certificates are verified, the Server grants access. The Client and Server then exchange information over the secure connection.

In order to enable this, you need to install the Thredd EHIm Trust Chain on your EHI Server. The EHI Server must validate the Client Certificates that Thredd sends to you. This ensures that a Chain of Trust is established.

A Trust Chain is a collection of trusted root certificates that are installed on the Server, which includes the root anchor (top level) and intermediate Certificate Authority (CA) certificates. An end entity uses these to establish a Chain of Trust, by checking the certificate chain, and validates the Client Certificate. The Server uses the Certificate Chain to prove the legitimacy of a Client Certificate by tracing it back to a single Root CA.

Considerations before you start

This migration process comprises several steps and requires you to change the Trust Chain that your Server is validating Client Certificates against. It also requires Thredd to update the EHI Client Certificate once you have informed Thredd that you are ready to support the new Trust Chain.

You should prepare for this by reading this guide, so that you can avoid any downtime and ensure a smooth transition during the Client Certificate Migration.

First, you need to configure your systems to use and trust the new certificate. Converting the Raidiam Trust Chain to the Thredd Trust Chain requires a phased approach:

- First phase: Download and install the Thredd Trust Chain, while retaining trust for Raidiam certificates. During this phase, your server trusts certificates both from Raidiam and certificates from Thredd.
- Transitional phase: Inform Thredd and confirm that you are ready for Thredd to switch the EHI Client Certificate.
- Last phase: Once Thredd confirms that the migration is complete, you must remove the Raidiam Trust Chain from your Trust Chain file.

When you are ready, follow this guide to migrate to using Thredd's new EHI Trust Chain.

Migration steps and overview

This guidance helps you to complete the following steps:

1. Download and install Thredd's new EHI Trust Chain.
2. Ensure that your mTLS termination point has access to Thredd's Online Certificate Status Protocol (OCSP) responder.
3. Inform Thredd to switch the EHI Client Certificate.
4. Remove the Raidiam Trust Chain from your Trust Chain file, and provide your EHI endpoint to Thredd.

Prerequisites

- You must have the Admin role for your organisation in Thredd Portal.
- You must have access to Thredd Portal.

Step 1: Download and install Thredd's new EHI Trust Chain

You must install the EHI Trust Chain on your Server and configure your systems to ensure that your Server presents this certificate to the Client (Thredd) during the mTLS handshake.

During this phase, your server trusts both certificates issued by Raidiam and certificates issued by Thredd.

Instructions for storing certificates can differ between server vendors. Refer to your vendor's server documentation for a guide.

To download and install the Trust Chain:

1. Log in to Thredd Portal.
2. Navigate to and select **System Admin**, select **Organisation**, and then select **EHI Configuration**.
3. Download the EHI Trust Chain.
4. Append the Thredd EHI Trust Chain to the existing Trust Chain. This is because your mTLS termination point (your load balancer, proxy, or server) settings are currently using a Raidiam Trust Chain file. Copy the contents of the Thredd EHI Trust Chain and append it to your existing file.
5. Configure your mTLS termination point (your load balancer, proxy, or server) settings to use the Thredd EHI Trust Chain. Point it to or copy the contents of the Thredd EHI Trust Chain file.

Before you proceed, you must ensure that your mTLS termination point is validating against the updated file, in which the contents include Thredd's EHI Trust Chain.

Refer to your server's technical documentation for information about how to complete this task.

Step 2: Ensure that your mTLS termination point has access to Thredd's Online Certificate Status Protocol (OCSP) Responder

Thredd recommends that you configure your mTLS termination point (load balancer, proxy, or server) to carry out certificate status checks. For example, to check if the certificate has been revoked.

This is optional. However, if you want to ensure that it can do this, your mTLS termination point must have access Thredd's Online Certificate Status Protocol (OCSP) Responder. The OCSP Responder provides both parties, such as your organisation and Thredd, with the ability to verify the validity of certificates issued within Thredd's private PKI hierarchy, and in real time.

Check the settings for your mTLS termination point:

1. Make sure that your mTLS termination point can access Thredd's OCSP Responder at: https://ocsp.threddid.com
2. Verify and change the settings if necessary. For example, by adding the OCSP Responder to the allowlist.
3. Refer to your server's documentation for information about how to check and change your mTLS termination point settings.

Step 3: Inform Thredd that you are ready to switch the EHI client certificate

You are now in a position where your server is trusting both Raidiam certificates and Thredd certificates. Inform Thredd that you have installed the new Trust Chain on your server, and confirm that you are ready for Thredd to switch the Client Certificate.

Thredd will conduct a test to ensure that the newly-issued certificate is accepted by your server. Thredd will inform you of the result of this test once it has completed.

If successful, Thredd requests your approval to update the Client Certificate on the EHI server. Thredd will notify you when the Client Certificate Migration is complete.

Step 4: Remove the Raidiam Trust Chain from your Trust Chain file

Once Thredd has confirmed that the Client Certificate Migration is complete, you can safely delete the Raidiam Trust Chain from your Trust Chain file. This removes your server's trust for Raidiam certificates.

Refer to your server's technical documentation for information about how to complete this task.

Provide the EHI endpoint to Thredd.

Adding Security Controls

You can choose to implement additional optional controls to enhance mTLS security. These include the:

- **IP Address Allowlist** – Thredd EHI messages are sent from a firewall with a fixed IP address. Optionally, you can add this IP address to your allowlist.
- **Certificate Pinning** – the mutual communication, between your Server and Thredd (the Client), is secured by Thredd's Transport Certificate. Your Server only trusts the Transport Certificate if it the Thredd CA has issued it. However, you can choose to also implement Certificate Pinning. Certificate Pinning blocks attempted requests made with the incorrect certificates. For more information, see Certificate and Public Key Pinning | OWASP Foundation.

General FAQs

The following detailed list of FAQs focuses on the Thredd Secure Framework and its components, and related topics. These components include Thredd Portal and Thredd CA.

Overview

What is the Secure Framework?

The Thredd Secure Framework is a combination of components that enable secure access to Thredd's resources using a common identity store. These include Thredd Portal, Thredd CA, and mTLS termination (for those clients using mTLS). It ensures secure communication and access control through features like mTLS, OAuth, and certificate-based authentication. For more information, see Thredd Secure Framework.

What is mTLS and how is it used in the Secure Framework?

mTLS (Mutual Transport Layer Security) is a security protocol used to establish trust between clients and servers.

- Transport Certificates issued by Thredd CA are used to establish secure connections.
- mTLS Termination requires on-premise infrastructure to establish Trust Chains to ensure certificates originate from legitimate sources.
- It is used in the External Host Interface (EHI) setup, and works in the background for the SOAP and REST API setups.

For more information about Mutual TLS, see Mutual TLS explained.

What is the role of Cloudentity in the Secure Framework?

Thredd previously used Cloudentity as a Software as a Service (SaaS) capability to act as an Identity Provider (IdP), OAuth OpenID Provider (OP), and Policy Decision Point (PDP). However, Thredd has enhanced its Secure Framework to allow clients to manage access to Thredd services via Thredd Portal. As a result, Thredd has phased out use of Cloudentity within the Thredd Secure Framework. For more information, see Thredd Secure Framework.

Can I still use or request a VPN setup?

No, Thredd does not support new VPN setups. You must connect securely to Thredd using the Secure Framework.

Certificates

What is Thredd CA, and what certificates does it provide?

Thredd CA is Thredd's Certificate Authority responsible for creating and managing certificates. Users with an Admin role can request and revoke certificates from within Thredd Portal. It provides:

- **Transport Certificates:** For secure connections between resources.
- **Root and Issuing Certificates:** Bundled in the EHI Trust Chain, for verifying Transport Certificates in EHI setups that use mTLS.

What certificates are required for different Thredd applications?

The certificates required for various Thredd applications are as follows:

- **REST API:** Requires Transport Certificates for mTLS connections.
- **SOAP Web Services:** Requires only Transport Certificates.
- **External Host Interface (EHI):** Requires Root and Issuing Certificates (available in Thredd Trust Chain).
- **Thredd Portal:** Pre-installed Transport Certificates.
- **Smart Client:** Bundled Transport Certificates in the installer.

For more information, see the following sections that are related to requesting certificates in the Thredd Identity and Access Management Migration Guide:

- Creating an Application and OAuth Client
- Requesting certificates for applications using mTLS
- Migrating to Thredd-issued certificates for EHI with mTLS

Single Sign-On

How does SSO work in the Secure Framework?

SSO capabilities are provided through Thredd Portal, allowing federated authentication. This enables users to authenticate once and gain access to multiple Thredd resources instead of logging in separately for each service. You can also configure your own IdP provider for SSO.

See the following section within the Thredd Identity and Access Management Migration Guide: Setting up Single Sign-On access for users

External Host Interface

How do I migrate my EHIm (mTLS) setup?

You need a Server Certificate and the EHI Trust Chain to enable mTLS.

- The Server Certificate is a certificate that you have obtained from a trusted Certificate Authority (CA) vendor, such as Verizon, Digicert, or Amazon Web Services.
- The EHI Trust Chain is available from Thredd via Thredd Portal.

In the Thredd Secure Framework, you will no longer use certificates issued by Raidiam Connect, and must update your existing EHIm configuration.

For more information, see Migrating to Thredd-issued certificates for EHI with mTLS.

Postman Setup

How do I use Postman to test the Secure Framework?

Prerequisites

- Install Postman on your system.
- Create an application in Thredd Portal.

Main steps

1. Ensure that you have the necessary certificates generated and signed by the Thredd Certificate Authority (CA).
2. Add variables from Thredd CA to the variables section in the Postman collection. You can obtain this information from your application's details in Thredd Portal.
3. If using mTLS, add certificates for mTLS and repeat this process for all required hosts.
4. Set the variables in the **Variables** tab.
5. Check your settings and finish setting up.
6. Test making an API call using the Postman Collection.

For a full step-by-step guidance, see:

- Mutual TLS: Using the Postman Collection to call REST APIs over mTLS

More information

Where can I find more information?

For more details, refer to the following resources:

- Connecting to the REST APIs – for connecting to Thredd API Hub and REST APIs over a secure connection
- API Hub website – API Reference and guides for Thredd's REST APIs
- Thredd Portal: Card and Transaction Management Guide
- External Interface (EHI) Guide (JSON)
- External Interface (EHI) Guide (XML)

Glossary

This page provides a list of glossary terms used in this guide.

C

Card Scheme (Network)

Card network, such as Discover, MasterCard, or Visa, responsible for managing transactions over the network and for arbitration of any disputes.

Certificate Authority

A Certificate Authority is an entity that validates the identities of entities (such as individuals, organizations, or websites) and binds them to cryptographic key pairs through the issuance of digital certificates.

Confidential Client

A client that can maintain the confidentiality of their credentials (e.g., client implemented on a secure server with restricted access to the client credentials), or capable of secure client authentication using other means. For more details, refer to RFC 6749 for the OAuth 2.0 Authorization Framework.

E

External Host Interface (EHI)

The external system to which Thredd sends real-time transaction-related data. The Program Manager uses their external host system to hold details of the balance on the cards in their programme and perform transaction-related services, such as payment authorisation, transaction matching and reconciliation.

I

Identity Provider (IDP)

An IDP is a system entity that creates, maintains, and manages identity information for principals and also provides authentication services for relying on applications within a federation or distributed network.

M

Mutual Transport Layer Security (mTLS)

mTLS is a method for mutual authentication that ensures the parties at each end of a network connection are who they claim to be by verifying that they both have the correct private key. The information within their respective TLS certificates provides additional verification.

O

OAuth OpenID Provider

An OAuth OpenID Provider (OP) is an entity that has implemented the OpenID Connect and OAuth 2.0 protocols. OPs can also be referred to by the role it plays, such as: a security token service, an identity provider (IDP), or an authorization server.

P

Program Manager

A customer who manages a card program. The program manager can create branded cards, load funds and provide other card or banking services to their end customers.

S

Scope

The permissions related to the resources your application is registered to access. Each scope relates to specific endpoints. For example, the cards.encrypted scope gives permission to use the Get Encrypted Data endpoint.

Secure Connectivity Framework

The Secure Connectivity Framework is an umbrella set of rules and standards for identity management, verification, and assurance within a sector. The framework establishes common principles, definitions, and Open Standards for data sharing to create the foundations of a trusted data-sharing ecosystem

Smart Client

Smart Client is a user interface for programme managers to manage their account on the Thredd Platform. Smart Client is installed as a desktop application.

T

Thredd CA

Thredd CA (Certificate Authority) acts as the Certificate Authority for issuing certificates. You can create applications for your organisation, as well as request Transport and Signing certificates.

Transport Certificate

A Transport Certificate (or TLS Certificate) is a data file that contains important information for verifying a server's or device's identity, including the public key, a statement of who issued the certificate (TLS certificates are issued by a Certificate Authority), and the certificate's expiration date.

Document History

This section provides details of what has changed since the previous document release.

Version	Date	Reason	Revised by
1.0	05/10/2026	Updated the JSON Web Key Set (JWKS) endpoint instructions in the Webhook integration section to distinguish between Production and UAT environments.	GF
	16/09/2026	Updated Using the Postman Collection to call REST APIs over mTLS and Using the Postman Collection to call REST APIs over TLS to cover set up Postman for Thredd Identity and Access Management (IAM) users.	JB
	16/09/2026	Updated the Webhook integration (sometimes referred to as the Thredd Event Delivery System or EDS) section.	GF
	24/08/2026	First version.	JB

Contact Us

Please contact us if you have queries relating to this document. Our contact details are provided below.

Thredd Ltd.

Support Email: occ@thredd.com

Telephone: +44 (0) 203 740 9682

Our Head Office

33 Kingsway

London

WC2B 6UF

UK

Technical Publications

If you want to contact our technical publications team directly, for queries or feedback related to this guide, you can email us at: docs@thredd.com.