Pivotal

PRODUCT DOCUMENTATION

Single Sign-On for PCF®

Version 1.1

User's Guide

Rev: 01

Table of Contents

Single Sign-On Overview

This topic provides an overview of the Single Sign-On service for Pivotal Cloud Foundry (PCF).

The Single Sign-On service is an all-in-one solution for securing access to applications and APIs on PCF. The Single Sign-On service provides support for native authentication, federated single sign-on, and authorization. Operators can configure native authentication and federated single sign-on, for example SAML, to verify the identities of application users. After authentication, the Single Sign-On service uses OAuth 2.0 to secure resources or APIs.

Single Sign-On

The Single Sign-On service allows users to log in through a single sign-on service and access other applications that are hosted or protected by the service. This improves security and productivity since users do not have to log in to individual applications.

Developers are responsible for selecting the authentication method for application users. They can select native authentication provided by the User Account and Authentication (UAA) or external identity providers. UAA is an open source identity server project under the Cloud Foundry (CF) foundation that provides identity based security for applications and APIs.

OAuth 2.0 Authorization

After authentication, the Single Sign-On service uses OAuth 2.0 for authorization. OAuth 2.0 is an authorization framework that delegates access to applications to access resources on behalf of a resource owner.

Developers define resources required by an application bound to a Single Sign-On (SSO) service instance and administrators grant resource permissions. See the Configure Applications topic for more details.

Product Snapshot

Current Single Sign-On for Pivotal Cloud Foundry Details

- **Version**: 1.2.1
- **Release Date**: 2016-09-20
- **Compatible Ops Manager Version(s)**: 1.8 or later
- **Compatible Elastic Runtime Version(s)**: 1.8 or later
- **AWS support?** Yes
- **Google Cloud Platform?** No
- **OpenStack support?** Yes
- **vSphere support?** Yes

Upgrading to the Latest Version

Consider the following compatibility information before upgrading Single Sign-On for Pivotal Cloud Foundry®.

Elastic Runtime Version	Supported Upgrades from SSO Versions	
	From	To
1.6.x	1.0.1-1.0.20	1.0.21
1.7.x	1.0.1-1.0.21	1.1.0
	1.1.0-1.1.2	
1.8.x	1.2.0-1.2.1	1.1.0-1.1.2
		1.2.2

Note: The Single Sign-On service tile operates in lockstep with Pivotal Elastic Runtime.

- The SSO v1.1.x tiles are compatible with PCF v1.7.x
- The SSO v1.2.x tiles are compatible with PCF v1.8.x & above

If you are upgrading from PCF 1.7 to PCF 1.8 and you are using SSO v1.1.x, you must update to a SSO v1.2.x service tile before proceeding with the upgrade.

Single Sign-On for Pivotal Cloud Foundry

Active Directory Federation Services (AD FS) Integration Guide

Azure Active Directory Integration Guide

CA Single Sign-On Integration Guide

Okta Integration Guide

PingFederate Integration Guide

- PingFederate Integration Guide
  - Configure PingFederate as an Identity Provider
  - Configure SSO Service
  - Testing
  - Troubleshooting

PingOne Cloud Integration Guide

- PingOne Cloud Integration Guide
  - Configure PingOne as an Identity Provider
  - Configure SSO Service
  - Testing
  - Troubleshooting

Additional Information

- Release Notes

Installation

This topic explains how to install Single Sign-On (SSO) for Pivotal Cloud Foundry.

Prerequisites

- Pivotal Cloud Foundry (Ops Manager and Elastic Runtime) version 1.7 or later.
- SSL Certificates.
- Application Security Groups.

Install SSO via Ops Manager

1. From Pivotal Network, select a **Single Sign-On** tile version and download the product release file.
2. From the Ops Manager Installation Dashboard, select the **Import a Product** button to upload the product file.
3. Click the plus sign icon next to the uploaded product to add this product to your staging area.
4. Click on the **Single Sign-On** tile to enter any configurations.

   Note: The Single Sign-On service tile requires a network with only one subnet. The tile does not install when configured with a network that has more than one subnet.

   Note: The SSO Identity Service Broker is deployed as a PCF application from a BOSH errand, and has no associated BOSH VMs that require selecting a corresponding network. If you are forced to select a network during installation, select the **Deployment** network, also known as the PAS or ERT network.

5. Click **Apply Changes** to install the product.

Update SSL and Load Balancer

You must update the SSL certificate for the domains listed below for each plan you create. Depending on your infrastructure and load balancer, you must also update your load balancer configuration for the following domains:

- `*.SYSTEM-DOMAIN`
- `*.APPS-DOMAIN`
- `*.login.SYSTEM-DOMAIN`
- `*.uaa.SYSTEM-DOMAIN`

Configure Application Security Groups

The Single Sign-On service requires the following network connections:

- TCP connection to load balancer(s) on port 443
- TCP and UDP connection to Domain Name Servers on port 53
- (Optional) TCP connection to your external identity provider on port 80 or 443

To enable access to the Single Sign-On service, you must ensure your Application Security Group allows access to the load balancer(s) and domain name servers that provide access to Cloud Controller and UAA. Optionally, you can configure access to your external identity provider to receive SAML metadata. For more details on how to set up application security groups, see the Application Security Groups topic.

Getting Started with Single Sign-On

This topic outlines the steps for installing and configuring the Single Sign-On service.

Install and Set Up SSO for Applications

1. Install Single Sign-On via Ops Manager.
2. Create a service plan. The Single Sign-On service is a multi-tenant service, and a service plan corresponds to a tenant. This allows an enterprise to segregate users or environments using plans. Each service plan is accessible at a tenant-specific URL in the format `https://AUTH-DOMAIN.login.SYSTEM-DOMAIN`.
3. Create a service instance. Single Sign-On service plans can provide single sign-on capabilities for applications in various spaces. A service instance lets you bind an application to a service plan.
4. Configure an identity provider. In addition to the Internal User Store, you can configure external identity providers to provide single sign-on to applications. External identity providers must support SAML 2.0.
5. Configure your applications. Single Sign-On supports both Pivotal Cloud Foundry-hosted applications as well as externally hosted applications. Your applications must be able to request an OAuth or OpenID Connect token.
6. Create resources for your applications. If your registered applications need to make external API calls, you can assign the API endpoints as resources permitted for the application. This will whitelist the endpoints for use by the application or client.

SSO User Roles

A user's role determines which parts of an SSO configuration it can manage. SSO uses the existing user roles PCF Administrator and Space Developer, as well as a SSO-specific Plan Administrator role. This chart shows the management permissions for each role.

Management access by role	PCF Administrator	Plan Administrator	Space Developer
Service plans	X		
Service instances	X	X	X
Identity providers	X	X	
Applications	X	X	X
Resources	X	X	X

Using SSO for Pivotal Cloud Foundry Components

In addition to applications, SSO supports single sign-on for components of Pivotal Cloud Foundry, including Ops Manager and Apps Manager. This allows users already managed in an external identity provider to sign into Pivotal services. Refer to the following pages for instructions on configuring SSO to enable users in an external identity store to access PCF components:

- Ops Manager, on Amazon Web Services, vSphere, or OpenStack
- Apps Manager

Manage Service Plans

This topic describes how Pivotal Cloud Foundry (PCF) Administrators manage Single Sign-On service plans.

Single Sign-On is a multi-tenant service, which enables a deployment to host multiple tenants as service plans. Each service plan can have its own administrators, applications and users. This lets enterprises segregate access by using separate plans. For example, the following tenants might require separate plans:

- Business units and geographical locations
- Employees, consumers, and partners
- Development, staging, and production instances

Administrators can create new Single Sign-On service plans at any time from the SSO dashboard.

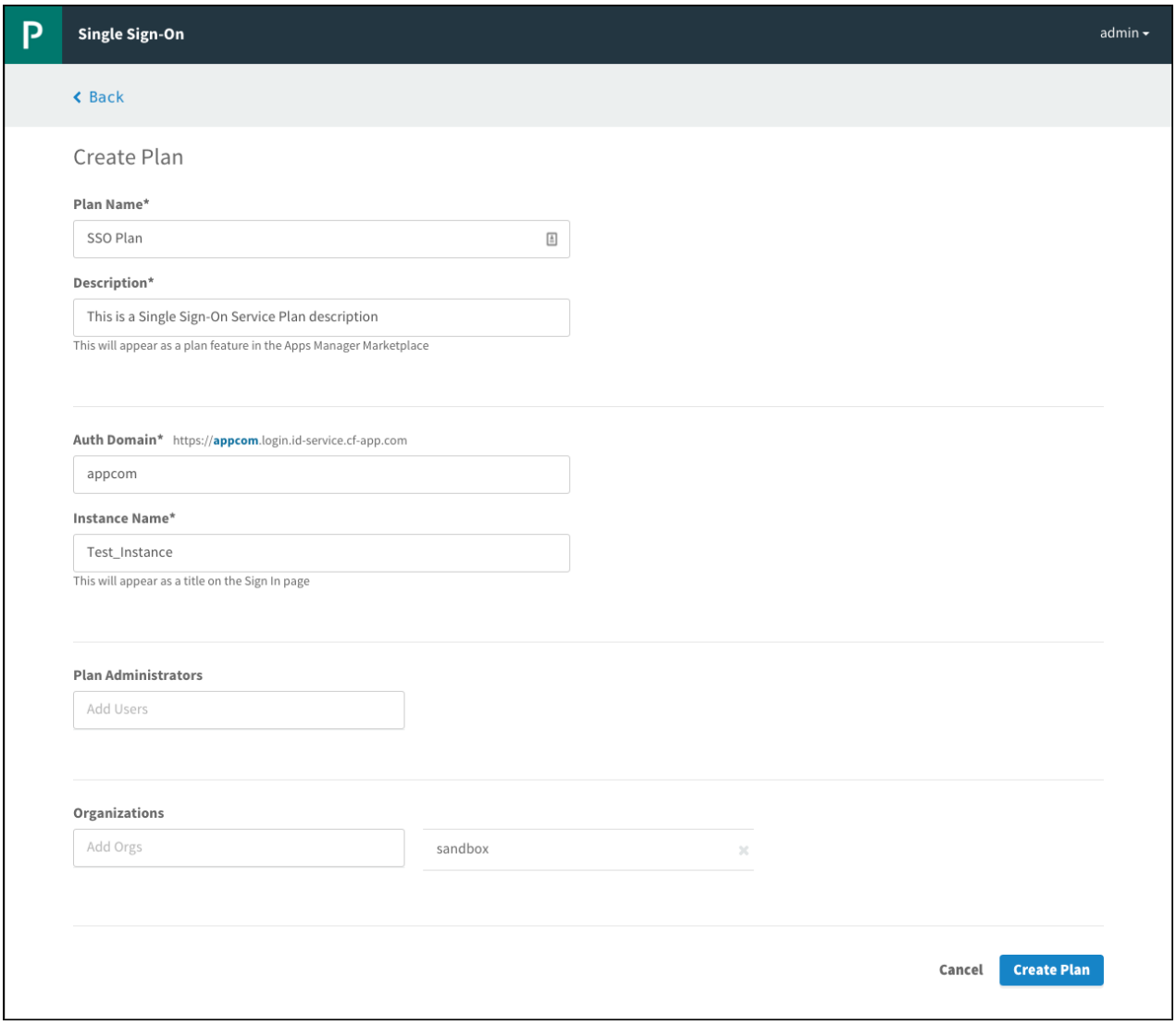
Create or Edit Service Plans

You can use the SSO dashboard to create and configure service plans at any time.

Note: You must create at least one plan for any service before your applications can use it.

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your User Account and Authentication (UAA) administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.

2. Click **New Plan** on the SSO dashboard to create a new Single Sign-On service plan.



3. Enter a **Plan Name**.
4. Enter a **Description** to appear as a plan feature in the Services Marketplace.
5. Enter an **Auth Domain** to be the URL where users authenticate to access applications covered by the service plan.
6. Enter an **Instance Name** to appear on the login page and in other user-facing content, such as email communications.
7. Add **Plan Administrators**. These users can view the plan and manage identity providers.
8. Under **Org Visibility**, select which organizations in your Pivotal Cloud Foundry deployment should have access to your Single Sign-On service plan. If you do not select any organizations, the plan will not be available for use and it will not be displayed in the Services Marketplace.
9. Click **Create Plan**. Your new plan appears in the Services Marketplace in the organizations you have selected. Users in those organizations view the plan either in Apps Manager or through the CF CLI by entering `cf marketplace` in a terminal window.

Delete Service Plans

1. Log in to the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your UAA administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.
2. Select the name of the plan you want to delete, and click **Edit Plan** in the dropdown menu.
3. Select **Delete** at the bottom of the page.
4. In the popup that appears, click **Delete Plan** to confirm that you want to delete the plan.

Note: This action cannot be undone. Deleting a Single Sign-On service plan removes from the SSO database all of the configurations, identity providers, users, application configurations and resources associated with the plan. It also deletes the associated service instances and service bindings. You must rebind any applications bound to the deleted service instances to new service instances.

Configure a Token Policy

Access tokens carry information about users and clients to servers that manage resources. Servers use access tokens to determine whether the client is authorized or not. Access tokens typically have a short-lived expiration time. *Refresh tokens* carry information necessary to retrieve a new access token after an existing access token expires. Refresh tokens typically have a longer expiration time than access tokens.

Note: The Single Sign-On service allows administrators to override the default expiry of access tokens (12 hours) and refresh tokens (30 days) by zone.

1. Log in to the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your UAA administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.
2. Select the name of the plan you would like to configure a token policy for, and click **Manage Token Policy** in the dropdown menu.
3. Enter the number of seconds for **Access Token Expiration** or select **Use System Default**.
4. Enter the number of seconds for **Refresh Token Expiration** or select **Use System Default**.
5. Click **Save**.

Manage Service Instances

This topic describes how Space Developers create an instance of a Single Sign-On service plan in their space and bind it to an application.

Create Service Instances

1. Log into Apps Manager at `https://apps.YOUR-SYSTEM-DOMAIN` as a Space Developer.
2. Navigate to the organization that the service plan is enabled for.
3. Select **Marketplace** and select the Single Sign-On service you want to create an instance of.
4. Choose your service plan and click **Select this plan**.
5. In the **Configure Instance** box, enter an **Instance Name**.
6. From the **Add to Space** dropdown menu, choose a space for the instance. This space hosts your application. The default is `development`.
7. From the **Bind to App** dropdown menu, choose an application to bind the service instance to. This option defaults to `[do not bind]`. If you do not bind the instance to an app, you can bind it at a later time.
8. Click **Add** to create the service instance.

Delete Service Instances

1. Log into Apps Manager at `https://apps.YOUR-SYSTEM-DOMAIN` as a Space Developer.
2. Navigate to the organization and space that contain the service instance you want to delete.
3. Under **Services** in the space page, find your service instance and click **Delete**.
4. Click **Delete** on the pop-up to confirm that you want to delete the service instance and service bindings.

Note: This action cannot be undone. Deleting a Single Sign-On service instance deletes the configurations on the service instance, as well as the associated service bindings. You must bind any applications bound to the deleted service instance to a new service instance.

Configure Identity Providers

This topic describes how administrators can use an internal user store or an external identity provider to manage user access to a Single Sign-On (SSO) service plan.

For each plan, SSO provides an internal user store that manages users. As an alternative to an internal user store, administrators can use an external identity provider to allow users who are externally managed to access applications.

Configure Internal User Store

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your User Account and Authentication (UAA) administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.
2. Click the plan name and select **Manage Identity Providers** from the dropdown menu.
3. Click **Internal User Store**.
4. Under **Authentication Policy**, optionally select one of the following:
   - **Disable Internal Authentication**: Select this option to prevent authentication against the internal user store. You must have at least one external identity provider configured.

     Note: The login page does not include the **Email** and **Password** fields if you select this option.

   - **Disable User Management**: Select this option to prevent all users, including administrators, from performing actions on internal users.

     Note: The login page does not include **Create Account** and **Reset Password** links if you select this option.
5. Under **Password Policy Settings**, select **Use Recommended Settings**, **Use Default Settings**, or enter custom settings in the fields below.
6. Click **Save Identity Provider**.

Add Users to the Internal User Store

You cannot add users to Service Plans from the SSO dashboard. In order to add users to the internal user store for a given Service Plan, you must use the UAA Command Line Interface (UAAC). If you do not already have the UAAC installed, run `gem install cf-uaac` in a terminal window.

The following steps describe how to use UAAC to add users to the internal user store.

Step 1: Create an Admin Client

1. Create an admin client that can manage users in the Service Plan. Include the following scopes for the client:
   - clients.admin
   - scim.read
   - scim.write
2. Record the **App ID** and **App Secret**. These are used as your client ID and client secret.

Step 2: Create Users

1. Target the auth domain of your SSO service plan. This is the URL you provided when creating a Service Plan in the SSO dashboard.

```
$ uaac target https://YOUR-AUTH-DOMAIN.login.YOUR-SYSTEM-DOMAIN
```

2. Fetch the token for the admin client created in Step 1.

```
$ uaac token client get ADMIN-CLIENT-ID
Client secret:
```

3. When prompted with `Client secret`, enter the admin client secret from Step 1.

4. Add new users by providing the user's email address, username, and password.

```
$ uaac user add --emails YOUR-USER@EMAIL.COM
User name:  YOUR-USER
Password:  ****
Verify password:  ****
user account successfully added
```

5. (Optional) You can also create groups and add users to them.

```
$ uaac group add
Group name:  YOUR-GROUP
meta
version: 0
created: 2016-02-19T23:17:17.000Z
lastmodified: 2016-02-19T23:17:17.000Z
schemas: urn:scim:schemas:core:1.0
id: 8725b5fd-8da2-4cfc-89b1-c57048f089c2
displayname: YOUR-GROUP
```

   To add a member to your new group, use the following command.

```
$ uaac member add YOUR-GROUP YOUR-USER
```

Define Password Policy for the Internal User Store

Administrators can define the password policy for SSO users that are stored in the internal user store. The internal user store password policy allows you to define and enforce password rules to manage the kind of passwords users can create.

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your UAA administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.

2. Click the plan name and select **Manage Identity Providers** from the dropdown menu.

3. Click **Internal User Store**.

4. Configure the following under the **Password Complexity** section:
   - **Min Length**: Specify the minimum password length.
   - **Uppercase**: Specify the minimum number of uppercase characters required in a password.
   - **Lowercase**: Specify the minimum number of lowercase characters required in a password.
   - **Special Characters**: Specify the minimum number of special characters required in a password.
   - **Numerals**: Specify the minimum number of numeric characters required in a password.

5. Configure the following under the **Lockout Policy** section:
   - **Failures Allowed**: Specify the number of failed login attempts allowed per hour before a user is locked out.
   - **Lockout Period**: Specify the number of seconds a user is locked out for after excessive failed login attempts.
   - **Password Expires**: Specify the number of months passwords are valid for before users needs to enter a new password.

6. Click **Save Identity Provider**.

Configure Service Provider SAML Settings

For each plan, the Single Sign-On service allows you to configure SAML settings when SAML is used for exchanging authentication and authorization data between the identity provider and the service provider. The SSO service provides the ability to sign authentication requests and require signed assertions from the external identity provider.

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your UAA administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.

2. Click the plan name and select **Manage Identity Providers** from the dropdown menu.
3. Click **Configure SAML Service Provider**.
4. Configure the following settings:
    - **Perform signed authentication requests**: The service provider signs requests sent to the external identity provider.
    - **Require signed assertions**: The service provider requires that responses from the external identity provider are signed.
5. Click **Save** to save the SAML configurations.
6. Click **Download Metadata**.

Add an External Identity Provider

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your UAA administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.
2. Click the plan name and select **Manage Identity Providers** from the dropdown menu.
3. Click **New Identity Provider**.
4. Enter an **Identity Provider Name**.
5. Enter a **Description**. This is displayed to Space Developers when selecting an identity provider for their application.
6. Enter the external identity provider metadata in one of the following ways:
    - Option 1: Provide the **Identity Provider Metadata URL** and click **Fetch Metadata**.
    - Option 2: Click **Upload Identity Provider Metadata** to upload XML metadata that you downloaded from your external identity provider.
7. Configure any **User Attributes** to propagate from the identity provider to the service provider. These attributes can include e-mail addresses, first or last names, or external groups. They are sent to applications via OpenID tokens along with any other stored user information issued by the Single Sign-On service.
    - Select a **User Scheme Attribute** from the dropdown menu.
    - Enter a **SAML Attribute Name** with the corresponding attribute from the incoming SAML assertion.
8. Configure any **Custom Attributes** that should be propagated from the identity provider to the service provider. These attributes will be sent to applications via OpenID tokens issued by the Single Sign-On service.
    - Enter a **Custom Attribute Name**.
    - Enter a **SAML Attribute Name** with the corresponding attribute from the incoming SAML assertion.
9. Click **Create Identity Provider** to save the identity provider.

Note: To configure the service provider SAML settings, such as the signing of authentication requests and incoming assertions, click on **Configure SAML Service Provider** on the Identity Providers page.

Delete an External Identity Provider

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your UAA administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.
2. Click the plan name and select **Manage Identity Providers** from the dropdown menu.
3. Click on the name of your external identity provider.
4. Click **Delete** at the bottom of the page.
5. In the popup that appears, click **Delete Identity Provider** to confirm that you want to delete the identity provider, along with all of its configurations.

Note: Deleting an external identity provider deletes all of its configurations. Users will no longer be able to authenticate using the external identity provider. This action cannot be undone.

Configure Group Whitelist for an External Identity Provider

An administrator can create groups from an external identity provider in Group Whitelist. By creating these groups, they are propagated in the ID token when a user authenticates through an external identity provider. This provides information to the application about the external groups that the user belongs to. An administrator can use these groups to assign permissions by group rather than individual users. For more details on how to create resource permission mappings, see Create or Edit Resource Permissions.

Note: The `roles` scope must be requested by the application and the external group must be listed in the Group Whitelist.

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your UAA administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.
2. Click the plan name and select **Manage Identity Providers** from the dropdown menu.
3. Click **Group Whitelist**.
4. Add a group name from your external identity provider.
5. Click **Save Group Whitelist**.

Configure Applications

This topic describes how Space Developers bind or register applications to their Single Sign-On (SSO) service instances.

If your application is hosted on Pivotal Cloud Foundry (PCF), refer to the Bind an Application Hosted on PCF section to bind the application to your SSO service instance from Apps Manager. If your application is externally hosted, refer to the Register an External Application section to register your application with your SSO service instance from the SSO dashboard.

When you bind or register an application with a SSO service instance, SSO creates an OAuth client. This OAuth client acts as an OAuth 2.0 authorization server and issues tokens.

Determine Your Application Type

Before you bind or register an application, you must know your SSO application type. Refer to the table below to determine the application type best suited for your application.

If your application authenticates end users, then your application type is `Web App`, `Native Mobile App`, or `Single-Page JavaScript App`. If your application does not authenticate end users, but rather accesses other services or APIs on its own behalf, then your application type is `Service-to-Service App`.

Application Type	SSO Application Type	OAuth Grant Type Equivalent
Web	`Web App`	`authorization code`
Native Mobile, Desktop, or Command Line	`Native Mobile App`	`password` (the resource owner's password)
Service-to-Service	`Service-to-Service App`	`client_credentials`
Single-Page JavaScript	`Single-Page JavaScript App`	`implicit`

Note: The Native Mobile App application type is intended only for highly trusted applications such as company owned and managed applications.

Preconfigure an Application Hosted on PCF

Follow the steps below to create environment variables that you can then set during a bind.

Set Application Type:

- **Option 1**: Set the grant type environment variable by performing the following steps:
  1. Log in to Apps Manager at `https://apps.YOUR-SYSTEM-DOMAIN.`
  2. Navigate to your application.
  3. Click the **Env Variables** tab.
  4. Click **Add an Env Variable**.
  5. For **Variable Name**, enter `GRANT_TYPE`.
  6. For **Value**, enter the OAuth grant type for your application type. For example, if your application is a Single-Page JavaScript App, specify `implicit`.
  7. Bind and restage your application.
- **Option 2**: Set the grant type environment variable by including the following in your application manifest. If you choose this option, you do not have to configure environment variables after deploying your app.

```
---
applications:
    - name: APPLICATION NAME
      env:
          GRANT_TYPE: OAUTH GRANT TYPE
```

Note: If you do not provide a `GRANT_TYPE`, the application type defaults to Web App.

Set Identity Provider:

Set the identity providers by including the following in your application manifest:

```
---
applications:
    - name: APPLICATION NAME
      env:
          SSO_IDENTITY_PROVIDERS: COMMA SEPARATED LIST OF IDENTITY PROVIDERS
```

Note: If you do not provide any `SSO_IDENTITY_PROVIDERS`, the internal user store will be selected by default.

Bind an Application Hosted on PCF

1. Log in to Apps Manager as a Space Developer.
2. Select the space where your application runs.
3. Under **Applications**, click the name of your application.
4. Click the **Services** tab.
5. Click **Bind a Service**.
6. Bind your application to a service to create an associated OAuth Client.
    a. Select an existing SSO service instance from the dropdown menu and click **Bind**.
    b. Create a new service instance:
        i. Click **or add from Marketplace**.
        ii. Select the **Single Sign-On** service under Services Marketplace.
        iii. Select a Service Plan, then click **Select this plan**.
        iv. Enter an **Instance Name**, select a space, select an app, then click **Add**.
7. Click **Manage** under the SSO service instance to launch the SSO dashboard.
8. Click your application.
9. Specify a value in the **App Launch URL** field that you want to set as the address of your application.
10. Upload an app icon for your application.
11. Click **Show on homepage** to display the application on the UAA or Pivotal Account home page.

    Note: If you would like application to display on the home page, you must enter an **App Launch URL** or upload an app icon.

12. Select one or more **Identity Providers** for your application. Internal User Store is the default.

    Note: When registering an externally hosted application, a Space Developer can choose from internal and external identity providers. If the Space Developer selects multiple identity providers, users must select which provider to use when they sign in. This option is available for all application types except `Service-to-Service App`.

13. If your Application Type is `Web App` or `Single-Page JavaScript App`, enter a whitelist of **Auth Redirect URIs** beneath **Redirect URIs**. The redirect query parameter specified on the OAuth request must match the URIs specified in this list. Otherwise, SSO rejects the request.
14. For the **Scopes** field, specify the permissions that the application can request on the user's behalf. This field defaults to `openid` for Web, Native Mobile, and Single-Page JavaScript Apps. This field defaults to `uaa.resource` for Service-to-Service Apps. If this application is purely for authentication purposes, then the `openid` scope is sufficient. If the application makes API calls on behalf of the end user, you must specify both the scopes enforced by the API and the scopes to be requested by the application.

Scope	Description
`openid`	Provides access to make OpenID Connect requests
`user_attributes`	Provides access to custom attributes from an external identity provider
`roles`	Provides access to external groups from an identity provider

Scope	Description
`roles`	Provides access to external groups from an identity provider
`uaa.resource`	Provides access to the check_token endpoint for service-to-service flows

Note: Under **Scopes**, you can select resources defined in any space if the application type is a `Web App`, `Native Mobile App`, or `Single-Page JavaScript App`. If the application type is a `Service-to-Service App`, you can only select resources defined within the space.

15. For **Auto-Approved Scopes**, select any scopes that the SSO service automatically approves when the app makes a request on behalf of a user. Select only scopes pertaining to applications owned and managed by your company. Do not select scopes that pertain to applications external to PCF.

16. Click **Save Config**. The **Next Steps** page appears, describing the endpoints required for application integration. Refer to the Integrate SSO with Applications section below for more details.

Register an External Application

1. Log in to Apps Manager as a Space Developer.

2. Select the space where your service instance is located.

3. Under **Services**, click **Manage** next to the SSO service instance. This launches the SSO dashboard.

4. Click **New App**.

5. Enter an **App Name**.

6. Choose an application type under **Select an Application Type**.

7. Enter an **App Launch URL** that specifies the address of your application.

8. Upload an app icon for your application.

9. Click **Show on homepage** to display the application on the UAA or Pivotal Account home page.

Note: To display the application on the home page, you must enter an **App Launch URL** or Upload an app icon.

10. Select one or more **Identity Providers** for your application. Internal User Store is the default.

Note: When registering an externally hosted application, a Space Developer can choose from internal and external identity providers. If the Space Developer selects multiple identity providers, users must select which provider to use when they sign in. This option is available for all application types except `Service-to-Service App`.

11. If your Application Type is `Web App` or `Single-Page JavaScript App`, enter a whitelist of **Auth Redirect URIs** beneath **Redirect URIs**. The redirect query parameter specified on the OAuth request must match the URIs specified in this list. Otherwise, SSO rejects the request.

12. For the **Scopes** field, specify the permissions that the application can request on the user's behalf. This field defaults to `openid` for Web, Native Mobile, and Single-Page JavaScript Apps. This field defaults to `uaa.resource` for Service-to-Service Apps. If this application is purely for authentication purposes, then the `openid` scope is sufficient. If the application makes API calls on behalf of the end user, you must specify both the scopes enforced by the API and the scopes to be requested by the application.

Scope	Description
`openid`	Provides access to make OpenID Connect requests
`user_attributes`	Provides access to custom attributes from an external identity provider
`roles`	Provides access to external groups from an identity provider
`uaa.resource`	Provides access to check_token endpoint for service-to-service flows

Note: Add the `user_attributes` scope to the client scopes to return user attributes from the ID token.

Note: Under **Scopes**, you can select resources defined in any space if the application type is a `Web App`, `Native Mobile App`, or `Single-Page JavaScript App`. If the application type is a `Service-to-Service App`, you can only select resources defined within the space.

13. For **Auto-Approved Scopes**, select any scopes that the SSO service automatically approves when the app makes a request on behalf of a user. Select

only scopes pertaining to applications owned and managed by your company. Do not select scopes that pertain to applications external to PCF.

14. Click **Create App**. The **Next Steps** page appears, describing the endpoints required for application integration. Refer to the Integrate SSO with Applications section below for more details.

Integrate SSO with Applications

Because SSO service is based on the OAuth protocol, your applications must be OAuth-aware.

Java Applications

If you are using Java, refer to the Single Sign-On Service Sample Applications. These are sample applications created using Spring Boot for all four application types. These applications use the SSO Service Connector, which auto-configures the application for OAuth. After binding the application to an SSO service instance, you must restart the application for the new SSO configuration to take effect.

Non-Java Applications

To configure non-Java applications for OAuth, supply the following properties as environment variables to your application after the SSO service bind. You can view this information on the **Next Steps** page of the SSO dashboard.

- **App ID**, also known as OAuth Client ID
- **App Secret**, also known as OAuth Client Secret
- **OAuth Authorization URL**, the endpoint for client authorization
- **OAuth Token URL**, the endpoint for token retrieval

To validate the token, you must verify the following:

1. The token is a properly signed JSON Web Token with an appropriate public key. The key can be downloaded from the **Token Verification Key** endpoint specified on the **Next Steps** page.
2. The value of `aud` in the token matches your **App ID**.
3. The value of `iss` matches `https://AUTH-DOMAIN.uaa.YOUR-SYSTEM-DOMAIN/oauth/token`.
4. The expiry time of the token, `exp`, has not passed.

Create Admin Client

You can create an admin client to perform administrative functions, such as manage identity providers, applications, users, groups, and resources in a specific zone where you create the client.

To create an admin client, complete the following steps:

1. Log in to Apps Manager.
2. Select the space where your service instance is located. This specifies the zone you manage as an admin client.
3. Under **Services**, click the **Single Sign-On** service.
4. Click **Manage** next to your SSO service instance to launch the SSO dashboard.
5. Click **New App**.
6. Enter an **App Name**.
7. Under **Select an Application Type**, select **Service-to-Service App**.
8. Click **Select Scopes** and choose what actions the admin client can perform from the following **Admin Permissions**:

Scope	Description

Scope	Description
`clients.admin`	Provides superuser access to create, modify, and delete clients
`clients.read`	Provides access to read information about clients
`clients.write`	Provides access to create and modify clients
`scim.create`	Provides access to create users
`scim.read`	Provides access to read information about users and group memberships
`scim.write`	Provides access to create, modify, and delete users and group memberships
`idps.read`	Provides access to read information about identity providers
`idps.write`	Provides access to create, modify, and delete identity providers

9. Click **Create App**.

Delete Application

Complete the procedure that corresponds with your application type.

Delete a PCF Application

To delete an application hosted on PCF, complete the following steps:

1. Log in to Apps Manager as a Space Developer.
2. Select the space where your application is located.
3. Under **Applications**, click the name of your application.
4. On the Application Page, click **Delete App**.
5. On the popup, click **Delete** to confirm that you want to delete the application and its configurations from Apps Manager and the service dashboard.

Delete an External Application

To delete an external application not hosted on PCF, complete the following steps:

1. Log in to Apps Manager as a Space Developer.
2. Select the space where your service instance is located.
3. Under **Services**, click **Manage** next to your SSO service instance to launch the SSO dashboard.
4. Click your application.
5. Click **Delete** at the bottom of the page.
6. On the popup, click **Delete App** to confirm that you want to delete the application and its configurations.

> **Note**: Deleting an externally hosted application removes the application and its configurations from the SSO dashboard. However, it still exists on your hosted platform.

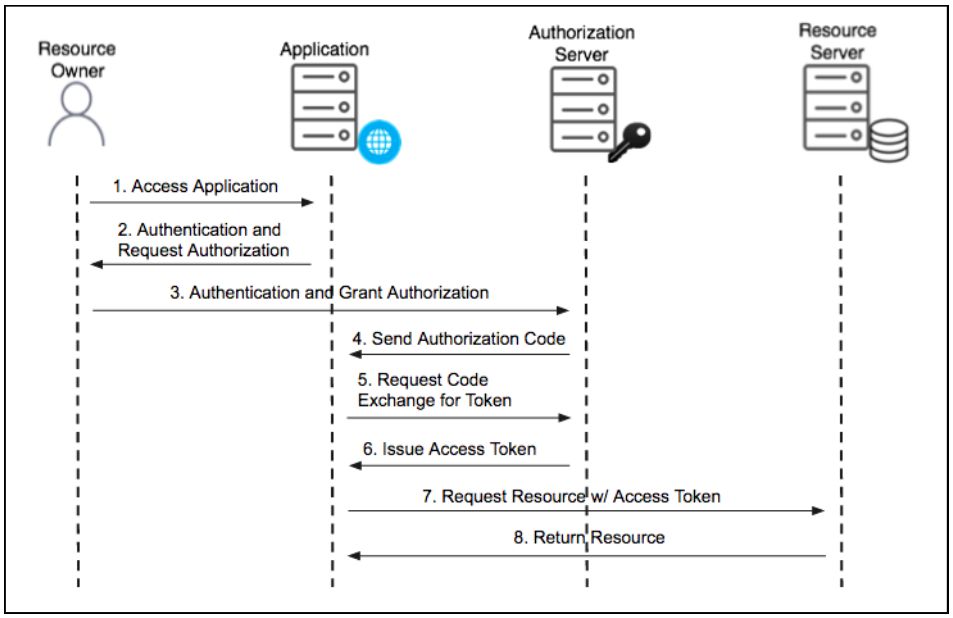
Web App

This topic describes the OAuth 2.0 Authorization Code grant type supported by Pivotal Single Sign-On (SSO). The authorization code grant type is the most commonly used grant type. This grant type is for server-side applications.

OAuth 2.0 Roles

- **Resource Owner**: A person or system capable of granting access to a protected resource.
- **Application**: A client that makes protected requests using the authorization of the resource owner.
- **Authorization Server**: The Single Sign-On server that issues access tokens to client applications after successfully authenticating the resource owner.
- **Resource Server**: The server that hosts protected resources and accepts and responds to protected resource requests using access tokens. Applications access the server through APIs.

Authorization Code Flow



1. **Access Application**: The user accesses the application and triggers authentication and authorization.
2. **Authentication and Request Authorization**: The application prompts the user for their username and password. The first time the user goes through this flow for the application, the user sees an approval page. On this page, the user can choose permissions to authorize the application to access resources on their behalf.
3. **Authentication and Grant Authorization**: The authorization server receives the authentication and authorization grant.
4. **Send Authentication Code**: After the user authorizes the application, the authorization server sends an authorization code to the application.
5. **Request Code Exchange for Token**: The application receives the authorization code and requests an access token from the authorization server. This gives the application access to the approved permissions.
6. **Issue Access Token**: The authorization server validates the authorization code and issues an access token.
7. **Request Resource w/ Access Token**: The application attempts to access the resource from the resource server by presenting the access token.
8. **Return Resource**: If the access token is valid, the resource server returns the resources that the user authorized the application to receive.

The resource server runs in PCF under a given space and organization. Developers set the permissions for the resource server API endpoints. To do this, they create resources that correspond to API endpoints secured by the Single Sign-On service. Applications can then access these resources on behalf of users.

Native Mobile App

For Native Mobile and Desktop applications, Pivotal Single Sign-On (SSO) supports the Resource Owner Password OAuth 2.0 grant type. This password grant type is for highly trusted applications where resource owners share their credentials directly with the application.

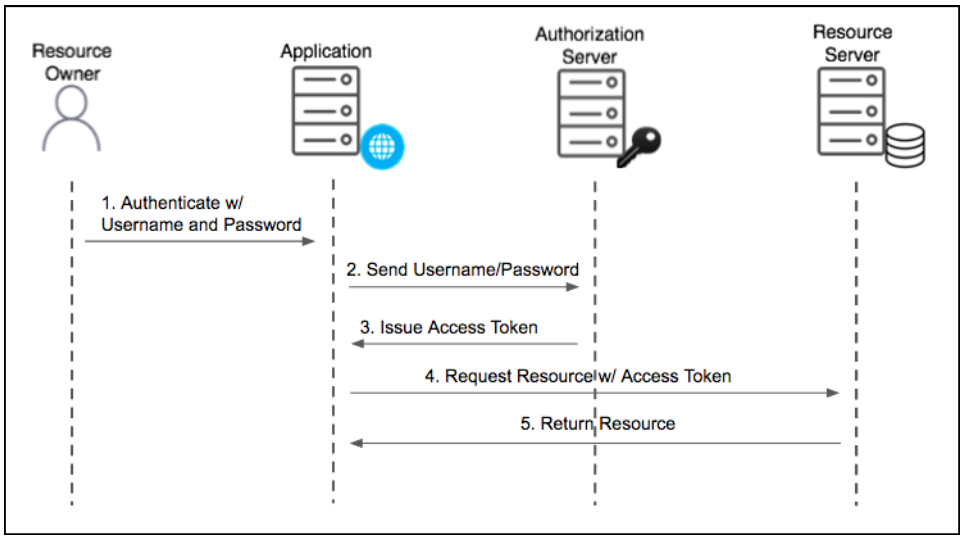
OAuth 2.0 Roles

The following roles are available in an OAuth 2.0 scenario:

- **Resource Owner**: A person or system capable of granting access to a protected resource.
- **Application**: A client that makes protected requests using the authorization of the resource owner.
- **Authorization Server**: The Single Sign-On server that issues access tokens to client applications after successfully authenticating the resource owner.
- **Resource Server**: The server that hosts protected resources and accepts and responds to protected resource requests using access tokens. Applications access the server through APIs.

Native Mobile App Flow

The following diagram shows the authentication flow used by mobile apps. In this scenario, the application is backed by a resource server and both are secured by the UAA authorization server.



1. **Authenticate w/ Username and Password**: The user authenticates with the application using their username and password.
2. **Send Username/Password**: The application sends the username and password to the authorization server for validation.
3. **Issue Access Token**: The authorization server validates the username and password and issues an access token.
4. **Request Resource w/ Access Token**: The application attempts to access the resource from the resource server by presenting the access token.
5. **Return Resource**: If the access token is valid, the resource server returns the resources that the user authorized the application to receive.

The resource server runs in PCF under a given space and organization. Developers set the permissions for the resource server API endpoints. To do this, they create resources that correspond to API endpoints secured by the Single Sign-On service. Applications can then access these resources on behalf of users.

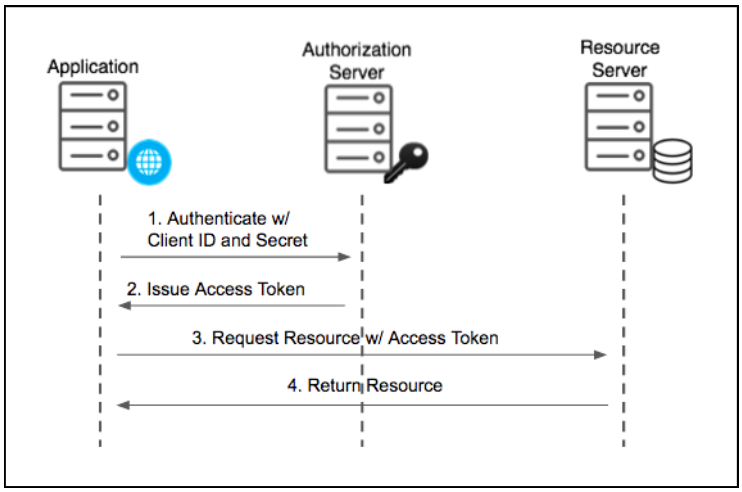
Service-to-Service App

For Service-to-Service applications, Pivotal Single Sign-On (SSO) supports the Client Credentials OAuth 2.0 grant type. The client credentials grant type is for applications that can request an access token and access resources on its own. This is often the case when there are services that call APIs without users.

OAuth 2.0 Actors

- **Application**: A client that makes protected requests using the authorization of the resource owner.
- **Authorization Server**: The Single Sign-On server that issues access tokens to client applications after successfully authenticating the resource owner.
- **Resource Server**: The server that hosts protected resources and accepts and responds to protected resource requests using access tokens. Applications access the server through APIs.

Client Credentials Flow



1. **Authenticate w/ Client ID and Secret**: The application authenticates with the authorization server using its client ID and client secret.
2. **Issue Access Token**: The authorization server validates the client ID and client secret and issues an access token.
3. **Request Resource w/ Access Token**: The application attempts to access the resource from the resource server by presenting the access token.
4. **Return Resource**: If the access token is valid, the resource server returns the resources to the application.

The resource server runs in PCF under a given space and organization. Developers set the permissions for the resource server API endpoints. To do this, they create resources that correspond to API endpoints secured by the Single Sign-On service. Administrators can create admin clients to perform automated management actions without a user. See Create Admin Client.

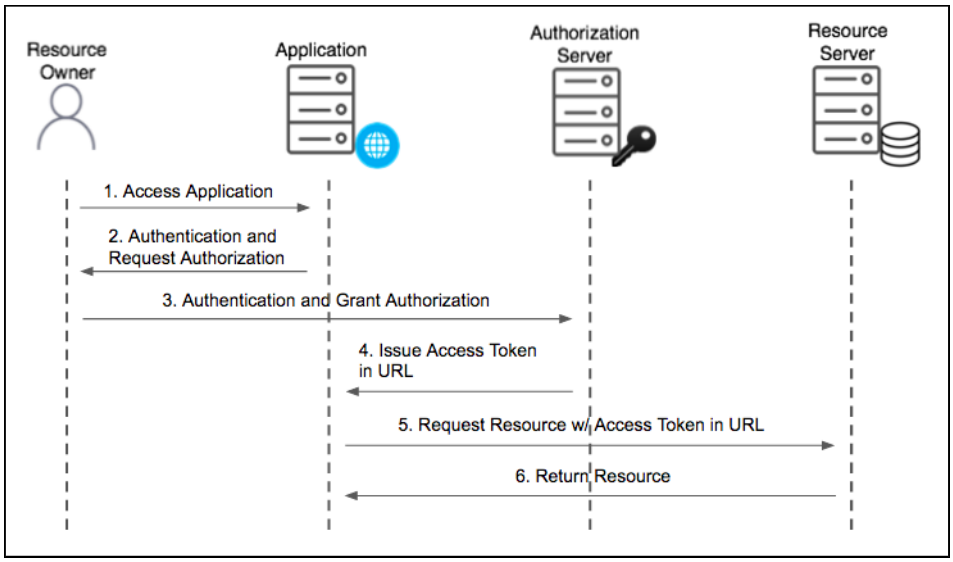
Single-Page Javascript App

This topic describes the OAuth 2.0 implicit grant type supported by Pivotal Single Sign-On (SSO). The implicit grant type is for applications with a client secret that is not guaranteed to be confidential.

OAuth 2.0 Roles

- **Resource Owner**: A person or system capable of granting access to a protected resource.
- **Application**: A client that makes protected requests using the authorization of the resource owner.
- **Authorization Server**: The Single Sign-On server that issues access tokens to client applications after successfully authenticating the resource owner.
- **Resource Server**: The server that hosts protected resources and accepts and responds to protected resource requests using access tokens. Applications access the server through APIs.

Implicit Flow



1. **Access Application**: The user accesses the application and triggers authentication and authorization.
2. **Authentication and Request Authorization**: The application prompts the user for their username and password. The first time the user goes through this flow for the application, the user sees an approval page. On this page, the user can choose permissions to authorize the application to access resources on their behalf.
3. **Authentication and Grant Authorization**: The authorization server receives the authentication and authorization grant.
4. **Issue Access Token**: The authorization server validates the authorization code and returns an access token with the redirect URL.
5. **Request Resource w/ Access Token in**: The application attempts to access the resource from the resource server by presenting the access token in the URL.
6. **Return Resource**: If the access token is valid, the resource server returns the resources that the user authorized the application to receive.

The resource server runs in PCF under a given space and organization. Developers set the permissions for the resource server API endpoints. To do this, they create resources that correspond to API endpoints secured by the Single Sign-On service. Applications can then access these resources on behalf of users.

Manage Resources

This topic describes how a Space Developer defines resources required by an application bound to a Single Sign-On (SSO) service instance, as well as how an administrator grants resource permissions.

Resources are the API endpoints that users and applications need access to retrieve information from the resource server. Since developers know what endpoints exist for their applications, they are responsible for creating resources. Afer resources are created, administrators will assign these resources to users and applications so that users can grant applications delegated access to the resources on their behalf.

Create or Edit Resources

If an application requires access to specific resources such as API endpoints, the Space Developer must define permissions for those resources in the SSO dashboard.

1. Log into Apps Manager as a Space Developer.
2. Select the space where your service instance is located.
3. Under **Services**, click **Manage** next to your SSO service instance to launch the SSO dashboard.
4. Click the **Resources** tab.
5. Click **New Resource**.
6. Enter a **Resource Name**.
7. Create **Permissions** that the OAuth client for your application needs to access from the resource server.
    a. Enter one or more **Attributes** or **Actions** for each permission.
    b. Enter a **Description** for each permission.
8. Click **Save Resource**. The administrator must create resource permissions so that users can access the resource. See the Create or Edit Resource Permissions section below for more details.

Note: Space Developers create resources within a space. Space Developers only see the resources created in the spaces they have access to and can only assign those to the applications in those spaces.

Delete Resources

1. Log into Apps Manager as a Space Developer.
2. Click the **Manage** link under the SSO service instance to launch the service dashboard.
3. Click the **Resources** tab.
4. Click your resource.
5. Click **Delete** at the bottom of the page.
6. On the popup, click **Delete Resource** to delete the resource.

Note: Deleting a resource removes it from the permission mappings and from the application. You must reconfigure the updated permissions in both areas.

Create or Edit Resource Permissions

After a Space Developer defines resources required by an application, an administrator must grant access to those resources. SSO allows administrators to map groups of users from the identity provider to the resource permissions defined by the Space Developer.

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your User Account and Authentication (UAA) administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.
2. Click the plan name and select **Manage Identity Providers** from the dropdown menu.
3. Click **Resource Permissions** for the identity provider that you want to define permissions for.
4. Click **New Permissions Mapping**.
5. Enter a **Group Name**.
6. Click **Select Permissions** to choose the permissions that users in the group should have access to.
7. Click **Save Permissions Mapping**.

Note: Groups with unsupported characters in Permission Mappings are not editable.

Delete Resource Permissions

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your User Account and Authentication (UAA) administrator credentials. You can find these credentials in your Pivotal Elastic Runtime tile in Ops Manager under the Credentials tab.
2. Click the plan name and select **Manage Identity Providers** from the dropdown menu.
3. Click **Resource Permissions** for the identity provider that you want to define permissions for.
4. Click the group name of the resource permission you want to delete.
5. Click **Delete** at the bottom of the page.
6. On the popup, click **Delete Permissions Mapping** to delete the resource.

Note: Groups with unsupported characters in Permission Mappings are not editable.

Active Directory Federation Services Integration Guide Overview

Active Directory Federation Services (AD FS) is a standards-based service that securely shares identity information between applications. This documentation describes how to configure a single sign-on partnership between AD FS as the Identity Provider (IdP) and the Single Sign-On Service (SSO) for Pivotal Cloud Foundry as the Service Provider (SP).

SSO supports service provider-initiated authentication flow and single logout. It does not support identity provider-initiated authentication flow. All SSO communication takes place over SSL.

Prerequisites

To integrate AD FS with Pivotal Cloud Foundry (PCF), you need the following:

Pivotal

- PCF, version 1.7.0 or later
- Single Sign-On, version 1.1.0 or later

Active Directory Federation Services

- Active Directory Federation Services subscription
- A user with Administrative privileges

Note: To configure SAML, you must have the Single Sign-On service broker installed on your PCF deployment. You need to create a plan, grant any plan administrators, and specify any organizations this plan should be the authentication authority for. For help configuring plans, see the Manage Service Plans topic.

Active Directory Federation Services Integration Guide

Configuring AD FS with SSO

Complete both steps below to integrate your deployment with AD FS and SSO.

1. Configure Active Directory Federation Services as an Identity Provider
2. Configure a Single Sign-On Service Provider

Testing and Troubleshooting

- Testing
- Troubleshooting

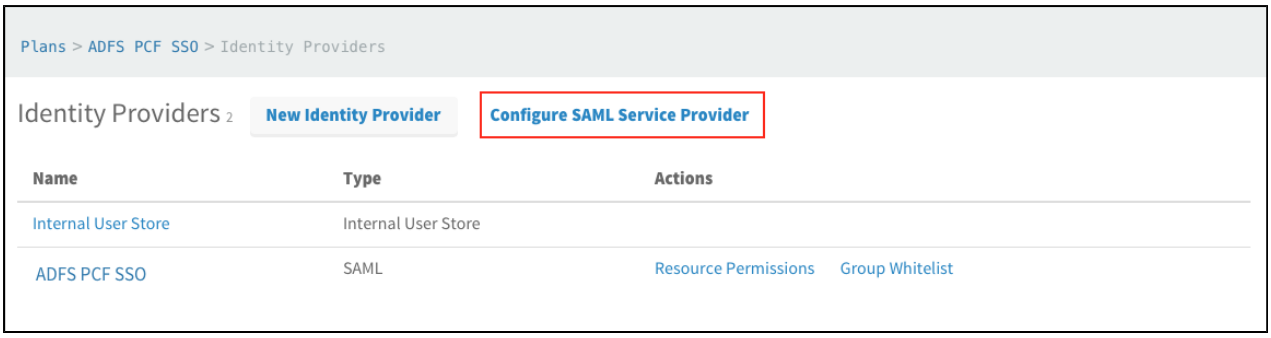
Configure Active Directory Federation Services as an Identity Provider

This topic describes how to set up Active Directory Federation Services (AD FS) as your identity provider by configuring SAML integration in both Pivotal Cloud Foundry (PCF) and AD FS.

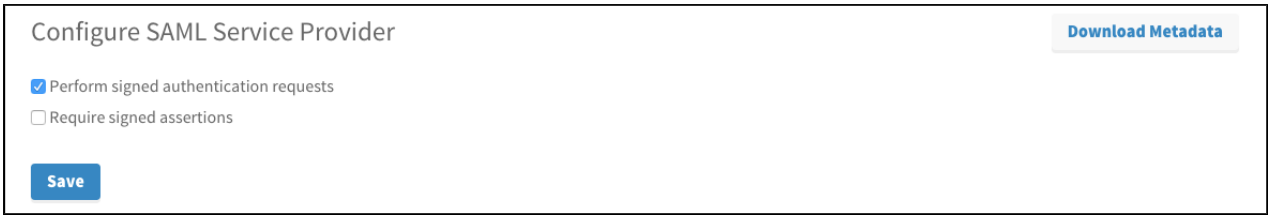
Set up SAML in PCF

1. Log in to the Single Sign-On (SSO) dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.
2. Select your plan and click **Manage Identity Providers** on the dropdown menu.

3. Click **Configure SAML Service Provider**.



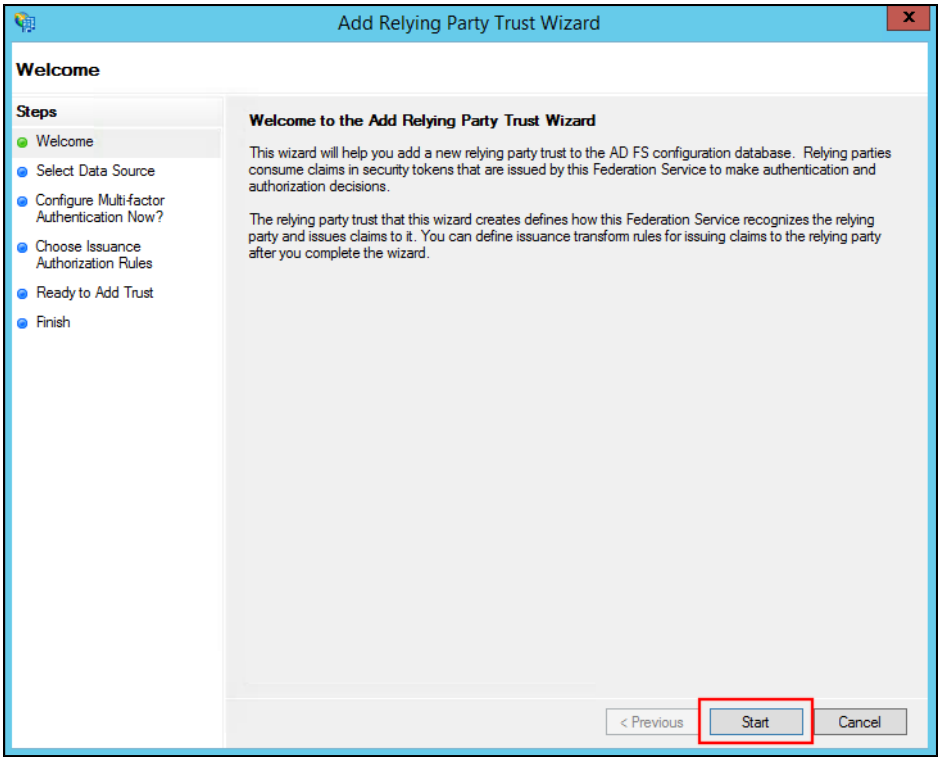
4. (Optional) Select **Perform signed authentication requests** to enforce SSO private key signature and identity provider validation.



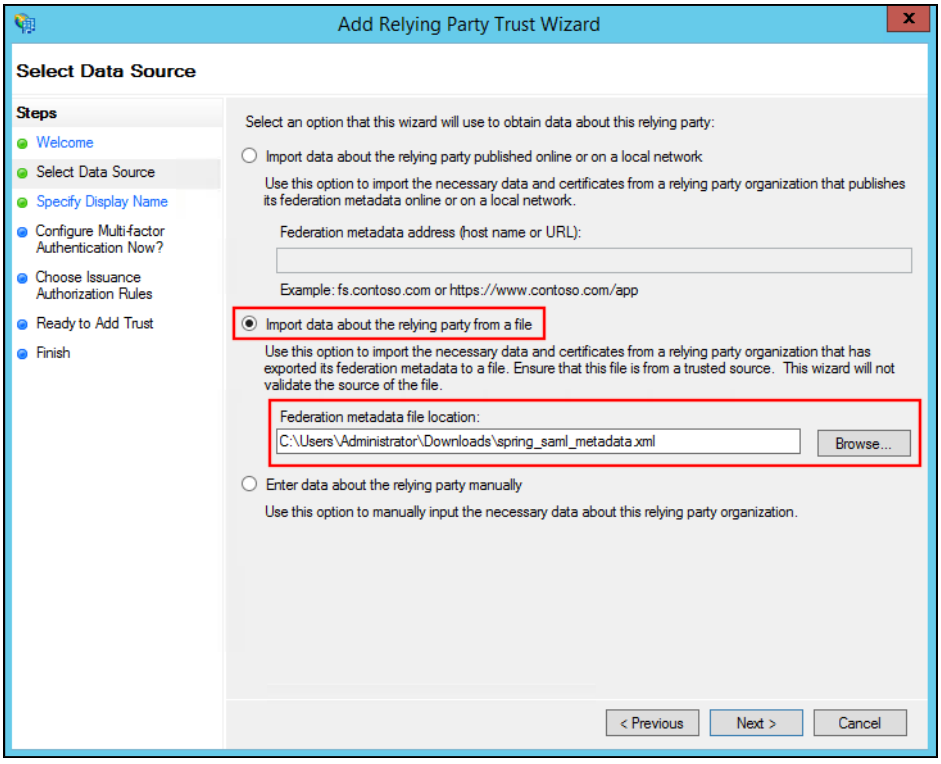
5. (Optional) Select **Require signed assertions** to validate the origin of signed responses.
6. Click **Download Metadata** to download the service provider metadata.
7. Click **Save**.

Set up SAML in Active Directory Federation Services

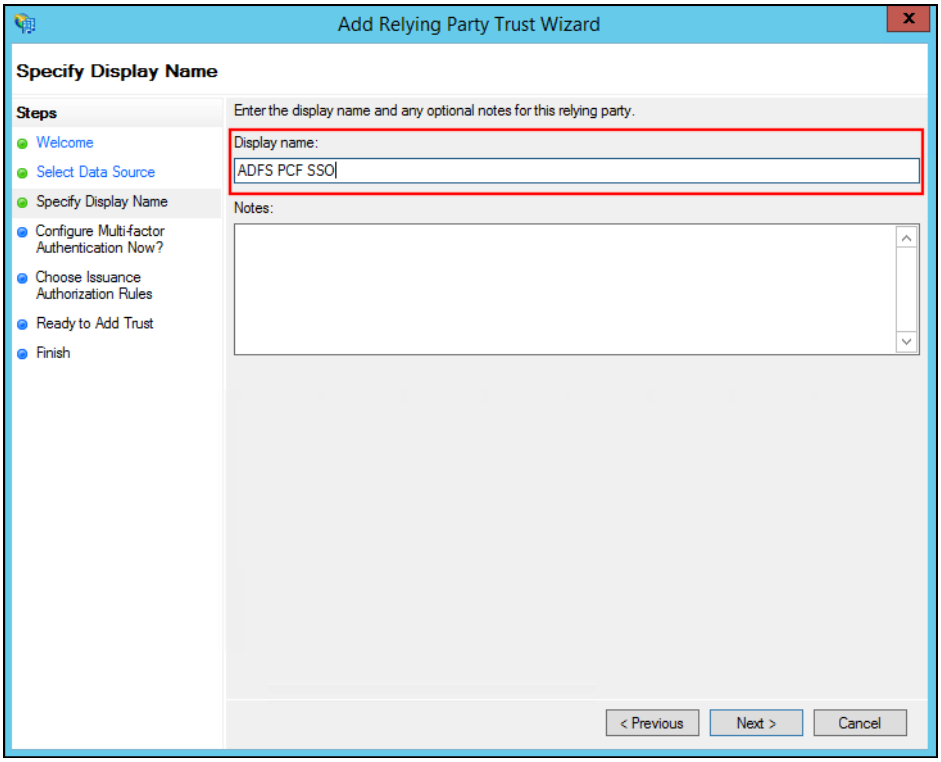
1. Open the **AD FS Management** console.
2. Click **Add Relying Party Trust...** in the Actions pane.
3. On the Welcome step, click **Start**.



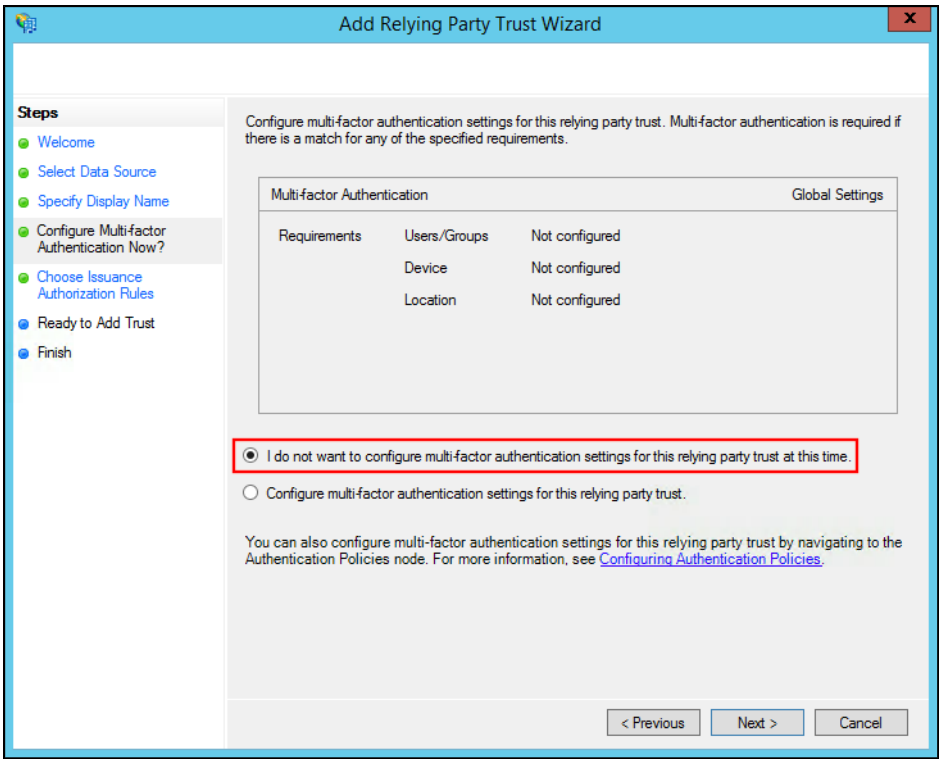
4. Select **Import data about the relying party from a file**, enter the path to the downloaded service provider metadata, and click **Next**.



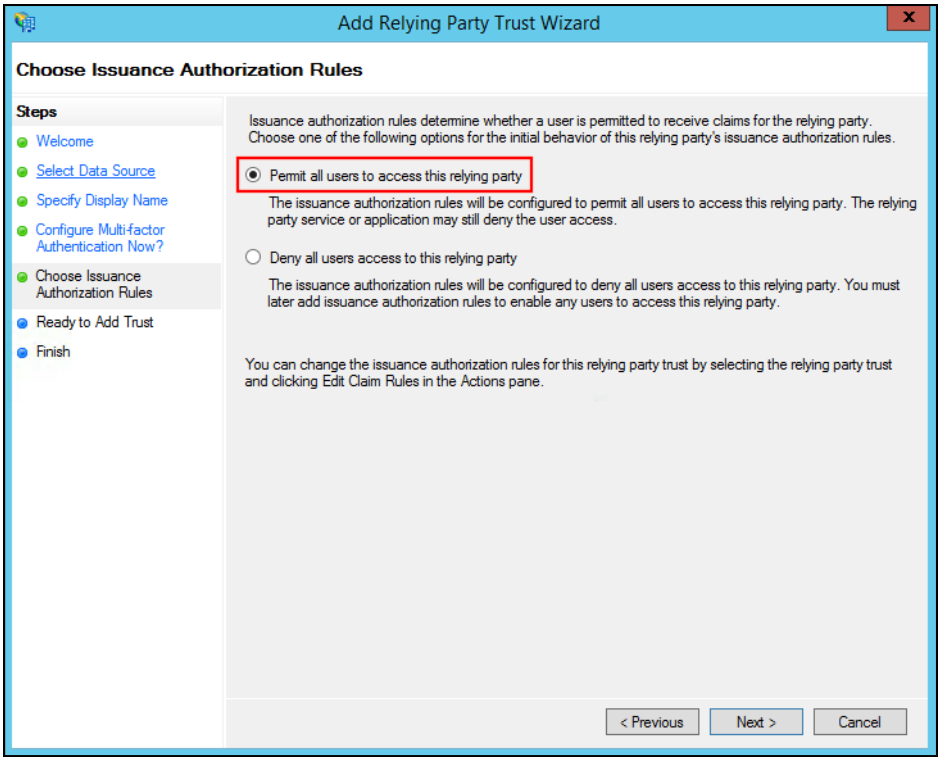
5. Enter a name for **Display name** and click **Next**.



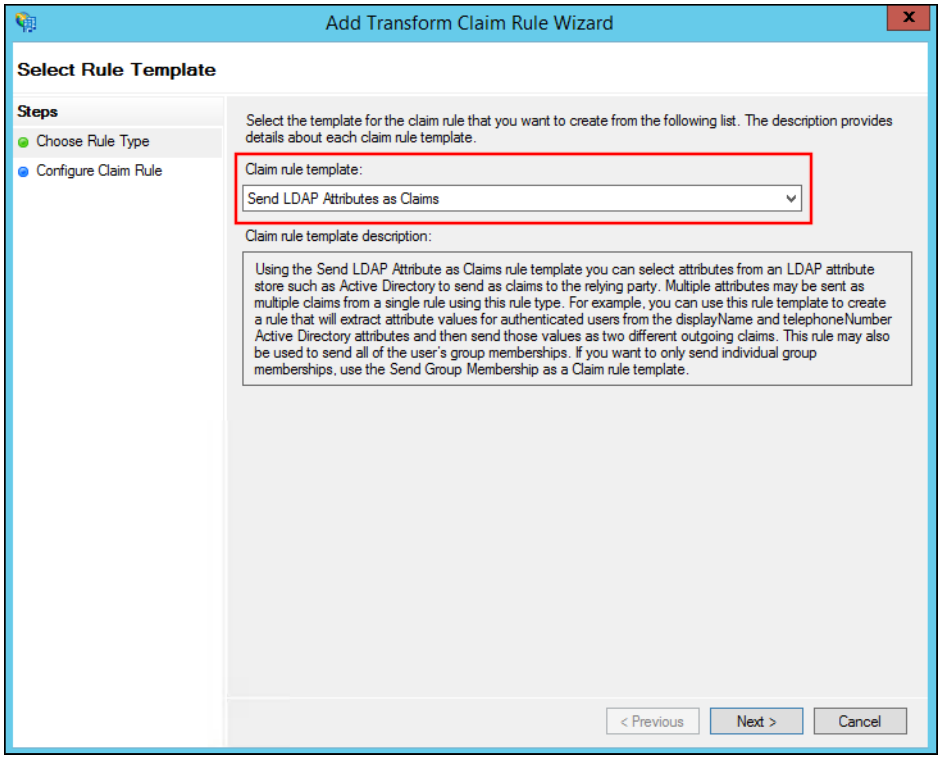
6. Leave the default multi-factor authentication selection and click **Next**.



7. Select **Permit all users to access this relying party** and click **Next**.

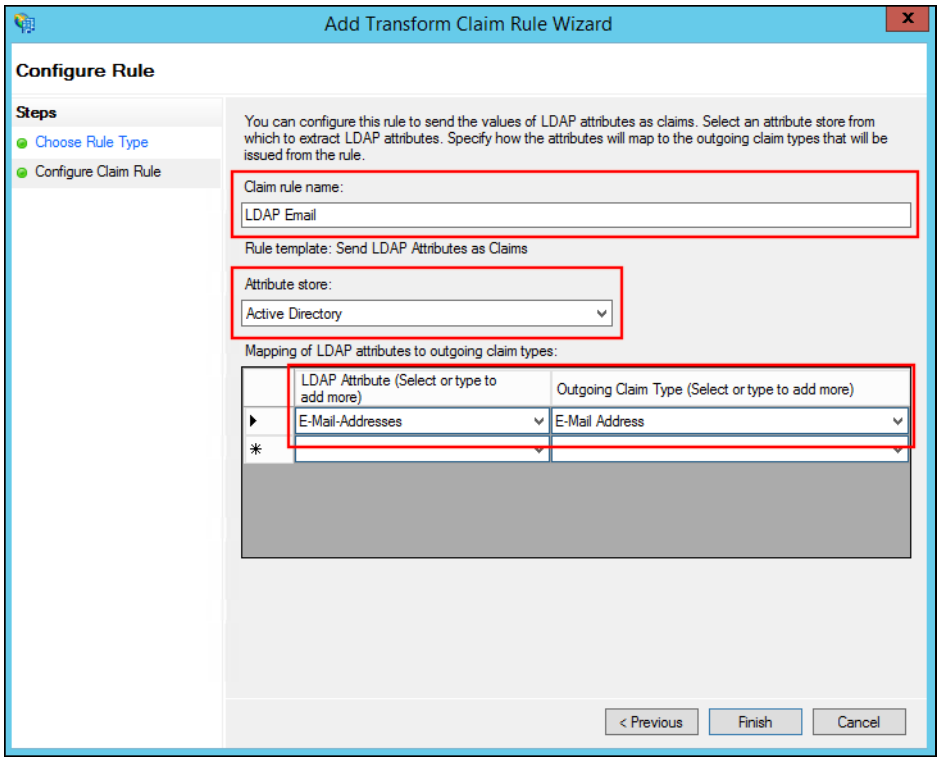


8. Review your settings and click **Next**.
9. Click **Close** to finish the wizard.
10. The claim rule editor should open by default. If it does not, select your Relying Party Trust and click **Edit Claim Rules...** in the Actions pane.
11. Create two claim rules by following these steps:
    a. Click **Add Rule**.
    b. Select **Send LDAP Attributes as Claims** for **Claim rule template** and click **Next**.



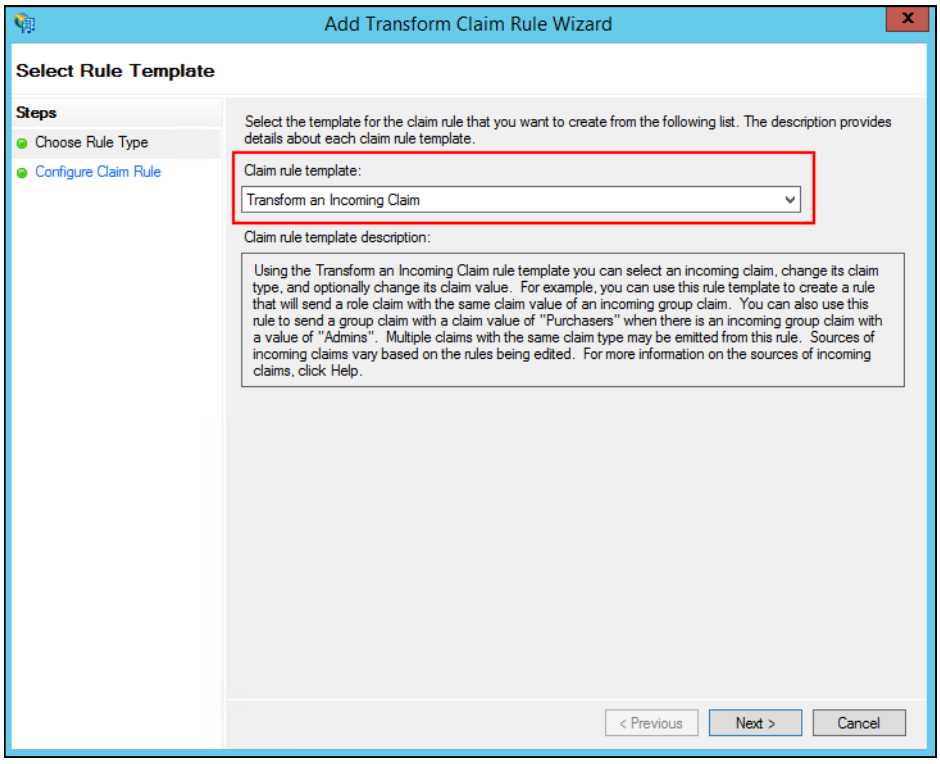
    c. Enter a **Claim rule name**.
    d. Select **Active Directory** for **Attribute store**.
    e. Select **E-Mail-Addresses** for **LDAP Attribute** and select **E-mail Address** for **Outgoing Claim Type**.

f. Click **Finish**.



g. Click **Add Rule**.

h. Select **Transform an Incoming Claim** for **Claim rule template** and click **Next**.



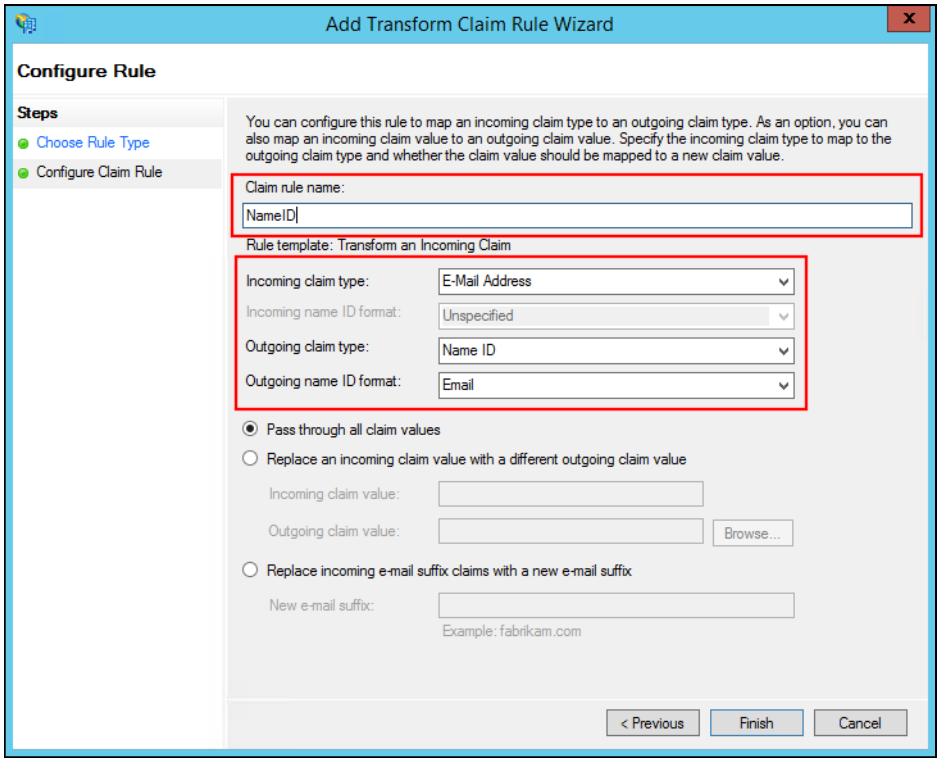
i. Enter a **Claim rule name**.

j. Select **E-Mail Address** for **Incoming claim type**.

k. Select **Name ID** for **Outgoing claim type**

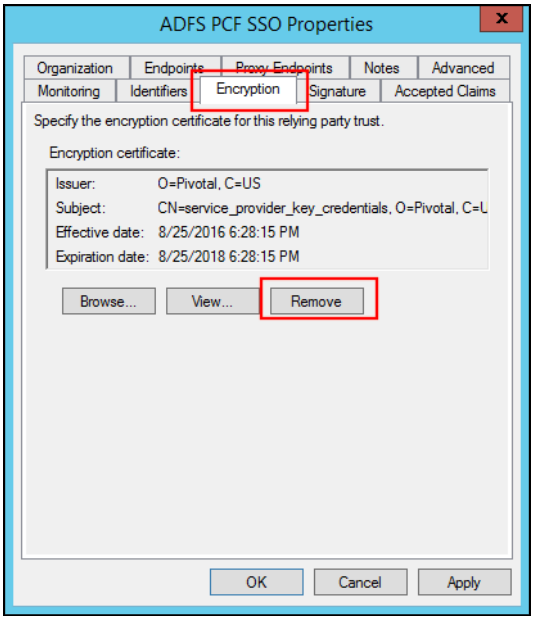
l. Select **Email** for **Outgoing name ID format**.

m. Click **Finish**.

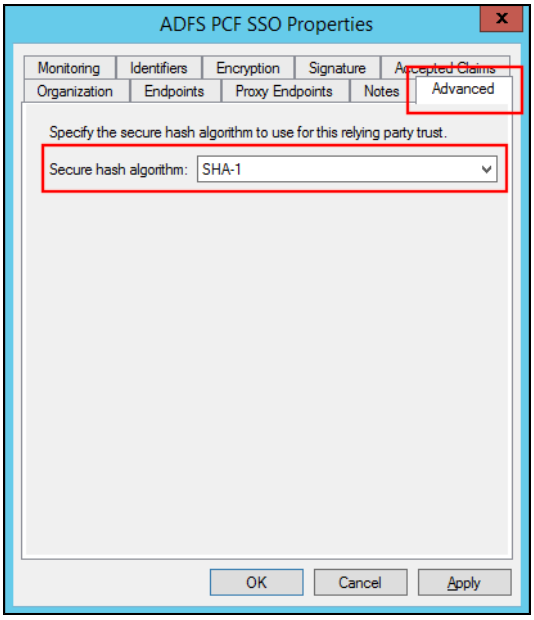


12. Double-click on the new Relying Party Trust to open the properties.

13. Select the **Encryption** tab and click **Remove** to remove the encryption certificate.



14. Select the **Advanced** tab and select **SHA-1** for the **Secure hash algorithm**.



15. (Optional) If you are using a self-signed certificate, disable CRL checks by following these steps:

    a. Open **Windows Powershell** as an Administrator.
    b. Execute the following command:

```
> set-ADFSRelyingPartyTrust -TargetName "< Relying Party Trust >" -SigningCertificateRevocationCheck None
```

16. (Optional) If you are using a self-signed certificate, add it to the ADFS trust store. Obtain the OpsManager certificate from `https://OPS_MANAGER_IP/api/v0/security/root_ca_certificate` and add this CA certificate to the ADFS trust store, so ADFS can trust the “Service Provider Key Certificate” certificate signed by OpsManager ROOT CA.

Note: Prior to PCF 1.10+, steps 13 and 14 are required as all PCF components (including SSO tile) have certificates are signed by an internal CA. In PCF 1.10+, customers can upload their own CA certificate to PCF.

1. (Optional) To specify any application or group attributes that you want to map to users in the ID token, click **Edit Claim Rules…** and configure **Send LDAP Attributes as Claims**.

Configure a Single Sign-On Service Provider

This topic describes how to add an external identity provider to your Pivotal Single Sign-On (SSO) service plan.

Download Identity Provider Metadata

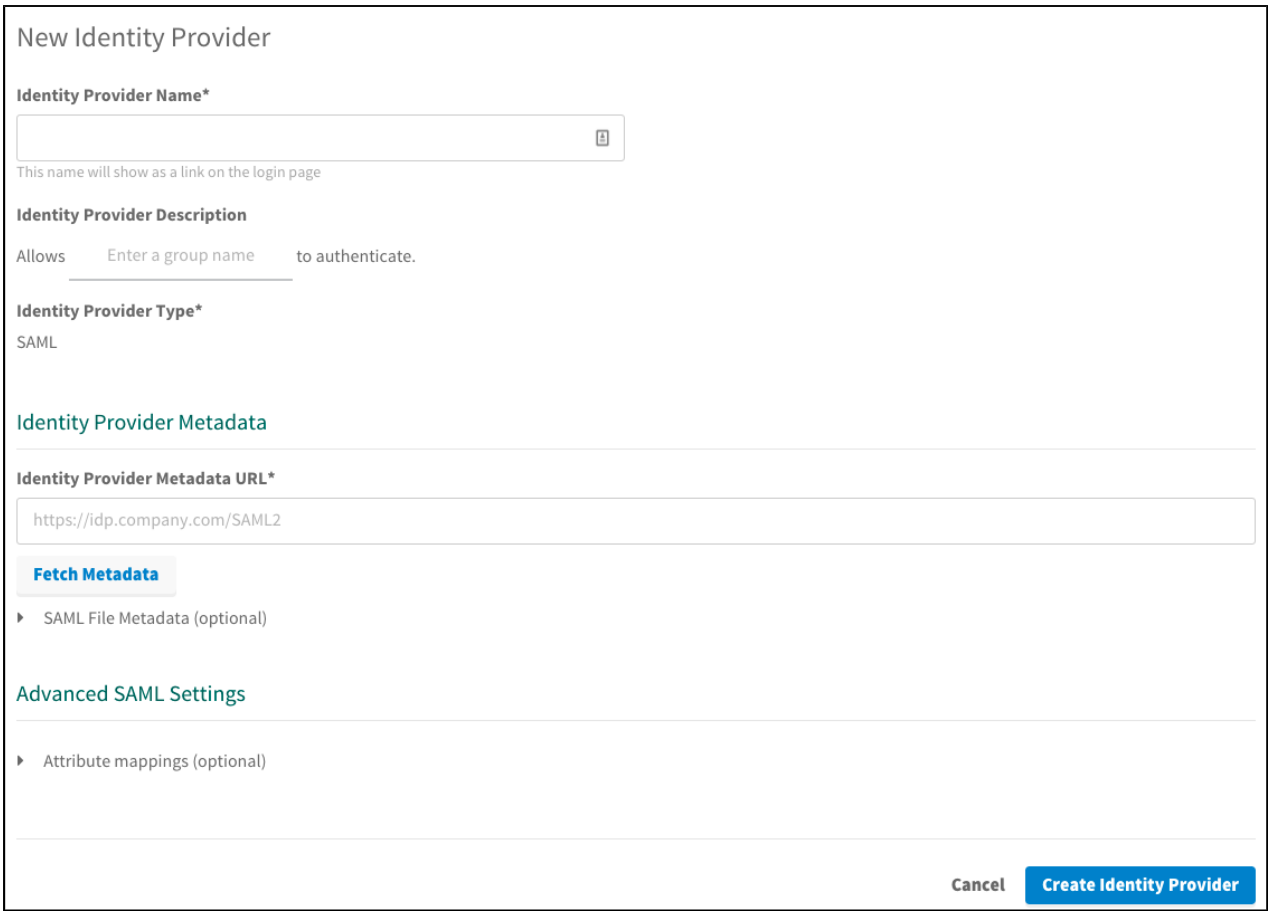
1. Download the metadata from your Active Directory Federation Services server at the following URL:
   `https://YOUR-ADFS-HOSTNAME/federationmetadata/2007-06/federationmetadata.xml`

Setting up SAML

1. Log in to the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.

2. Select your plan and click **Manage Identity Providers** on the dropdown menu.

3. Click **New Identity Provider** to create a new identity provider.



4. To create a new identity provider, perform the following steps:

   a. Enter an identity provider name in **Identity Provider Name**.

   b. (Optional) Enter a description in **Identity Provider Description**.
   c. Click **SAML File Metadata (optional)**, then click **Upload Identity Provider Metadata** to upload your metadata XML.
   d. (Optional) Under **Advanced SAML Settings**, click **Attribute Mappings** to enter the mappings.

5. Click **Create Identity Provider**.

6. Click **Resource Permissions**.

7. Click **New Permissions Mapping** and perform the following steps:

   a. Enter a **Group Name**.
   b. For **Select Permissions**, select the permissions to grant to the members of the group from the external identity provider.

8. Navigate to the identity provider list.

9. Click **Group Whitelist** and enter the group names from the external identity provider that should be propagated in the ID token.

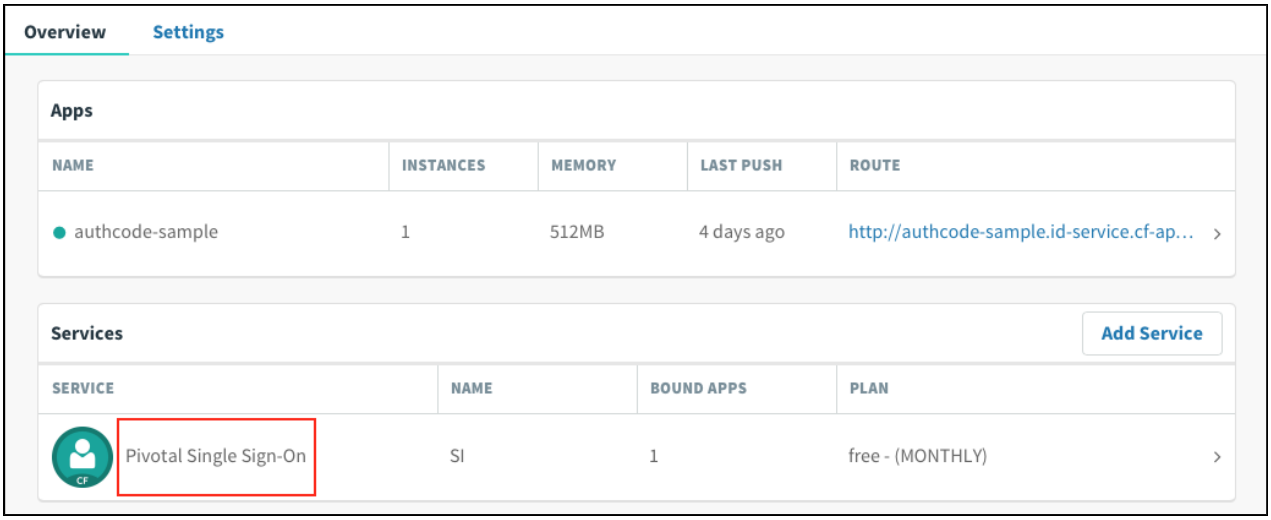
Testing

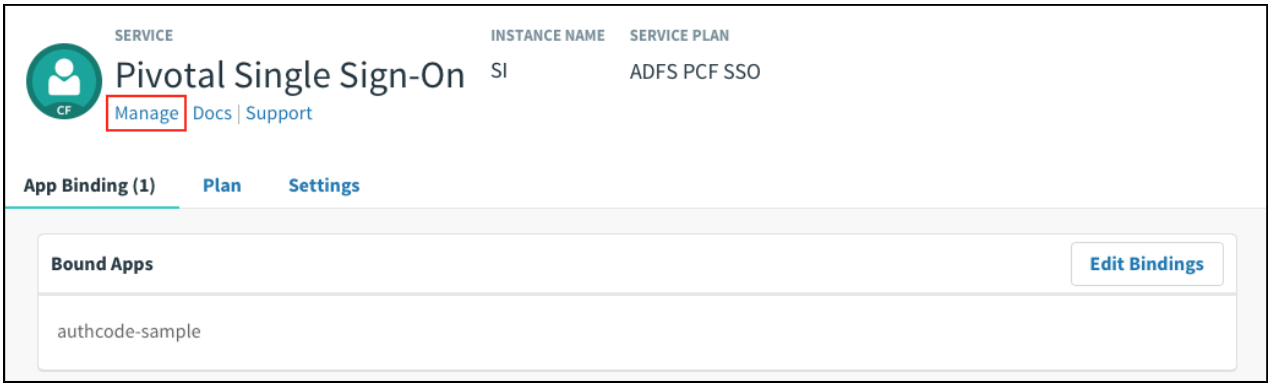
This topic describes how an administrator can test the connection between SSO and Active Directory Federation Services (AD FS). An administrator can test both service provider and identity provider connections.

Test Your Service Provider Connection

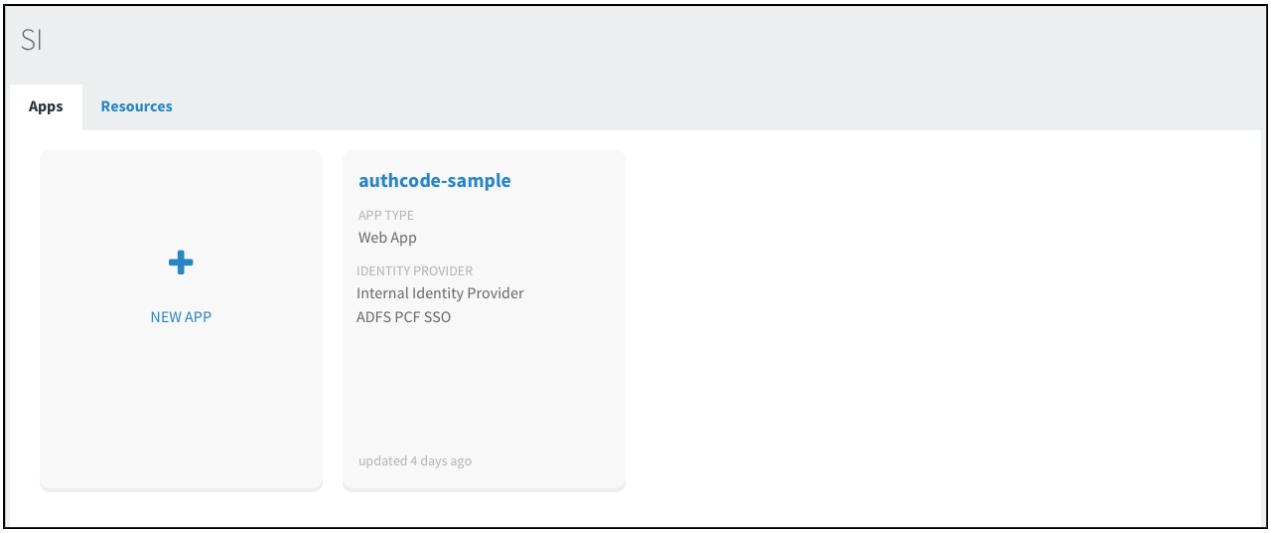
1. Log in to Apps Manager at `https://apps.YOUR-SYSTEM-DOMAIN` and navigate to the organization and space where your application is located.

2. Under **Services**, locate the service instance of the Single Sign-On (SSO) plan bound to your application. Click on the service instance and click **Manage**.

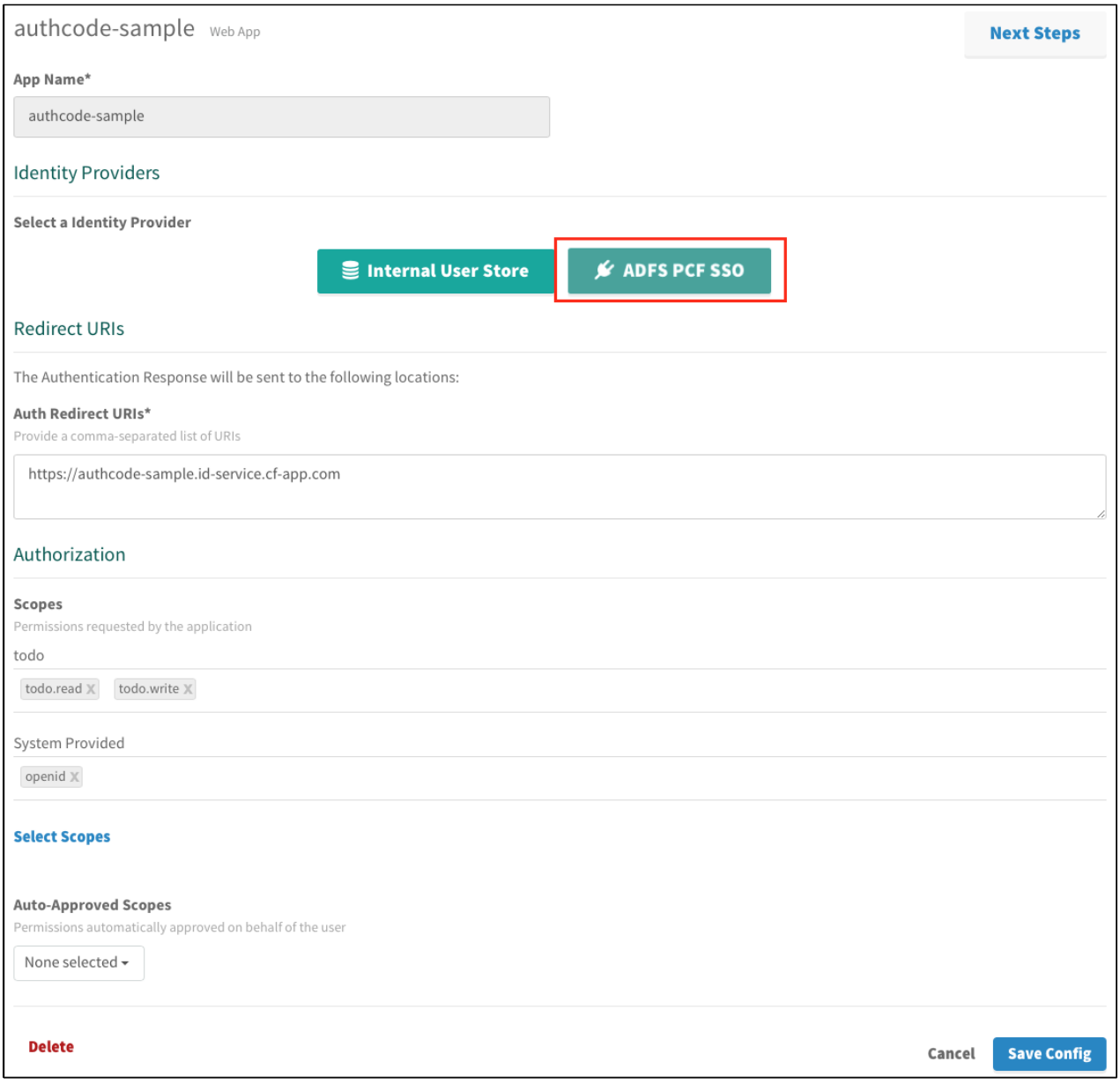




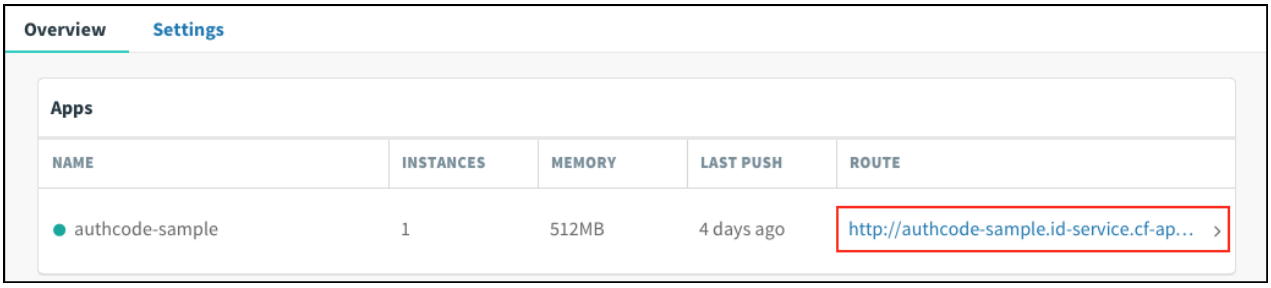
3. Under the **Apps** tab, click your application.



4. Under **Identity Providers**, select the AD FS identity provider.



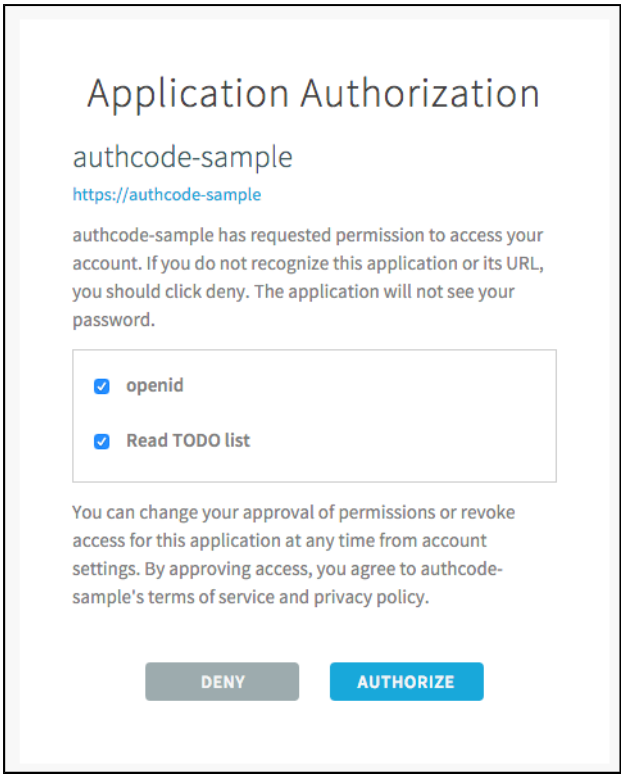
5. Return to Apps Manager and click on the URL below your application to be redirected to the identity provider to authenticate.



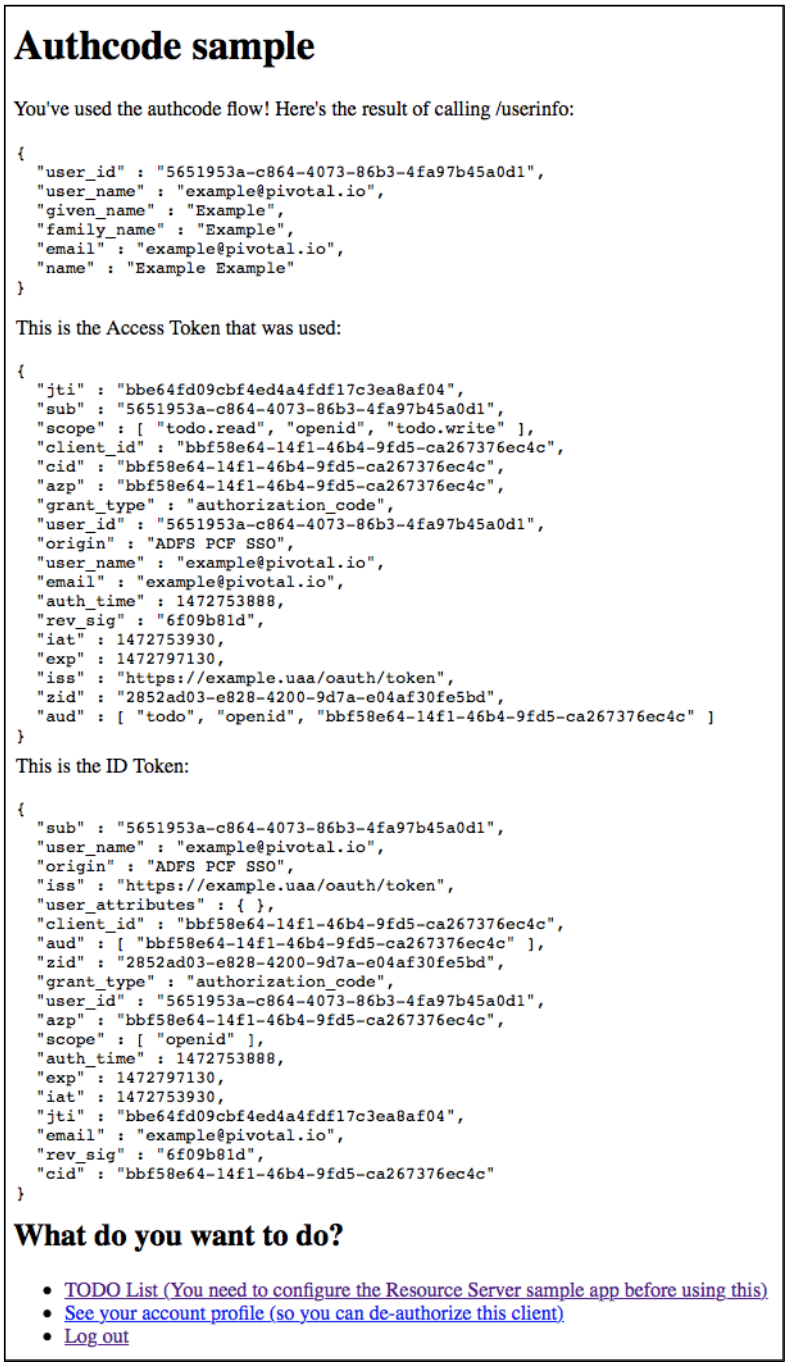
6. Click the link.

7. On the identity provider sign-in page, enter your credentials and click **Sign in**.

8. The application asks for authorization to the necessary scopes. Click **Authorize**.



9. The access token and ID token displays.



Authcode sample

You've used the authcode flow! Here's the result of calling /userinfo:


```
{
  "user_id" : "5651953a-c864-4073-86b3-4fa97b45a0d1",
  "user_name" : "example@pivotal.io",
  "given_name" : "Example",
  "family_name" : "Example",
  "email" : "example@pivotal.io",
  "name" : "Example Example"
}
```


This is the Access Token that was used:


```
{
  "jti" : "bbe64fd09cbf4ed4a4fdf17c3ea8af04",
  "sub" : "5651953a-c864-4073-86b3-4fa97b45a0d1",
  "scope" : [ "todo.read", "openid", "todo.write" ],
  "client_id" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c",
  "cid" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c",
  "azp" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c",
  "grant_type" : "authorization_code",
  "user_id" : "5651953a-c864-4073-86b3-4fa97b45a0d1",
  "origin" : "ADFS PCF SSO",
  "user_name" : "example@pivotal.io",
  "email" : "example@pivotal.io",
  "auth_time" : 1472753888,
  "rev_sig" : "6f09b81d",
  "iat" : 1472753930,
  "exp" : 1472797130,
  "iss" : "https://example.uaa/oauth/token",
  "zid" : "2852ad03-e828-4200-9d7a-e04af30fe5bd",
  "aud" : [ "todo", "openid", "bbf58e64-14f1-46b4-9fd5-ca267376ec4c" ]
}
```


This is the ID Token:


```
{
  "sub" : "5651953a-c864-4073-86b3-4fa97b45a0d1",
  "user_name" : "example@pivotal.io",
  "origin" : "ADFS PCF SSO",
  "iss" : "https://example.uaa/oauth/token",
  "user_attributes" : { },
  "client_id" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c",
  "aud" : [ "bbf58e64-14f1-46b4-9fd5-ca267376ec4c" ],
  "zid" : "2852ad03-e828-4200-9d7a-e04af30fe5bd",
  "grant_type" : "authorization_code",
  "user_id" : "5651953a-c864-4073-86b3-4fa97b45a0d1",
  "azp" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c",
  "scope" : [ "openid" ],
  "auth_time" : 1472753888,
  "exp" : 1472797130,
  "iat" : 1472753930,
  "jti" : "bbe64fd09cbf4ed4a4fdf17c3ea8af04",
  "email" : "example@pivotal.io",
  "rev_sig" : "6f09b81d",
  "cid" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c"
}
```


What do you want to do?

- TODO List (You need to configure the Resource Server sample app before using this)
- See your account profile (so you can de-authorize this client)
- Log out

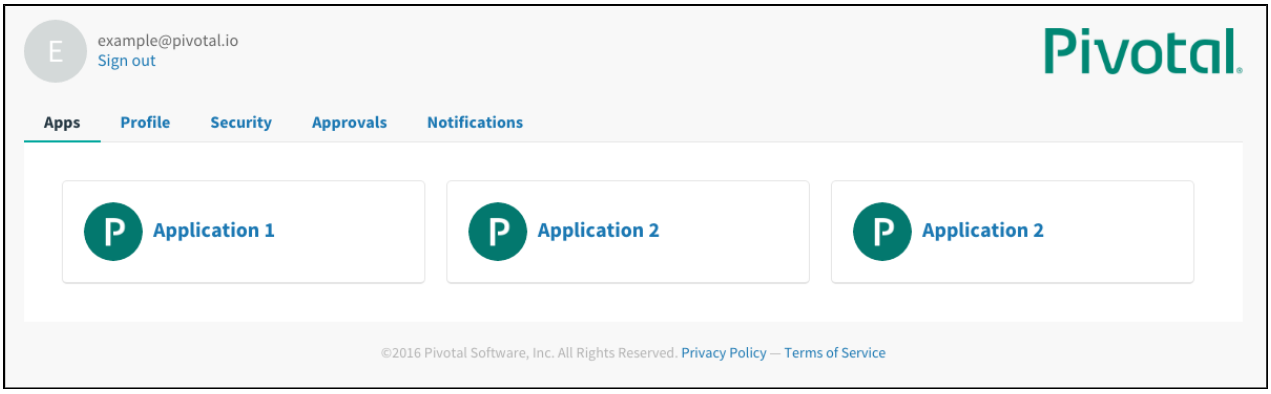
Test Your Identity Provider Connection

Note: SSO does not support identity provider-initiated flow into applications, but it does redirect the user to the User Account and Authentication (UAA) page to select applications assigned to the user.

1. Sign in to AD FS.

2. Navigate to your application and click it.

3. You are redirected to the page that lists applications you have access to.

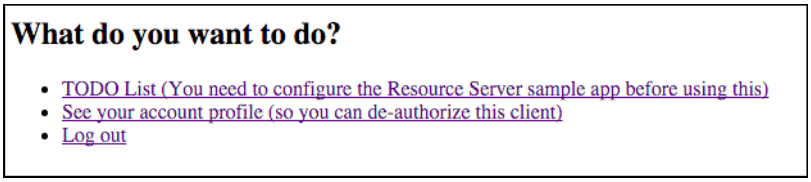


Test Your Single Sign-Off

Test single sign-off to ensure that when users log out of the application, they are logged out of AD FS as well.

1. Sign in to the sample application. Information about the access and ID token displays, as well as the “What do you want to do?” section.

2. Under “What do you want to do?”, click **Log out**.



3. You are logged out and redirected to the AD FS login page.

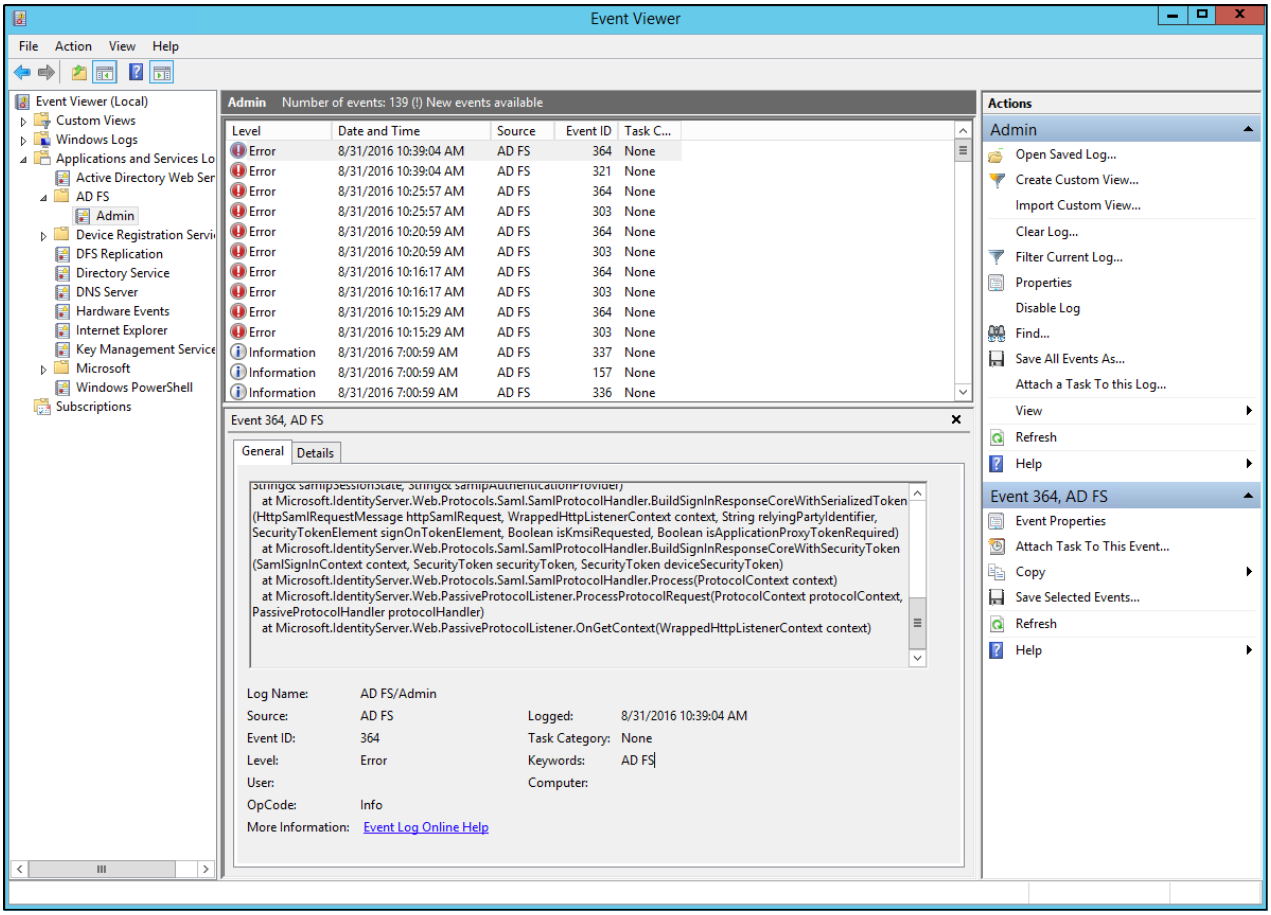
Troubleshooting

This topic describes how to resolve errors that arise when configuring a single sign-on partnership between Active Directory Federation Services and Pivotal Single Sign-On (SSO).

Event Viewer

1. Navigate to **Administrative Tools**.

2. Launch **Event Viewer**.



3. Examine any errors and its details to diagnose problems.

Azure Active Directory Integration Guide Overview

Azure Active Directory (Azure AD) is Microsoft's multi-tenant cloud based directory and identity management service. This documentation describes how to configure a single sign-on partnership between Azure AD as the Identity Provider (IdP) and the Single Sign-On Service (SSO) for Pivotal Cloud Foundry® as the Service Provider (SP).

SSO supports service provider-initiated authentication flow and single logout. It does not support identity provider-initiated authentication flow. All SSO communication takes place over SSL.

Prerequisites

To integrate Azure AD with Pivotal Cloud Foundry® (PCF), you need:

Pivotal

- PCF, version 1.7.0 or later.
- Single Sign-On, version 1.1.0 or later.

Azure Active Directory

- Azure Active Directory subscription.
- A user with admin privileges.

Note: To configure SAML, you must have the Single Sign-On service broker installed on your PCF deployment. You need to create a plan, grant any plan administrators, and specify any organizations this plan should be the authentication authority for. For help configuring plans, see the Manage Service Plans topic.

Azure AD Integration Guide

Configuring Azure AD with SSO

Complete both steps below to integrate your deployment with Azure AD and SSO.

1. Configure Azure AD as an Identity Provider
2. Configure a Single Sign-On Service Provider

Testing and Troubleshooting

- Testing
- Troubleshooting

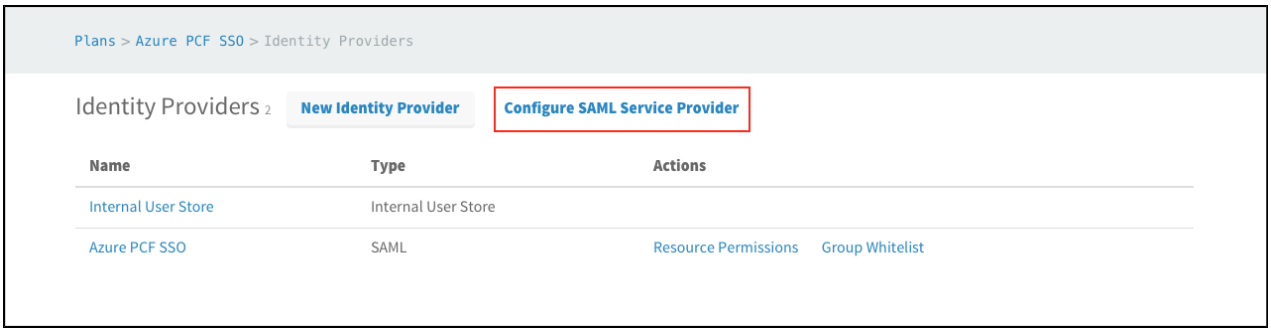
Configure Azure Active Directory as an Identity Provider

This topic describes how to set up Azure Active Directory (AD) as your identity provider by configuring SAML integration in both Pivotal Cloud Foundry® (PCF) and Azure AD.

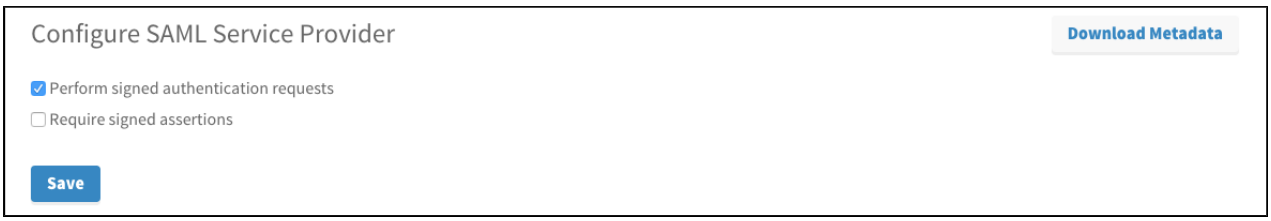
Set up SAML in PCF

1. Log into the Single Sign-On (SSO) dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.
2. Select your plan and click **Manage Identity Providers** on the dropdown menu.

3. Click **Configure SAML Service Provider**.



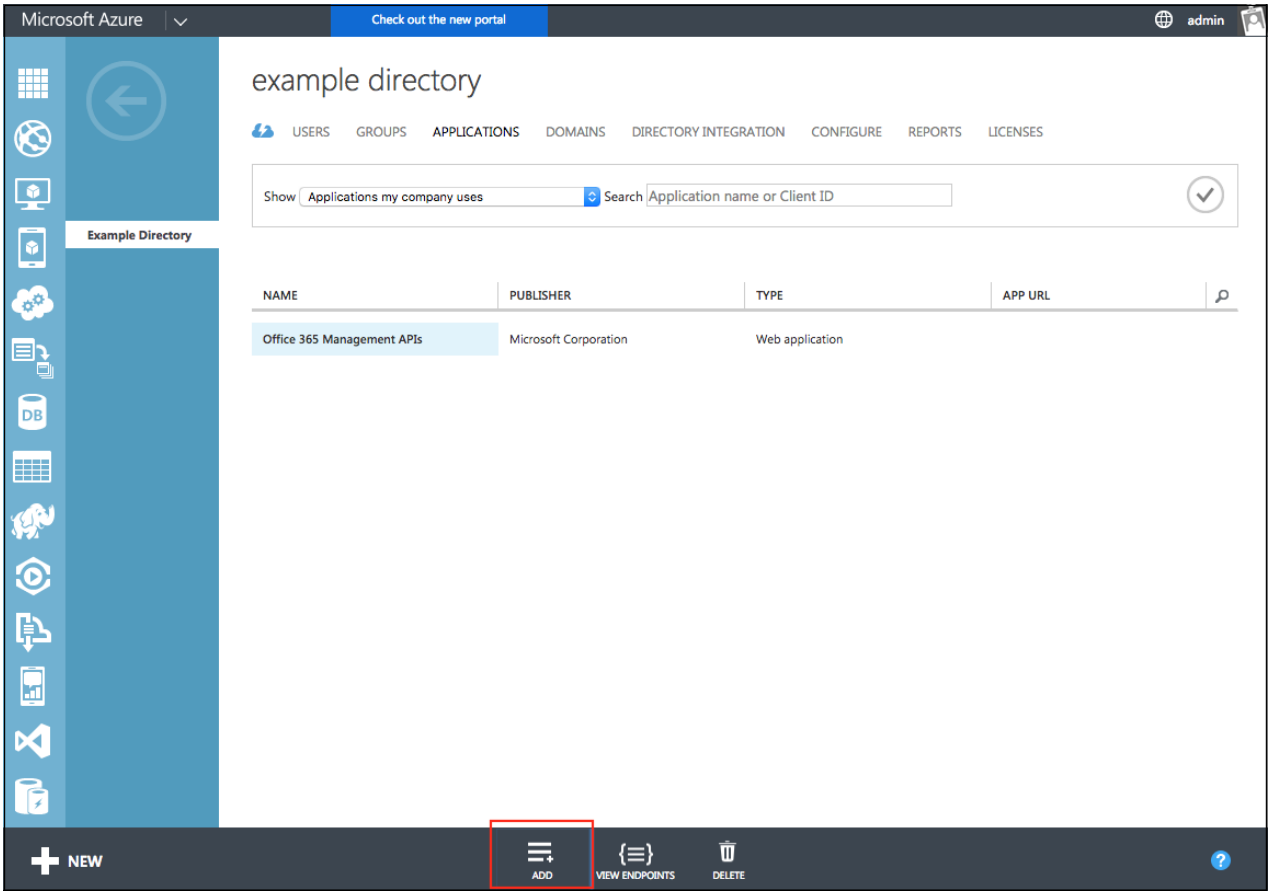
4. (Optional) Select **Perform signed authentication requests** to enforce SSO private key signature and identity provider validation.



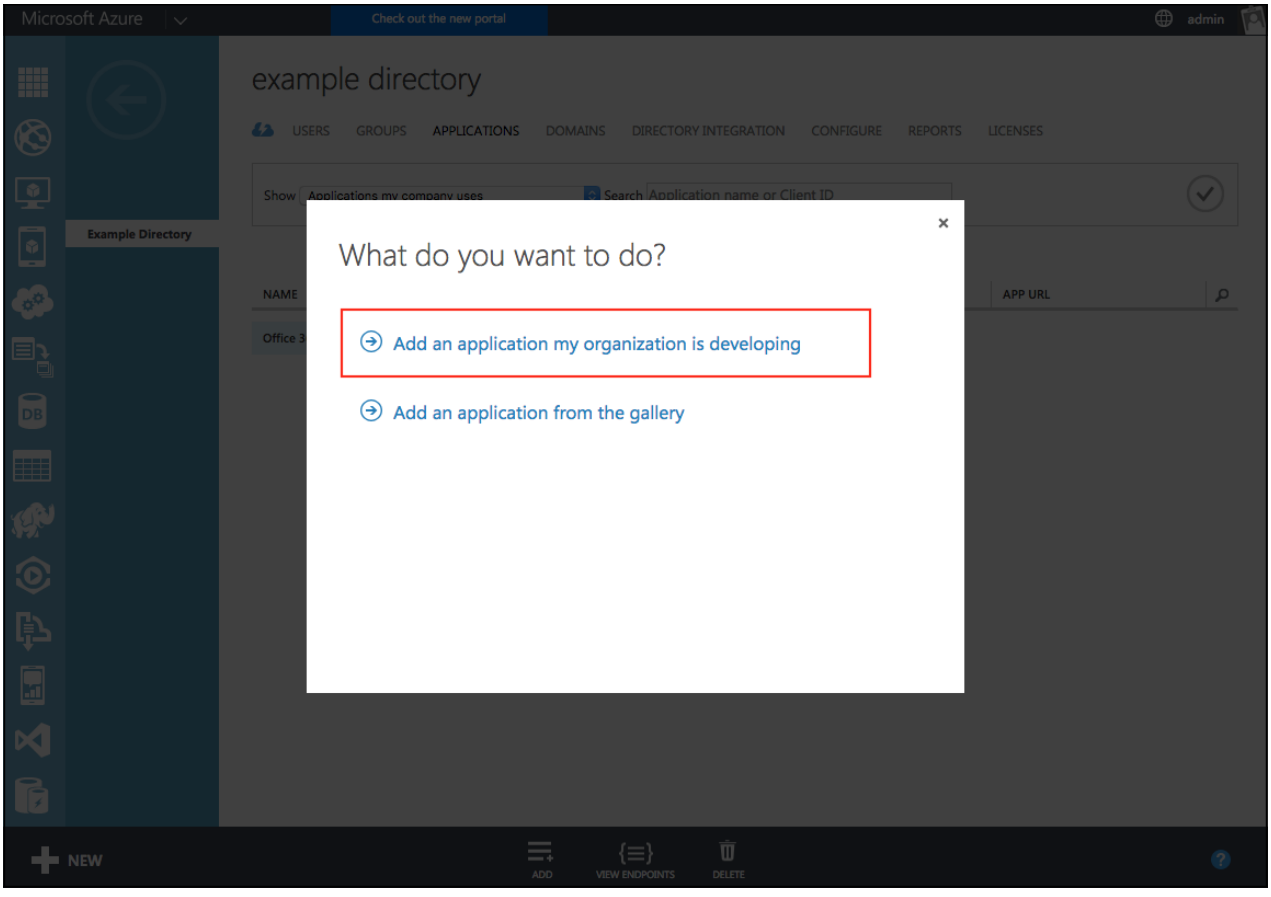
5. (Optional) Select **Require signed assertions** to validate the origin of signed responses.
6. Click **Download Metadata** to download the service provider metadata.
7. Click **Save**.

Set up SAML in Azure Active Directory

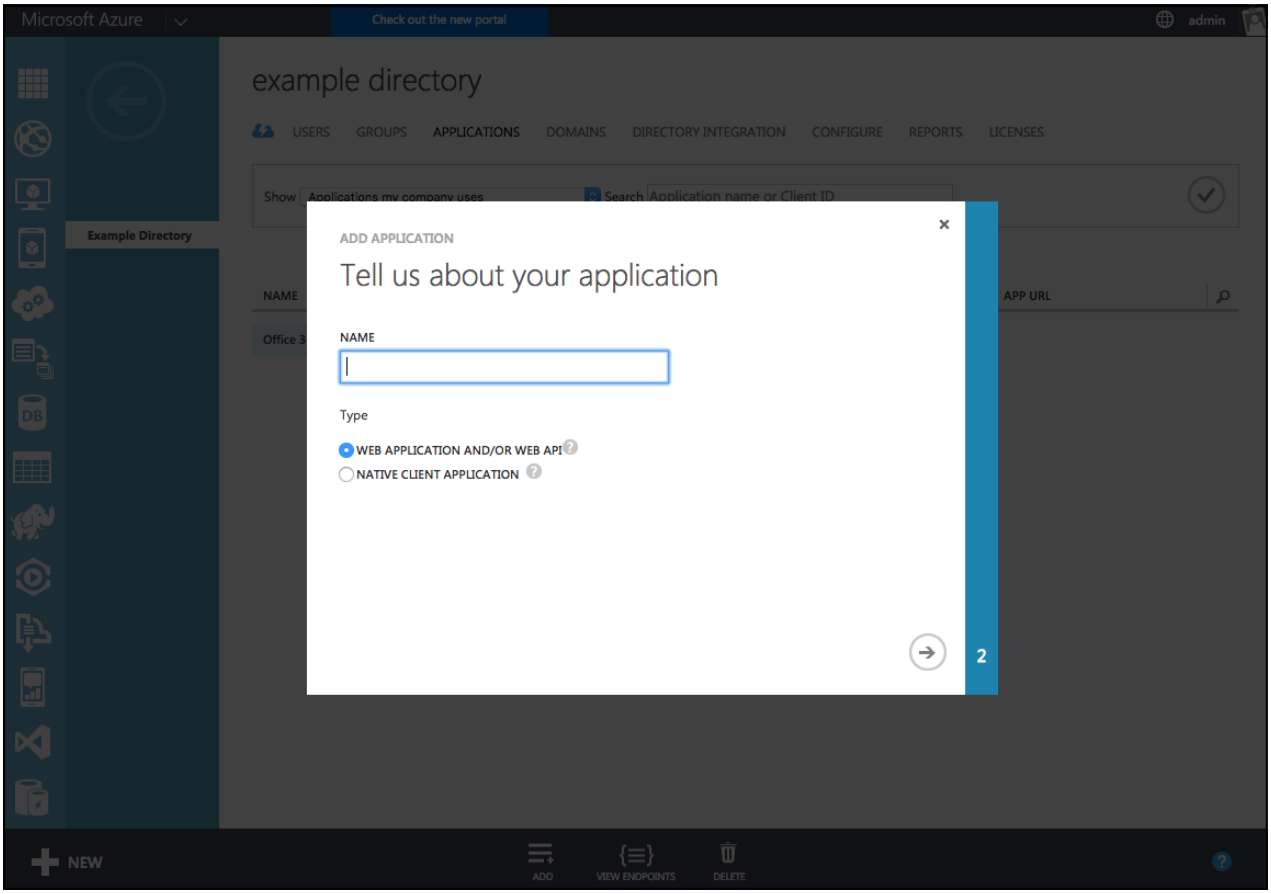
1. Sign into Azure AD at `https://manage.windowsazure.com` as an administrator.
2. Navigate to the applications dashboard by clicking on your directory and the **Applications** tab.
3. Click the **Add** button to add a new application.



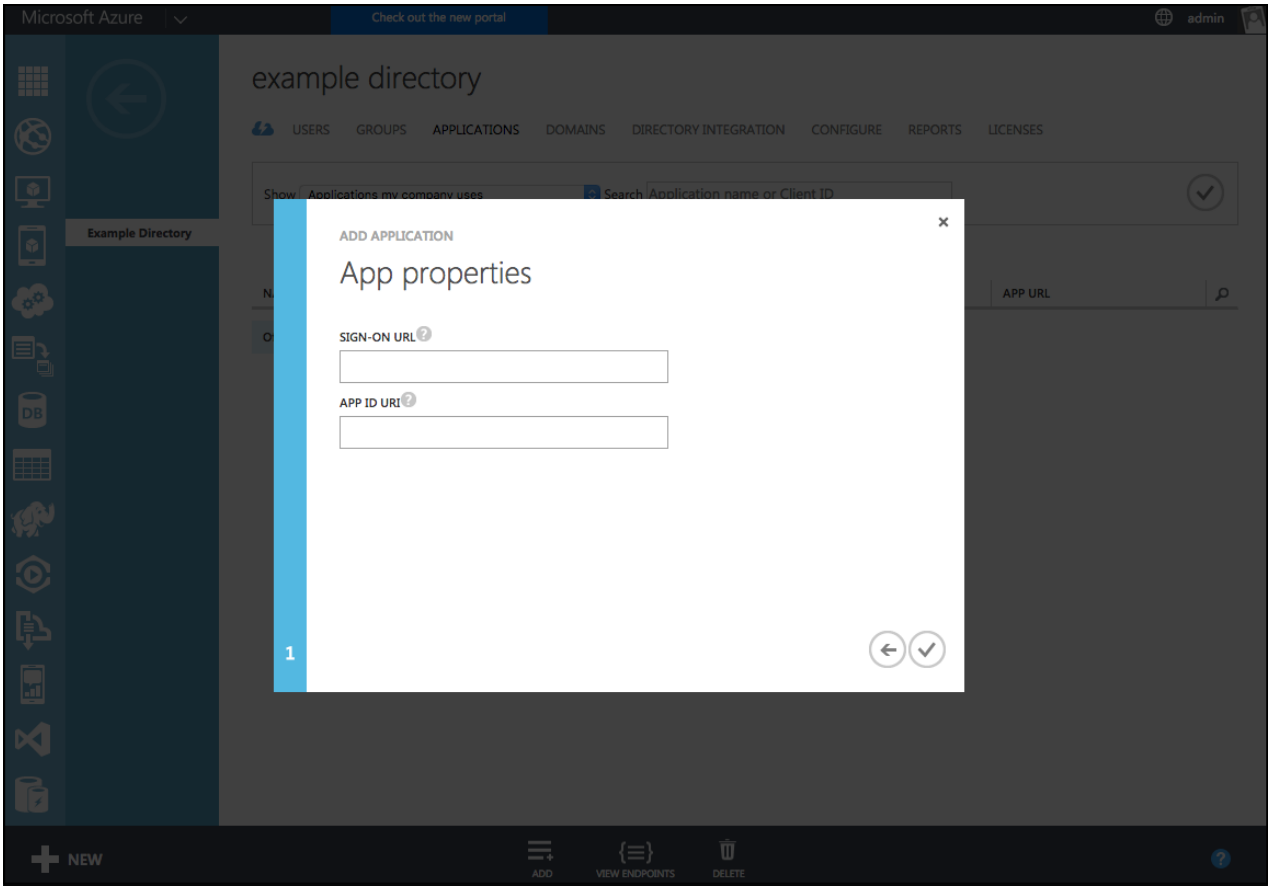
4. Select **Add an application my organization is developing**



5. Enter the **Name** and **Type** for the application.

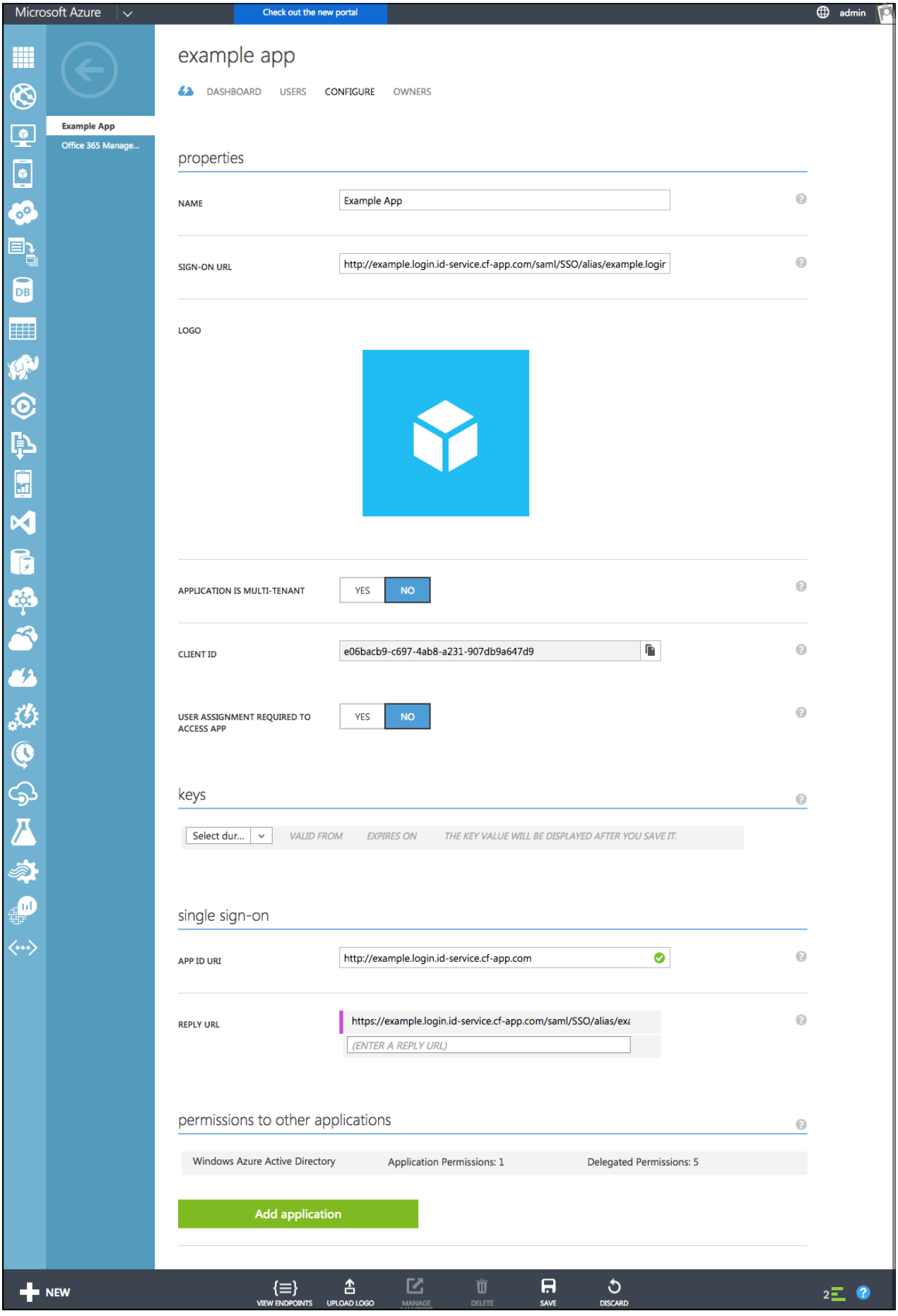


6. Enter the **Sign-On URL** and **App ID URI** for the application.



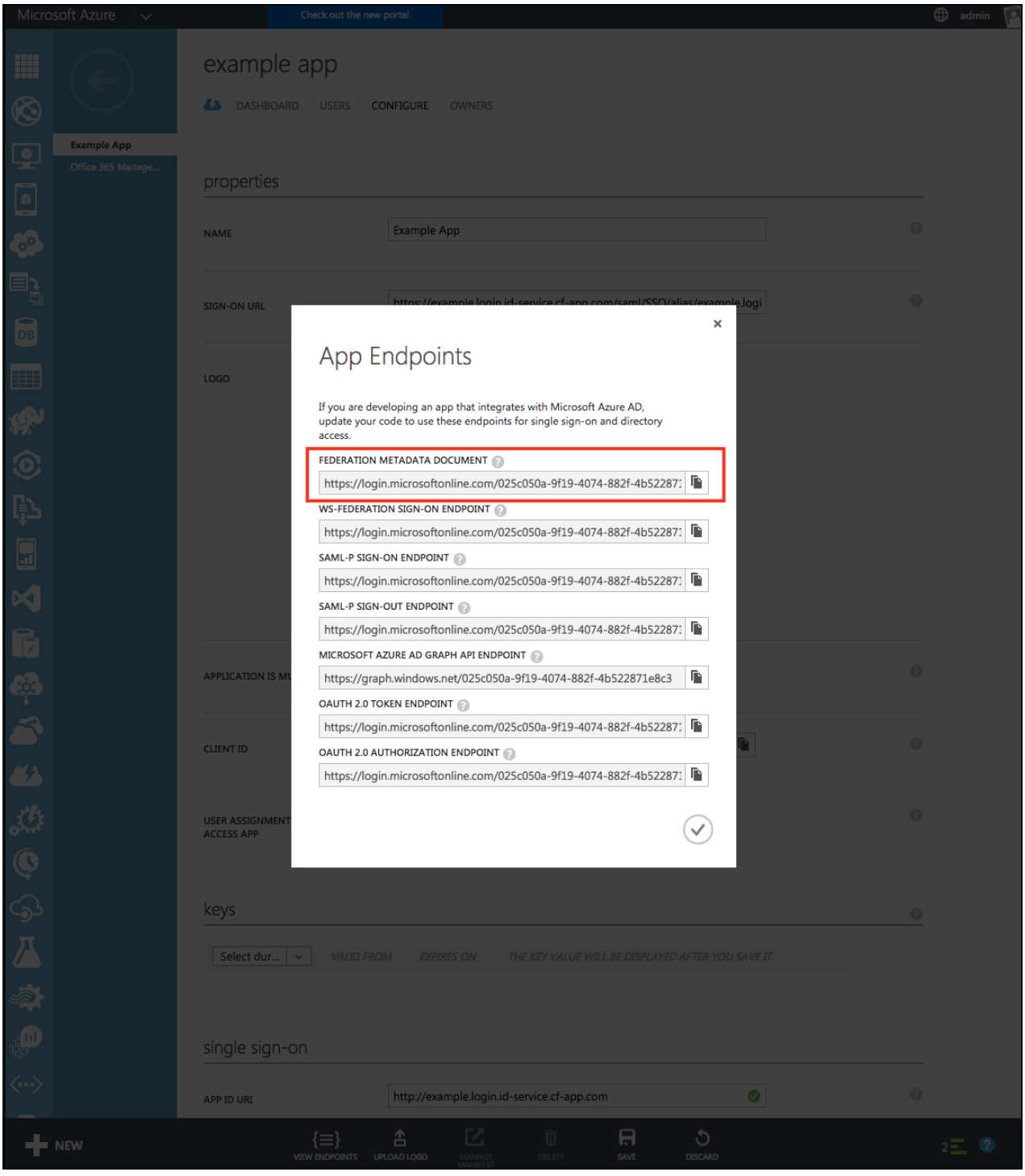
7. Click the application and configure the following properties:

a. Enter the application **Name**.
b. Enter the `AssertionConsumerService Location URL` from your downloaded service provider metadata into **Sign-On URL**. For example, `https://AUTH-DOMAIN/saml/SSO/alias/AUTH-DOMAIN`.
c. Configure the application **Logo**, **Application is Multi-Tenant** and **User Assignment Required to Access App** properties.
d. Enter your `Auth Domain URL` into **App ID URI**.
e. Enter the `AssertionConsumerService Location URL` from your downloaded service provider metadata into **Reply URL**.



8. Click the **Save** button.

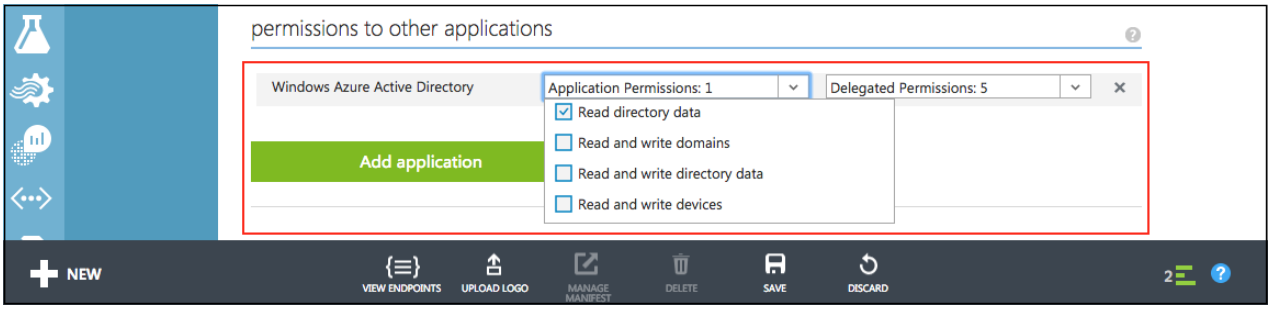
9. Click **View Endpoints** and download the **Federation Metadata Document**.



Set up Claims Mapping

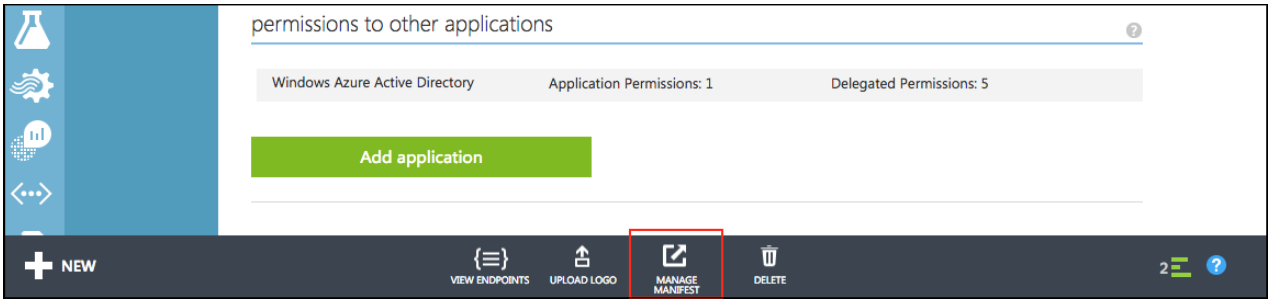
1. To enable user attribute mappings, grant the application the following permissions to Windows Azure Active Directory:

    a. Read directory data.
    b. Read all groups.
    c. Read all users' full profiles or Read all users' basic profiles.



2. To pass group membership claims to the application, perform the following steps:

   a. Click **Manage Manifest**.
   b. Click **Download Manifest** followed by **Download manifest**.
   c. Locate `groupMembershipClaims` and set the value to either:
      - `SecurityGroup` - Groups claim will contain identifiers of all security groups of which the user is a member.
      - `All` - Groups claim will contain the identifiers of all security groups and distribution lists of which the user is a member.
   d. Click **Manage Manifest**.
   e. Click **Upload Manifest** and select the modified manifest.



Configure a Single Sign-On Service Provider

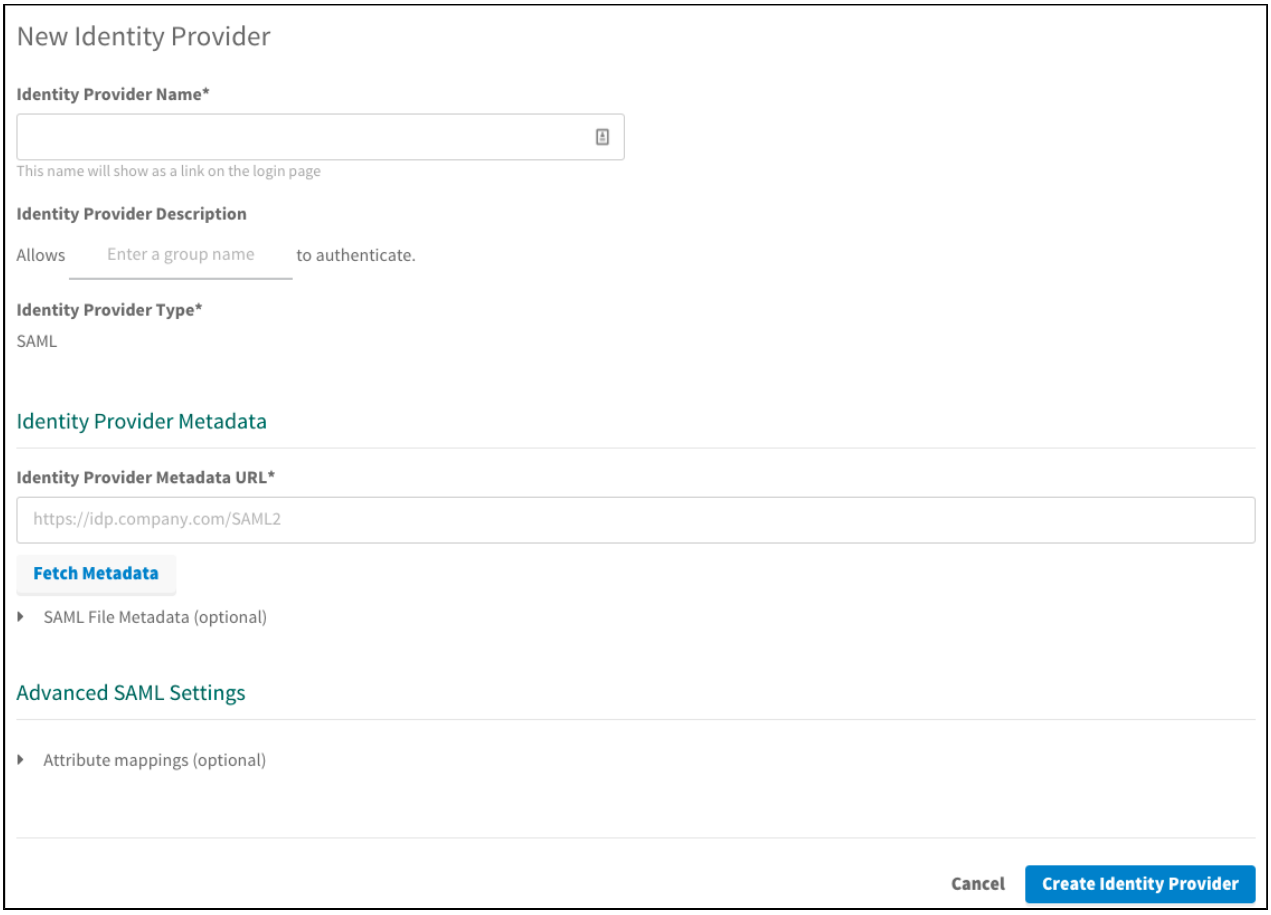
This topic describes how to add an external identity provider to your Pivotal Single Sign-On (SSO) service plan.

Setting up SAML

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.

2. Select your plan and click **Manage Identity Providers** on the dropdown menu.

3. Click **New Identity Provider** to create a new identity provider.



4. To create a new identity provider, perform the following steps:

    a. Enter an identity provider name into **Identity Provider Name**.
    b. (Optional) Enter a description into **Identity Provider Description**.
    c. Click **SAML File Metadata (optional)** follwed by clicking the **Upload Identity Provider Metadata** button to upload your metadata XML.

> **Note**: The Single Sign-On does not support DOS file format imports. Convert the file in one of the following ways:
>
> - Option 1: Execute `dos2unix` on the metadata file.
> - Option 2: Create a Unix file, then copy and paste the contents from the downloaded metadata file to the newly created file.

d. (Optional) Under **Advanced SAML Settings**, click **Attribute Mappings** to enter the mappings.

5. Click **Create Identity Provider**.

Configure Group Permissions

1. Add groups to be propagated from the external identity provider to the ID token by following these steps:

    a. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.
    b. Select your plan and click **Manage Identity Providers** on the dropdown menu.
    c. Click **Group Whitelist** next to your identity provider.
    d. Enter the group names.
    e. Click **Save Group Whitelist**.

2. Map the groups to resources defined in the SSO service by following these steps:

    a. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.
    b. Select your plan and click **Manage Identity Providers** on the dropdown menu.
    c. Click **Resource Permissions**.
    d. Click **New Permissions Mapping** and perform the following steps:

        i. Enter a **Group Name**.
        ii. For **Select Permissions**, select the permissions that the members of the group from the external identity provider should have access to.
        iii. Click **Save Permissions Mapping**.

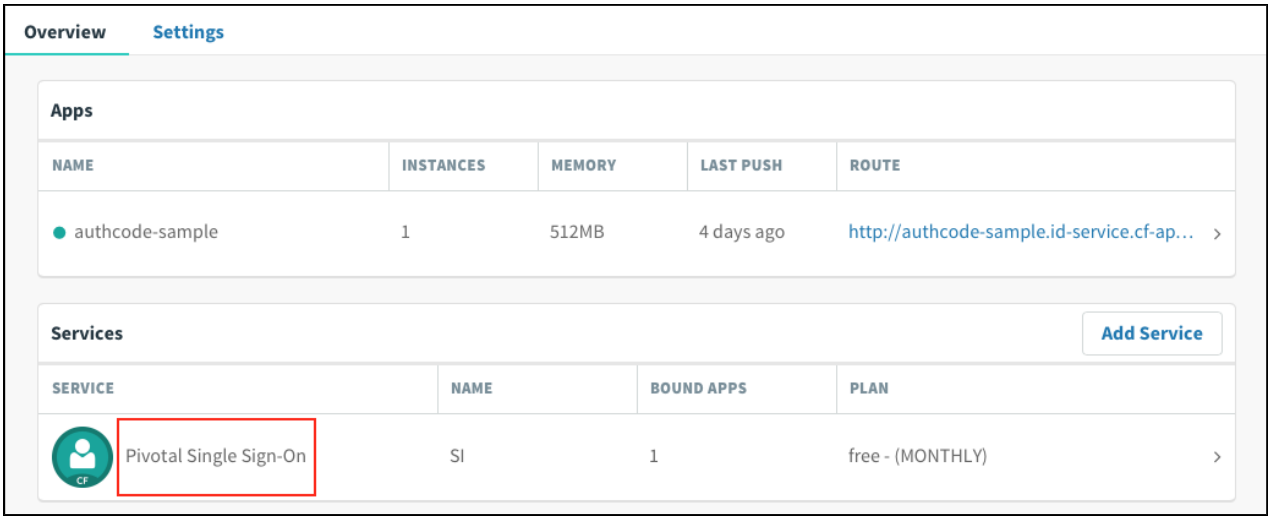
Testing

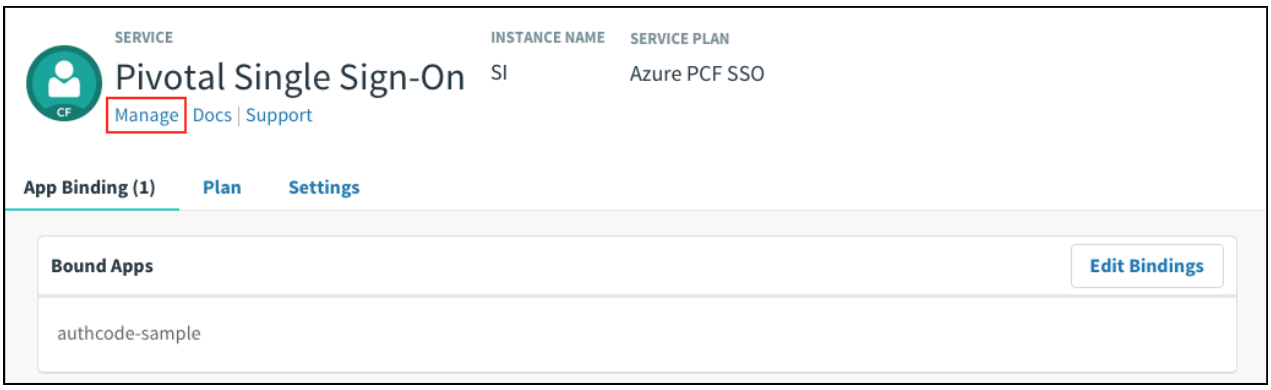
This topic describes how an administrator can test the connection between SSO and Azure Active Directory. An administrator can test both service provider and identity provider connections.

Test Your Service Provider Connection

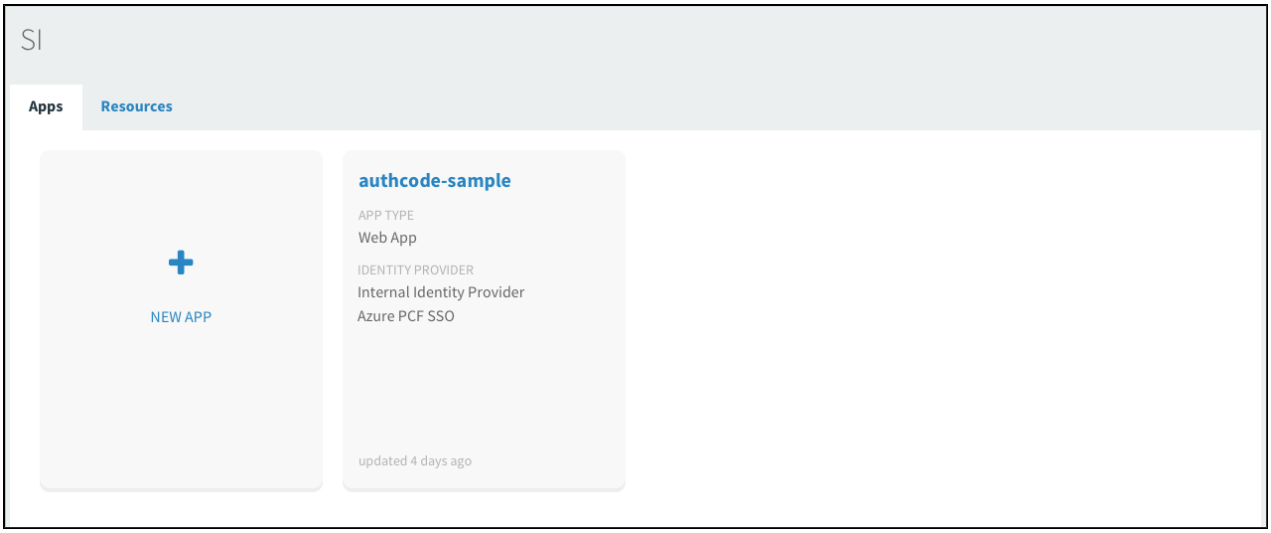
1. Log in to Apps Manager at `https://apps.YOUR-SYSTEM-DOMAIN` and navigate to the organization and space where your application is located.

2. Under **Services**, locate the service instance of the Single Sign-On (SSO) plan bound to your application. Click on the service instance and click **Manage**.

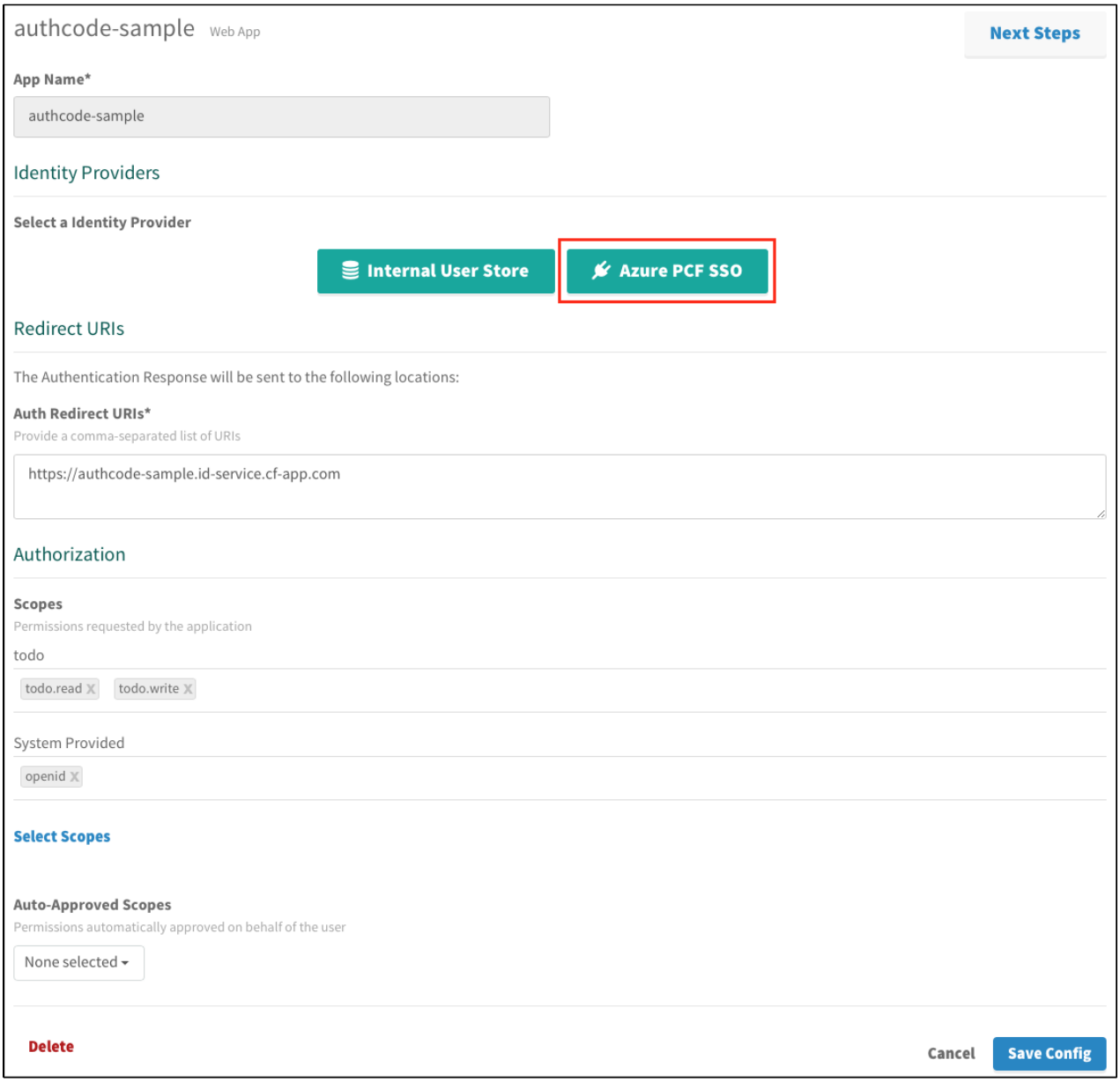




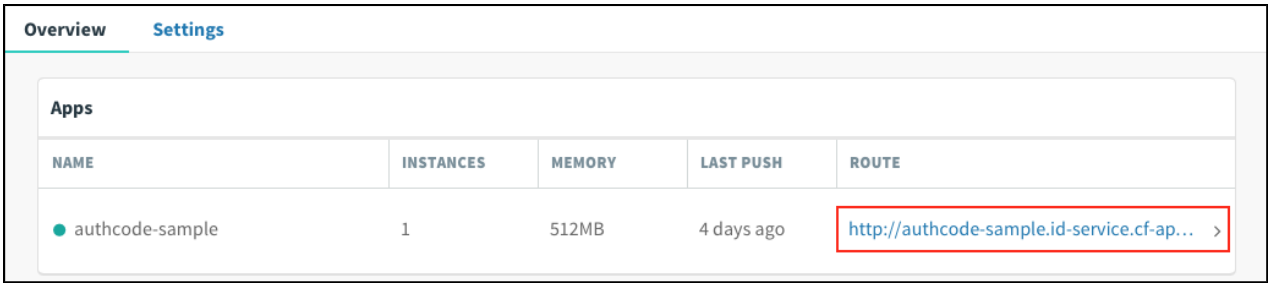
3. Under the **Apps** tab, click your application.



4. Under **Identity Providers**, select the Azure AD identity provider.

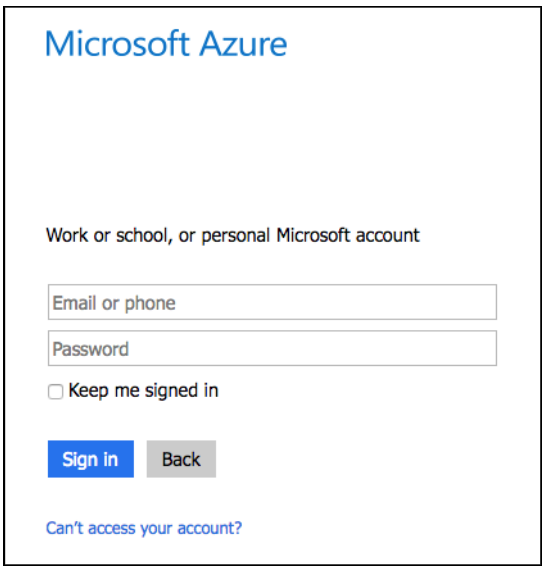


5. Return to Apps Manager and click on the URL below your application to be redirected to the identity provider to authenticate.

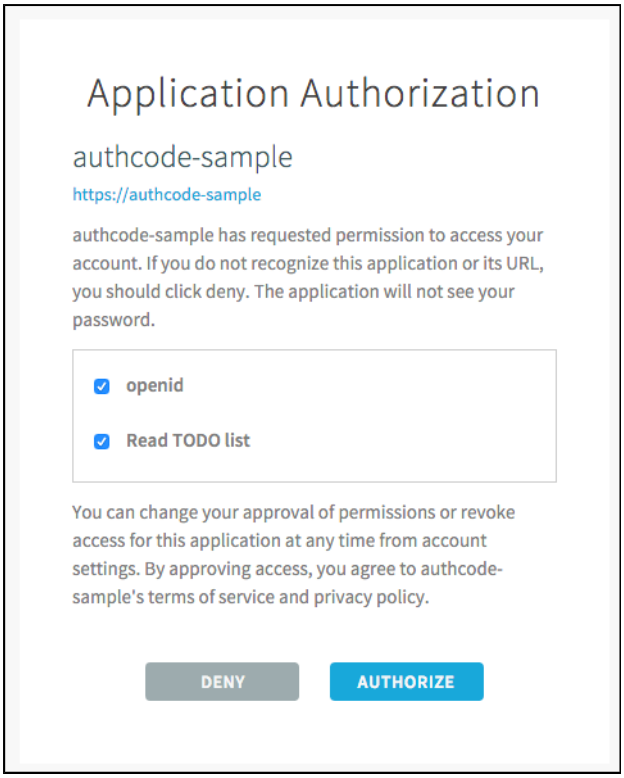


6. Click the link.

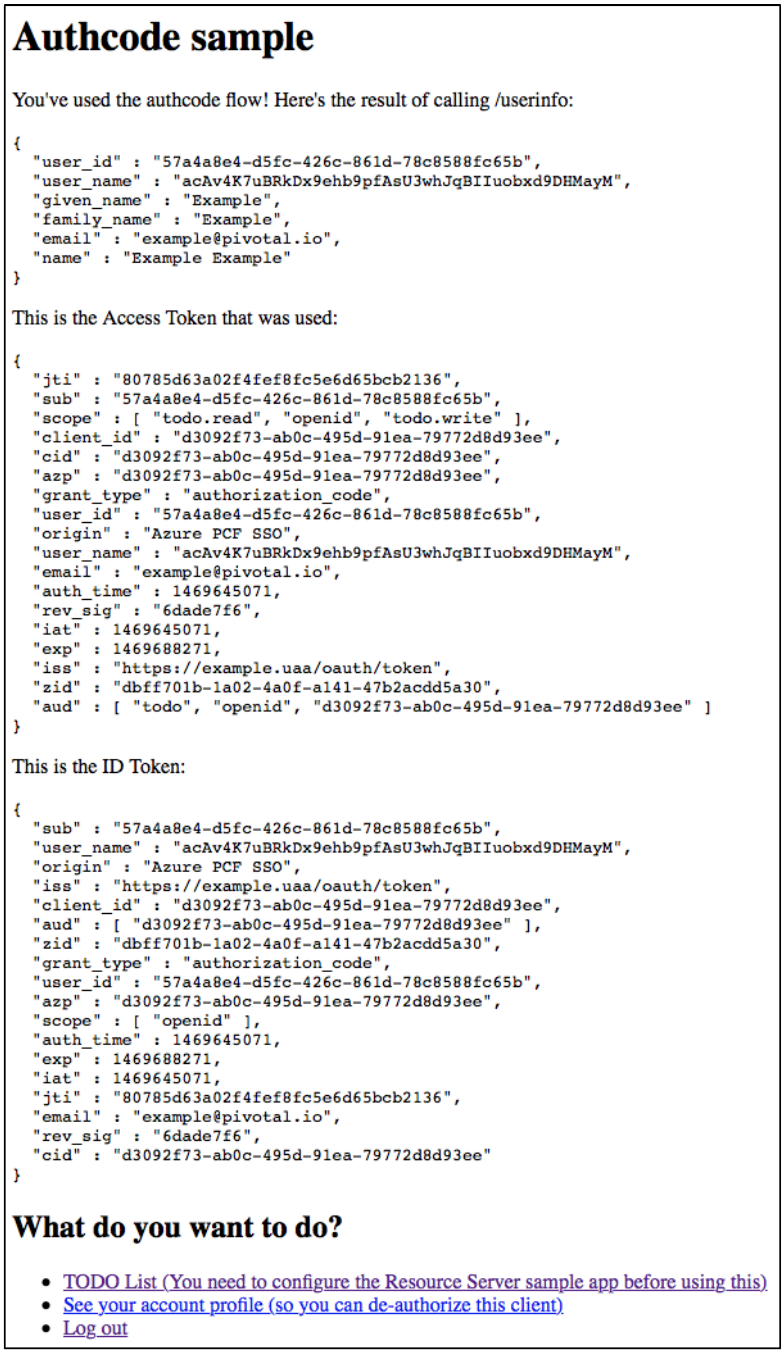
7. On the identity provider sign-in page, enter your credentials and click **Sign In**.



8. The application asks for authorization to the necessary scopes. Click **Authorize**.



9. The access token and ID token displays.



Authcode sample

You've used the authcode flow! Here's the result of calling /userinfo:


```
{
  "user_id" : "57a4a8e4-d5fc-426c-861d-78c8588fc65b",
  "user_name" : "acAv4K7uBRkDx9ehb9pfAsU3whJqBIIuobxd9DHMayM",
  "given_name" : "Example",
  "family_name" : "Example",
  "email" : "example@pivotal.io",
  "name" : "Example Example"
}
```


This is the Access Token that was used:


```
{
  "jti" : "80785d63a02f4fef8fc5e6d65bcb2136",
  "sub" : "57a4a8e4-d5fc-426c-861d-78c8588fc65b",
  "scope" : [ "todo.read", "openid", "todo.write" ],
  "client_id" : "d3092f73-ab0c-495d-91ea-79772d8d93ee",
  "cid" : "d3092f73-ab0c-495d-91ea-79772d8d93ee",
  "azp" : "d3092f73-ab0c-495d-91ea-79772d8d93ee",
  "grant_type" : "authorization_code",
  "user_id" : "57a4a8e4-d5fc-426c-861d-78c8588fc65b",
  "origin" : "Azure PCF SSO",
  "user_name" : "acAv4K7uBRkDx9ehb9pfAsU3whJqBIIuobxd9DHMayM",
  "email" : "example@pivotal.io",
  "auth_time" : 1469645071,
  "rev_sig" : "6dade7f6",
  "iat" : 1469645071,
  "exp" : 1469688271,
  "iss" : "https://example.uaa/oauth/token",
  "zid" : "dbff701b-1a02-4a0f-a141-47b2acdd5a30",
  "aud" : [ "todo", "openid", "d3092f73-ab0c-495d-91ea-79772d8d93ee" ]
}
```


This is the ID Token:


```
{
  "sub" : "57a4a8e4-d5fc-426c-861d-78c8588fc65b",
  "user_name" : "acAv4K7uBRkDx9ehb9pfAsU3whJqBIIuobxd9DHMayM",
  "origin" : "Azure PCF SSO",
  "iss" : "https://example.uaa/oauth/token",
  "client_id" : "d3092f73-ab0c-495d-91ea-79772d8d93ee",
  "aud" : [ "d3092f73-ab0c-495d-91ea-79772d8d93ee" ],
  "zid" : "dbff701b-1a02-4a0f-a141-47b2acdd5a30",
  "grant_type" : "authorization_code",
  "user_id" : "57a4a8e4-d5fc-426c-861d-78c8588fc65b",
  "azp" : "d3092f73-ab0c-495d-91ea-79772d8d93ee",
  "scope" : [ "openid" ],
  "auth_time" : 1469645071,
  "exp" : 1469688271,
  "iat" : 1469645071,
  "jti" : "80785d63a02f4fef8fc5e6d65bcb2136",
  "email" : "example@pivotal.io",
  "rev_sig" : "6dade7f6",
  "cid" : "d3092f73-ab0c-495d-91ea-79772d8d93ee"
}
```

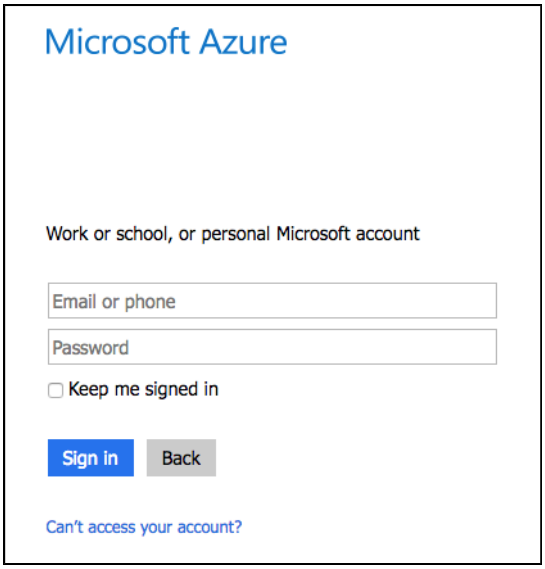

What do you want to do?

- TODO List (You need to configure the Resource Server sample app before using this)
- See your account profile (so you can de-authorize this client)
- Log out

Test Your Identity Provider Connection

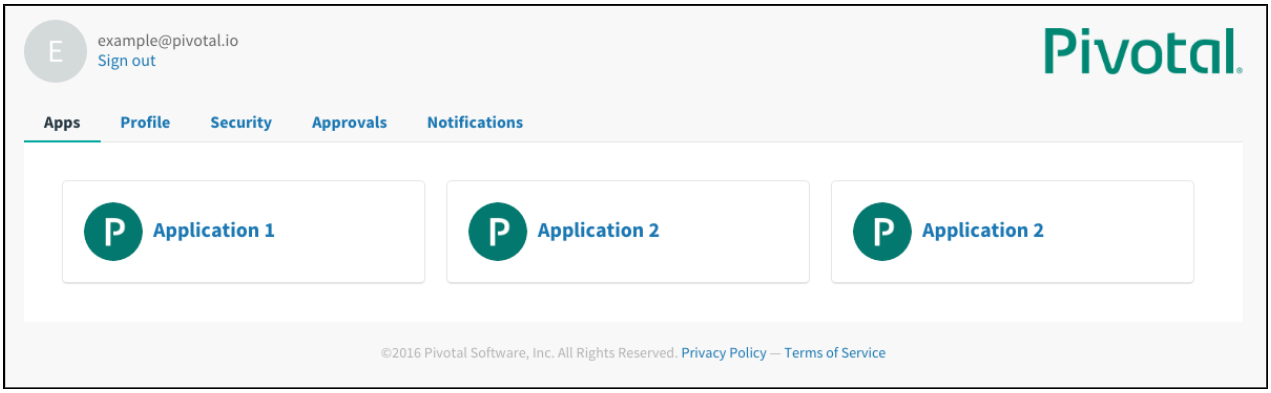
Note: SSO does not support identity provider-initiated flow into applications, but it does redirect the user to the User Account and Authentication (UAA) page to select applications assigned to the user.

1. Sign in to Azure AD.



2. Navigate to your application and click it.

3. You are redirected to the page that lists applications you have access to.

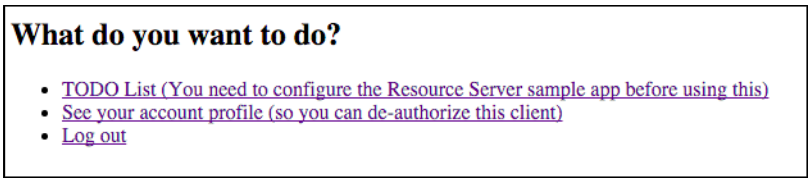


Test Your Single Sign-Off

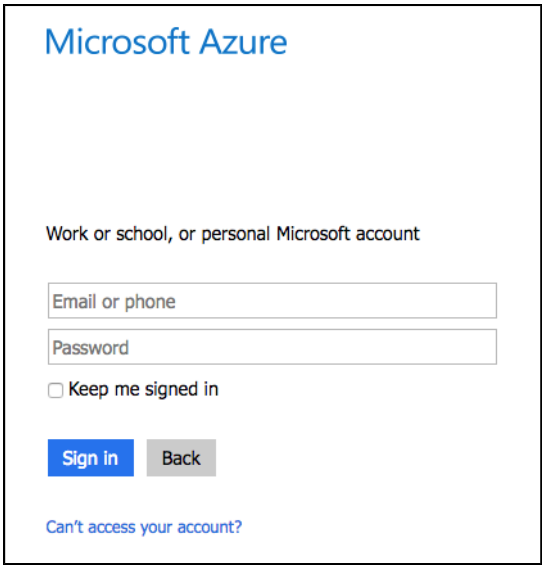
Test single sign-off to ensure that when users log out of the application, they are logged out of Azure AD as well.

1. Sign into the sample application. Information about the access and ID token displays, as well as the "What do you want to do?" section.

2. Under "What do you want to do?", click **Log out**.



3. You are logged out and redirected to the Azure AD login page.



Troubleshooting

This topic describes how to resolve common errors that arise when configuring a single sign-on partnership between Azure Active Directory and Pivotal Single Sign-On (SSO).

App ID Not Found

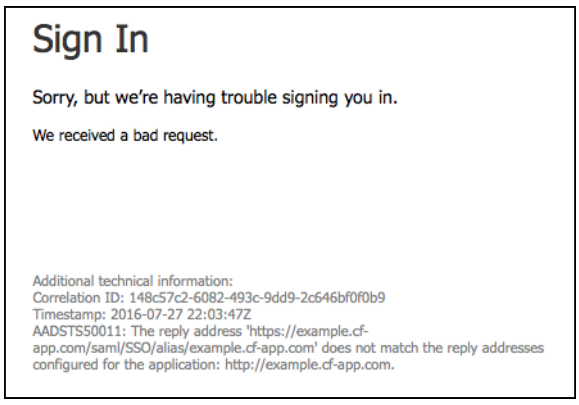
Symptom:

Explanations:

- The App ID URI is misconfigured on Azure AD.

Reply URL Does Not Match

Symptom:



Explanation:

- The Reply URL is misconfigured on Azure AD.

Missing Name ID

Symptom:

Explanation:

- The identity provider metadata has the `RoleDescriptor` elements or is missing configurations for Name ID. See Configure Identity Provider Metadata.

CA Single Sign-On Integration Guide Overview

CA Single Sign-On (formally known as CA SiteMinder) is a Web Access Management system that supports advanced authentication, risk-based security policies, and federated identities. This documentation describes how to configure a single sign-on partnership between CA Single Sign-On as the Identity Provider (IdP) and the Single Sign-On Service (SSO) for Pivotal Cloud Foundry as the Service Provider (SP).

SSO supports service provider-initiated authentication flow and single logout. It does not support identity provider-initiated authentication flow. All SSO communication takes place over SSL.

Prerequisites

To integrate CA Single Sign-On with Pivotal Cloud Foundry (PCF), you need the following:

Pivotal

- PCF, version 1.7.0 or later
- Single Sign-On, version 1.1.0 or later

CA Single Sign-On

- CA Single Sign-On 12.52
- A Signed Certificate by a Certificate Authority

Note: To configure SAML, you must have the Single Sign-On service broker installed on your PCF deployment. You need to create a plan, grant any plan administrators, and specify any organizations this plan should be the authentication authority for. For help configuring plans, see the Manage Service Plans topic..

CA Single Sign-On Integration Guide

Configuring CA Single Sign-On with SSO

Complete both steps below to integrate your deployment with CA Single Sign-On and SSO.

1. Configure CA Single Sign-On as an Identity Provider
2. Configure a Single Sign-On Service Provider

Testing and Troubleshooting

- Testing
- Troubleshooting

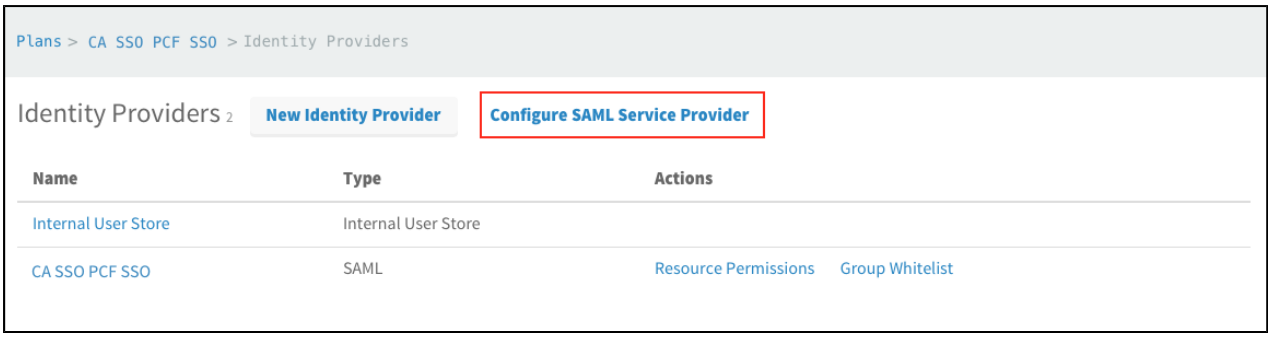
Configure CA Single Sign-On as an Identity Provider

This topic describes how to set up CA Single Sign-On as your identity provider by configuring SAML integration in both Pivotal Cloud Foundry (PCF) and CA Single Sign-On.

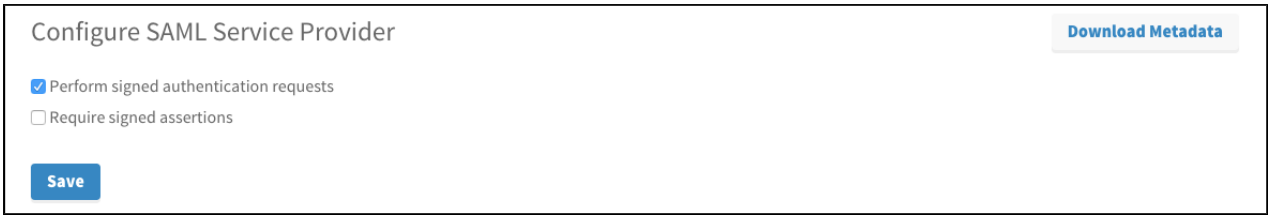
Set up SAML in PCF

1. Log in to the Single Sign-On (SSO) dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.
2. Select your plan and click **Manage Identity Providers** on the dropdown menu.

3. Click **Configure SAML Service Provider**.



4. (Optional) Select **Perform signed authentication requests** to enforce SSO private key signature and identity provider validation.



5. (Optional) Select **Require signed assertions** to validate the origin of signed responses.
6. Click **Download Metadata** to download the service provider metadata.
7. Click **Save**.

Set up SAML in CA Single Sign-On

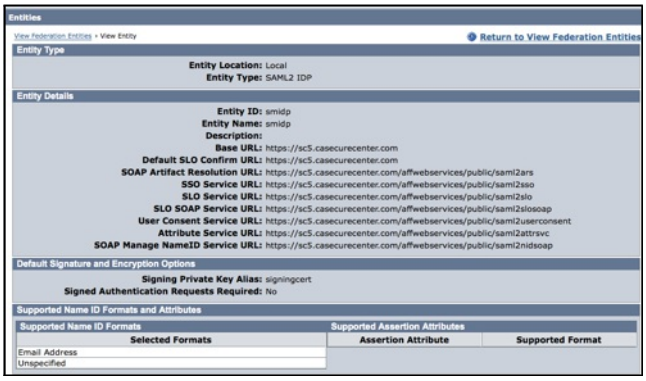
1. Sign in as a CA Single Sign-On administrator.
2. Click the **Federation** tab.
3. Click on the **Entities** link.
4. Click the **Create Entity** button and perform the following steps:
    a. Select **Local** for **Entity Location**.
    b. Select **SAML2 IDP** for **New Entity Type**.

c. Click the **Next** button.

5. In the **Entities** section, perform the following steps:

   a. Enter an **Entity ID**.
   b. Enter an **Entity Name**.
   c. Enter a **Description**.
   d. Enter the fully-qualified domain name for your CA Single Sign-On as the **Base URL**.
   e. Select or import a **Signing Private Key Alias**.
   f. Select a **Name ID format**.
   g. Click the **Next** button.

6. Confirm the Entity Details and click the **Finish** button.



7. Click the **Federation** tab.

8. Click on the **Entities** link.

9. Click the **Import Metadata** button and perform the following steps:

   a. Click **Browse** and select the downloaded metadata for **Metadata file**.
   b. Select **Remote Entity** for **Import As**.
   c. Select **Create New** for **Operation**.
   d. Click the **Next** button.

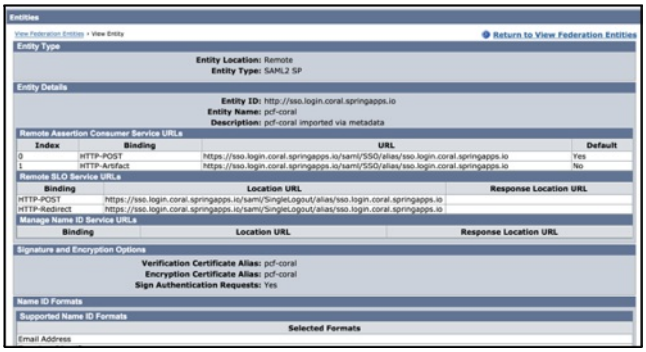
10. In the **Select Entity Defined in Metadata File** section, perform the following steps:

    a. Enter an **Entity Name**.
    b. Click the **Next** button.

11. In the **Select Key Entries to Import** section, perform the following steps:

    a. Enter an **Alias**.
    b. Click the **Next** button.

12. Confirm the Entity Details and click the **Finish** button.

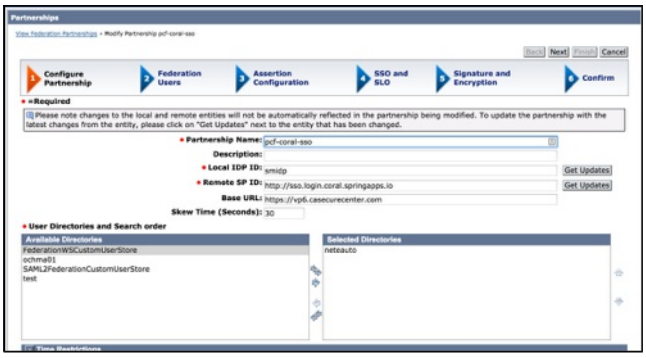


13. Click on the **Federation** tab.

14. Click **Create Partnership** and select **SAML2 IDP -> SP**.

15. In the **Configure Partnership** section, perform the following steps:

a. Enter a **Partnership Name**.
b. Enter a **Description**.
c. Select a previously created local entity for **Local IDP**.
d. Select a previously created remote entity for **Remote SP**.
e. Enter a **Skew Time**.
f. Add any **User Directories**.
g. Click the **Next** button.



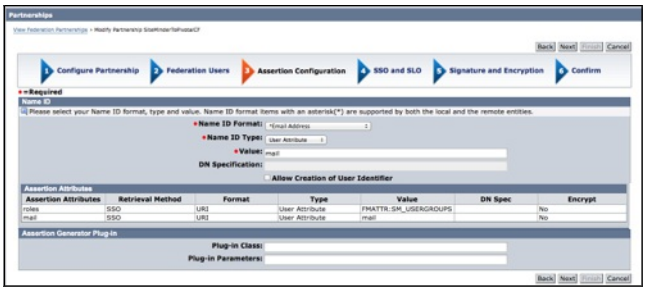
16. Configure **Federation Users** by adding the users you want to include in the partnership and click **Next**.

17. In the **Assertion Configuration** section, perform the following steps:

a. Select a **Name ID Format**.
b. Select **User Attribute** as the **Name ID Type**.
c. Enter `mail` as the **Value**.
d. (Optional) Under **Assertion Attributes**, specify any application or group attributes that you want to map to users in the ID token.

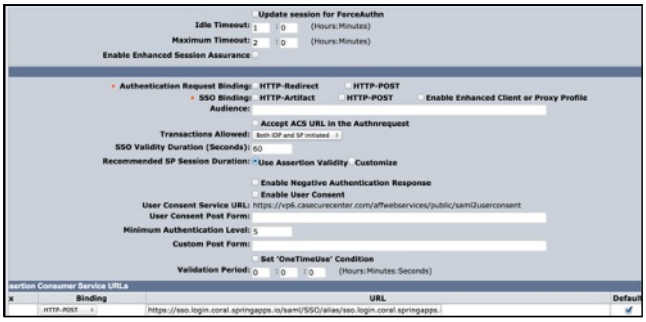
> **Note:** The value for sending a user's groups is **FMATTR:SM_USERGROUPS**.

e. Click the **Next** button.



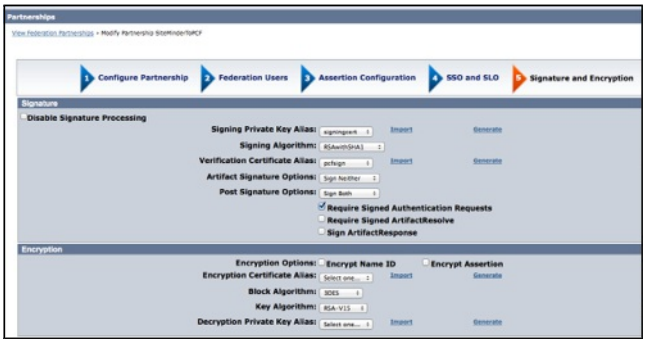
18. In the **SSO and SLO** section, perform the following steps:

a. Enter the **Authentication URL**.
b. Select **HTTP-Post** for **SSO Binding**.
c. Select **Both IDP and SP initiated** for **Transactions Allowed**.
d. Click the **Next** button.



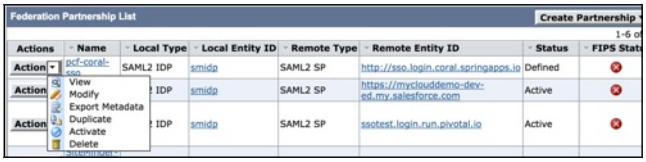
19. In the **Signature and Encryption** section, perform the following steps:

    a. Select your key alias for **Signing Private Key Alias**.
    b. Select your certificate alias for **Verification Certificate Alias**.
    c. Click the **Next** button.



20. Confirm the Partnership Details and click the **Finish** button.

21. Click the **Action** button and click **Activate**.



22. Click the **Action** button and click **Export Metadata**.

Configure a Single Sign-On Service Provider

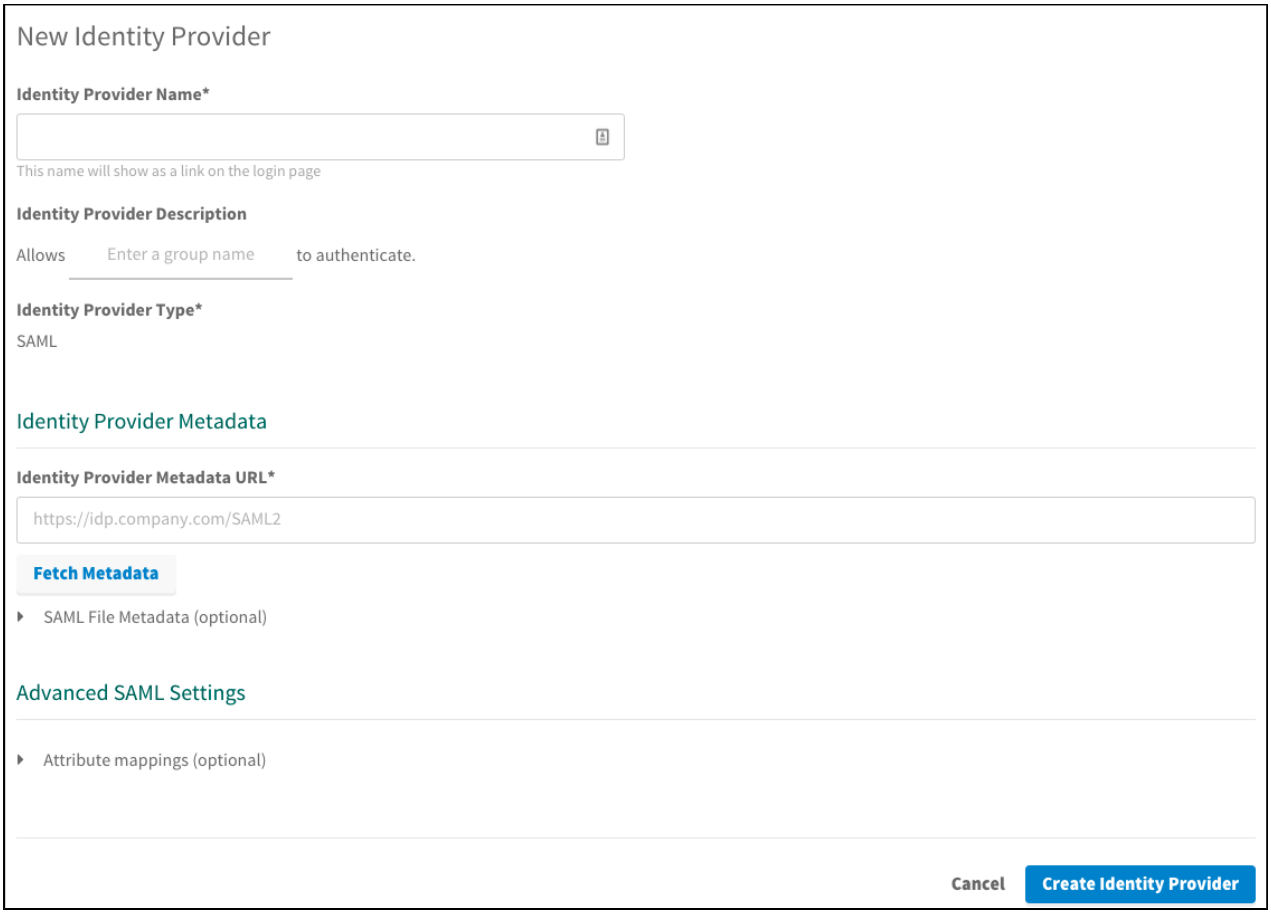
This topic describes how to add an external identity provider to your Pivotal Single Sign-On (SSO) service plan.

Setting up SAML

1. Log in to the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.

2. Select your plan and click **Manage Identity Providers** on the dropdown menu.

3. Click **New Identity Provider** to create a new identity provider.



4. To create a new identity provider, perform the following steps:

    a. Enter an identity provider name in **Identity Provider Name**.
    b. (Optional) Enter a description in **Identity Provider Description**.
    c. Click **SAML File Metadata (optional)** followed by clicking the **Upload Identity Provider Metadata** button to upload your metadata XML.
    d. (Optional) Under **Advanced SAML Settings**, click **Attribute Mappings** to enter the mappings.

5. Click **Create Identity Provider**.

6. Click **Resource Permissions**.

7. Click **New Permissions Mapping** and perform the following steps:

    a. Enter a **Group Name**.
    b. For **Select Permissions**, select the permissions that the members of the group from the external identity provider should have access to.

8. Navigate to the identity provider list.

9. Click **Group Whitelist** and enter the group names from the external identity provider that should be propagated in the ID token.

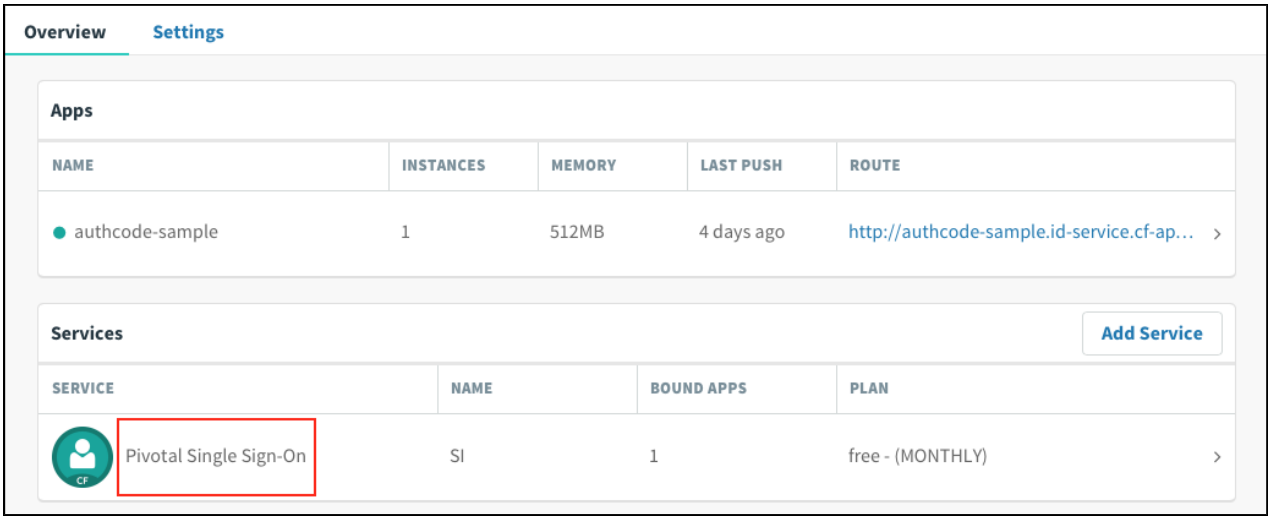
Testing

This topic describes how an administrator can test the connection between SSO and CA Single Sign-On. An administrator can test both service provider and identity provider connections.

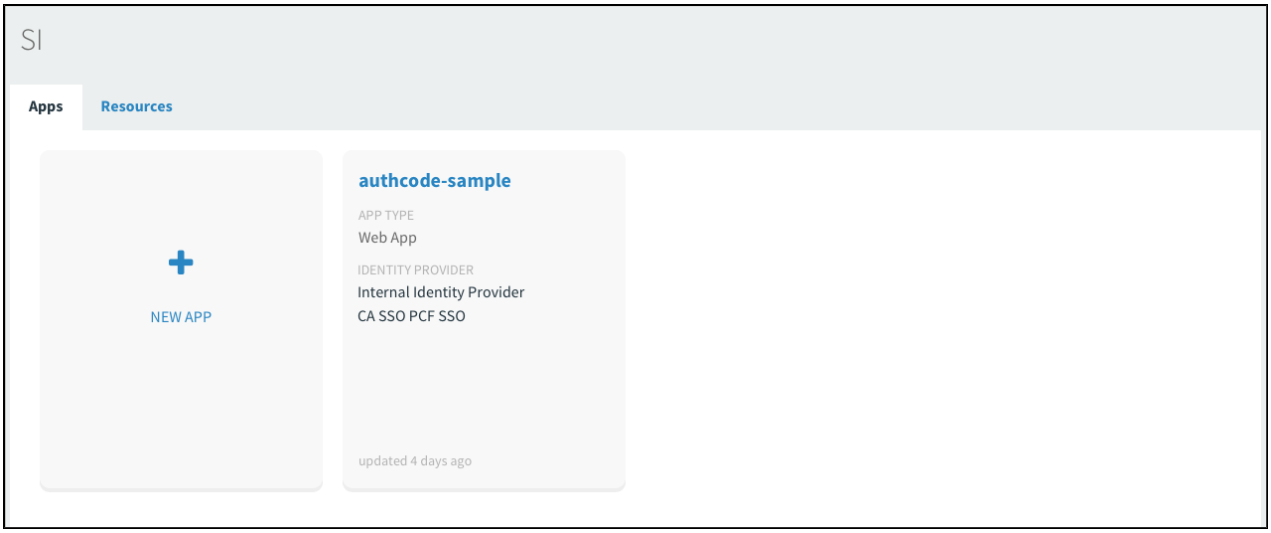
Test Your Service Provider Connection

1. Log in to Apps Manager at `https://apps.YOUR-SYSTEM-DOMAIN` and navigate to the organization and space where your application is located.

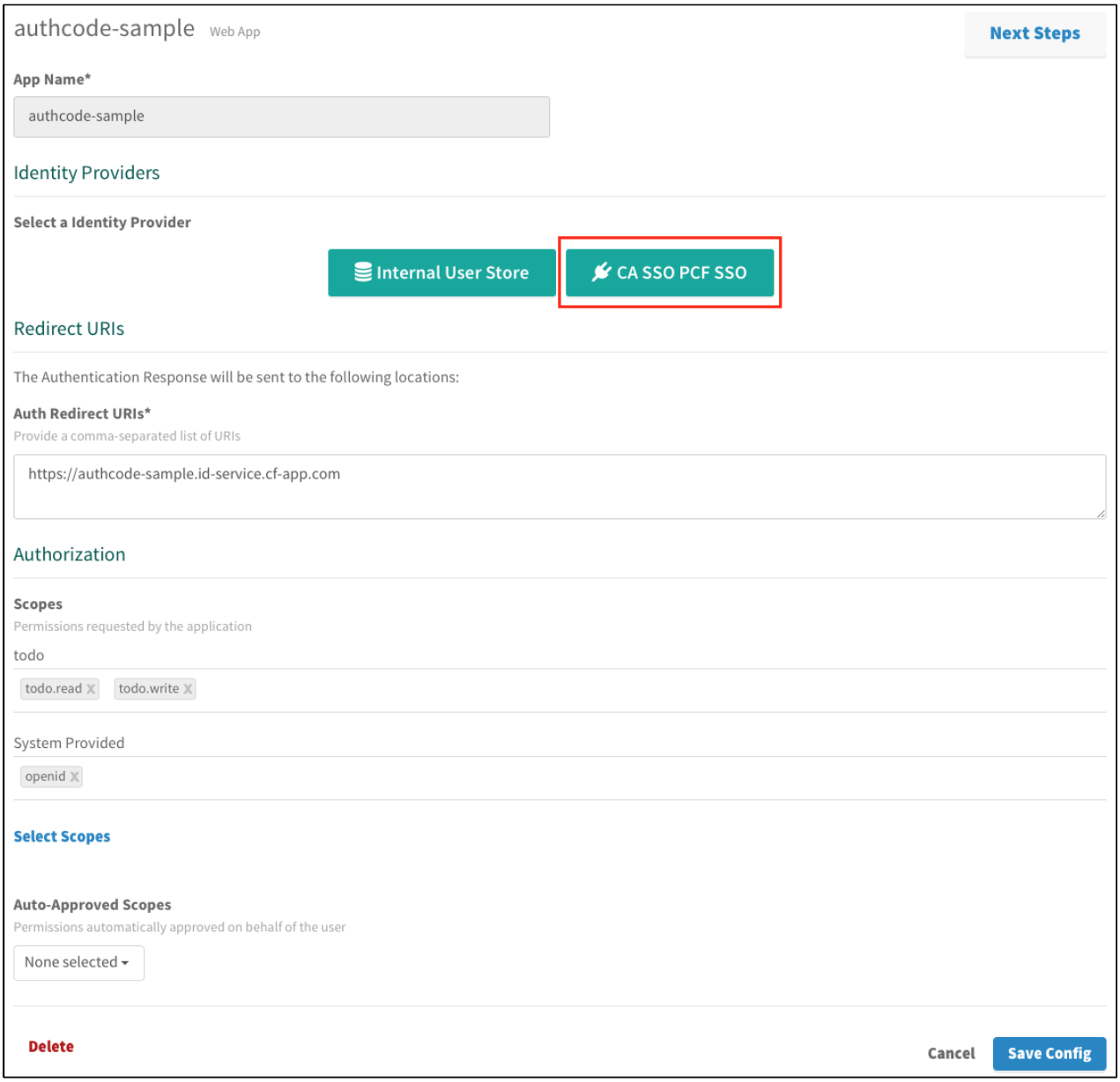
2. Under **Services**, locate the service instance of the Single Sign-On (SSO) plan bound to your application. Select the service instance and click **Manage**.



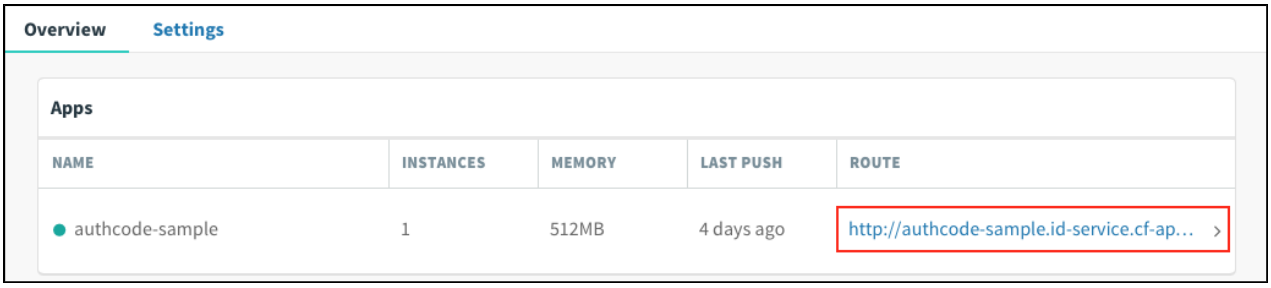
3. Under the **Apps** tab, click your application.



4. Under **Identity Providers**, select the CA Single Sign-On identity provider.

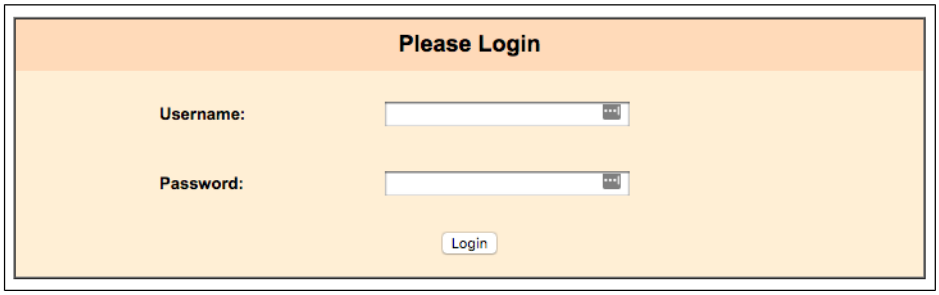


5. Return to Apps Manager and click on the URL below your application to be redirected to the identity provider to authenticate.

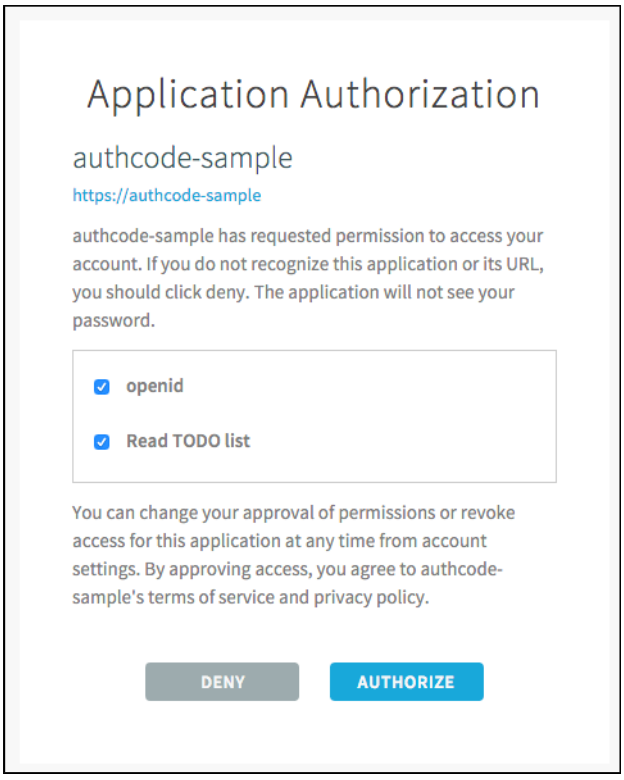


6. Click the link.

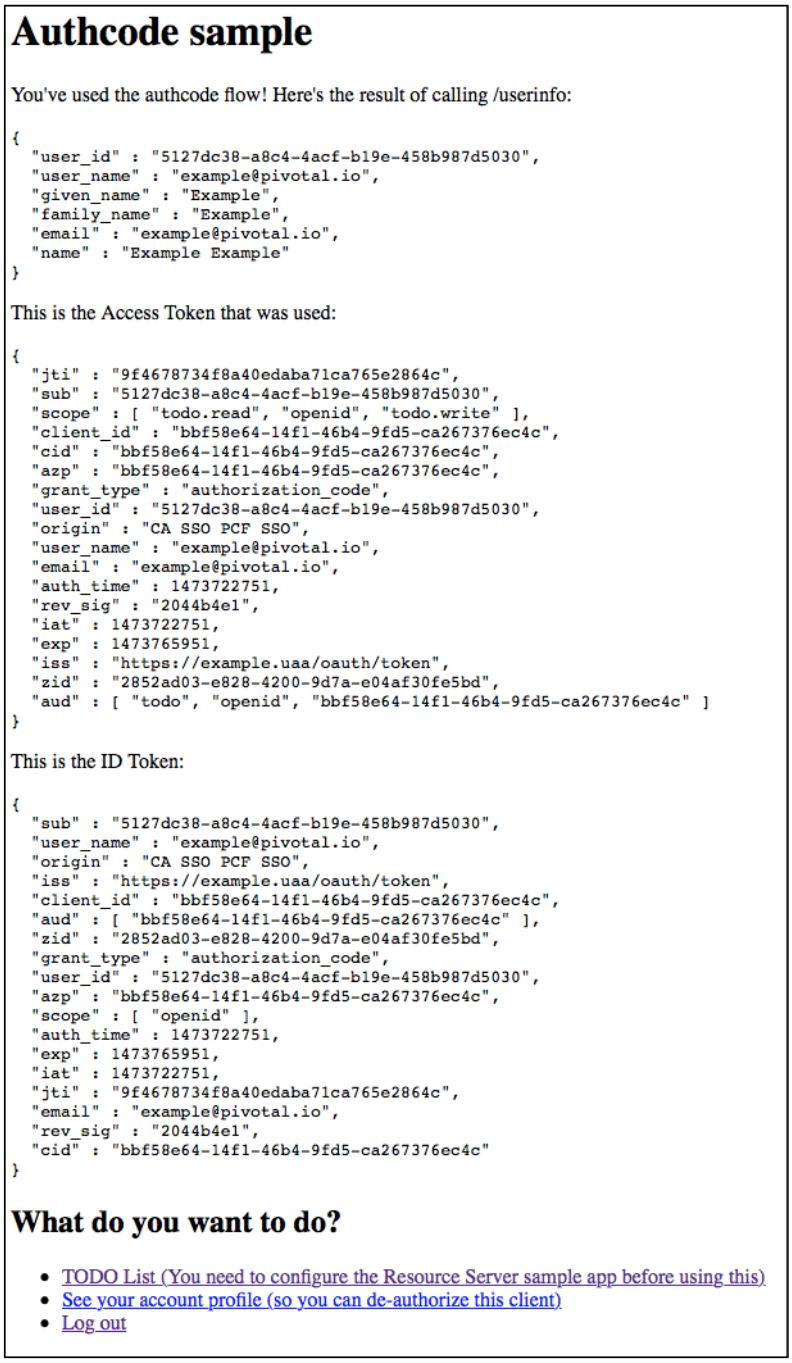
7. On the identity provider sign-in page, enter your credentials and click **Sign On**.



8. The application asks for authorization to the necessary scopes. Click **Authorize**.



9. The access token and ID token displays.



Authcode sample

You've used the authcode flow! Here's the result of calling /userinfo:


```
{
  "user_id" : "5127dc38-a8c4-4acf-b19e-458b987d5030",
  "user_name" : "example@pivotal.io",
  "given_name" : "Example",
  "family_name" : "Example",
  "email" : "example@pivotal.io",
  "name" : "Example Example"
}
```


This is the Access Token that was used:


```
{
  "jti" : "9f4678734f8a40edaba71ca765e2864c",
  "sub" : "5127dc38-a8c4-4acf-b19e-458b987d5030",
  "scope" : [ "todo.read", "openid", "todo.write" ],
  "client_id" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c",
  "cid" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c",
  "azp" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c",
  "grant_type" : "authorization_code",
  "user_id" : "5127dc38-a8c4-4acf-b19e-458b987d5030",
  "origin" : "CA SSO PCF SSO",
  "user_name" : "example@pivotal.io",
  "email" : "example@pivotal.io",
  "auth_time" : 1473722751,
  "rev_sig" : "2044b4e1",
  "iat" : 1473722751,
  "exp" : 1473765951,
  "iss" : "https://example.uaa/oauth/token",
  "zid" : "2852ad03-e828-4200-9d7a-e04af30fe5bd",
  "aud" : [ "todo", "openid", "bbf58e64-14f1-46b4-9fd5-ca267376ec4c" ]
}
```


This is the ID Token:


```
{
  "sub" : "5127dc38-a8c4-4acf-b19e-458b987d5030",
  "user_name" : "example@pivotal.io",
  "origin" : "CA SSO PCF SSO",
  "iss" : "https://example.uaa/oauth/token",
  "client_id" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c",
  "aud" : [ "bbf58e64-14f1-46b4-9fd5-ca267376ec4c" ],
  "zid" : "2852ad03-e828-4200-9d7a-e04af30fe5bd",
  "grant_type" : "authorization_code",
  "user_id" : "5127dc38-a8c4-4acf-b19e-458b987d5030",
  "azp" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c",
  "scope" : [ "openid" ],
  "auth_time" : 1473722751,
  "exp" : 1473765951,
  "iat" : 1473722751,
  "jti" : "9f4678734f8a40edaba71ca765e2864c",
  "email" : "example@pivotal.io",
  "rev_sig" : "2044b4e1",
  "cid" : "bbf58e64-14f1-46b4-9fd5-ca267376ec4c"
}
```

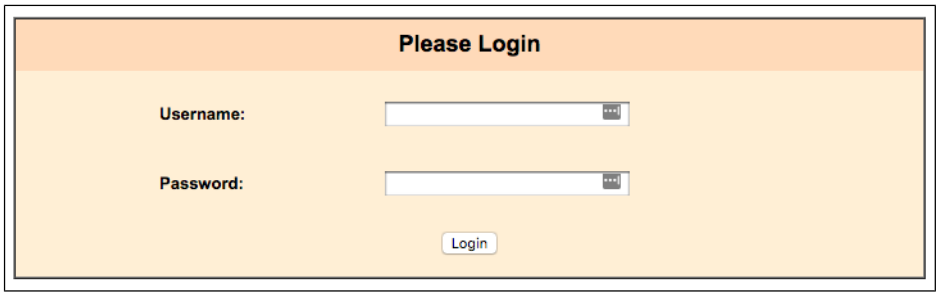

What do you want to do?

- TODO List (You need to configure the Resource Server sample app before using this)
- See your account profile (so you can de-authorize this client)
- Log out

Test Your Identity Provider Connection

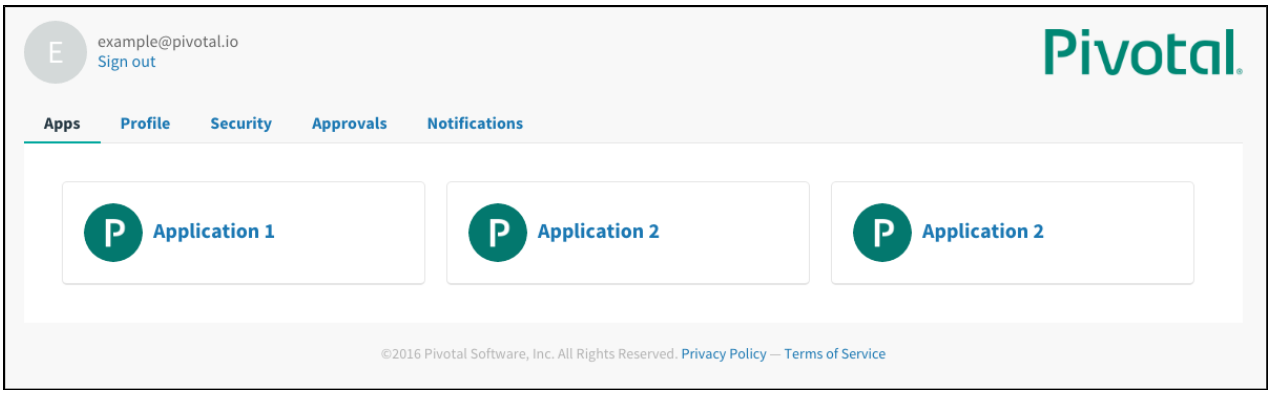
Note: SSO does not support identity provider-initiated flow into applications, but it does redirect the user to the User Account and Authentication (UAA) page to select applications assigned to the user.

1. Sign in to CA Single Sign-On.



2. Navigate to your application and click it.

3. You are redirected to the page that lists applications you have access to.

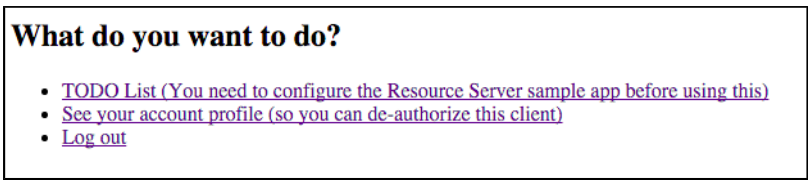


Test Your Single Sign-Off

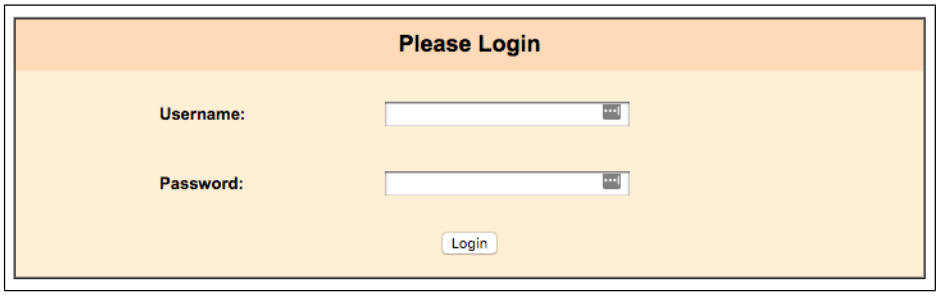
Test single sign-off to ensure that when users log out of the application, they are logged out of CA Single Sign-On as well.

1. Sign in to the sample application. Information about the access and ID token displays, as well as the “What do you want to do?” section.

2. Under “What do you want to do?”, click **Log out**.



3. You are logged out and redirected to the CA Single Sign-On login page.

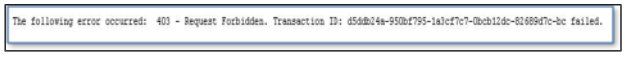


Troubleshooting

This topic describes how to resolve common errors that arise when configuring a single sign-on partnership between PingOne Cloud and Pivotal Single Sign-On (SSO).

CA Single Sign-On Partnership is Inactive

Symptom:

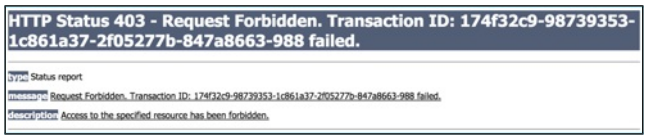


Explanations:

- The CA Single Sign-On is inactive in CA Single Sign-On.

Service Provider Entity ID Misconfigured

Symptom:

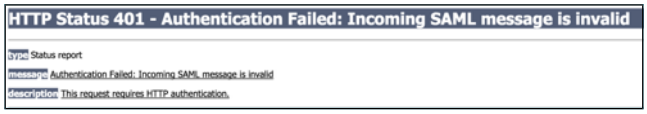


Explanation:

- The service provider Entity ID is misconfigured in CA Single Sign-On.

Incoming SAML message is invalid

Symptom:



Explanation:

- The identity provider Entity ID is misconfigured in CA Single Sign-On or in PCF Single Sign-On.
- The Name ID Format was misconfigured in CA Single Sign-On

Assertion Consumer Service URL Misconfigured

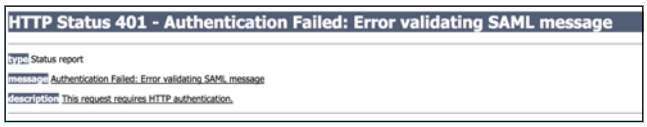
Symptom:

Explanation:

- The service provider Assertion Consumer Service (ACS) is misconfigured in CA Single Sign-On.

Audience Field Misconfigured

Symptom:

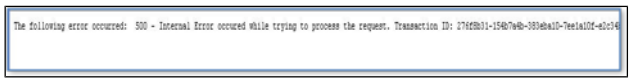


Explanation:

- The service provider Audience Field is misconfigured in CA Single Sign-On.

Expired Certificate

Symptom:



Explanation:

- The certificate has expired in CA Single Sign-On.

Identity Provider SSO URL Misconfigured

Symptom:

Explanation:

- The identity provider SSO URL is misconfigured in PCF Single Sign-On.

Okta Integration Guide Overview

Okta is an enterprise identity management and single sign-on service that integrates with applications in the cloud, on-premises, or on a mobile device. This documentation describes how to configure a single sign-on partnership between Okta as the Identity Provider (IdP) and the Single Sign-On Service (SSO) for Pivotal Cloud Foundry as the Service Provider (SP).

SSO supports service provider-initiated authentication flow and single logout. It does not support identity provider-initiated authentication flow. All SSO communication takes place over SSL.

Prerequisites

To integrate Okta with Pivotal Cloud Foundry (PCF), you need:

Pivotal

- PCF, version 1.7.0 or later.
- Single Sign-On, version 1.1.0 or later.

Okta

- Okta, version 2016.07 or later.
- A user with Application Admin privileges.

Note: To configure SAML, you must have the Single Sign-On service broker installed on your PCF deployment. You need to create a plan, grant any plan administrators, and specify any organizations this plan should be the authentication authority for. For help configuring plans, see the Manage Service Plans topic.

Okta Integration Guide

Configuring Okta with SSO

Complete both steps below to integrate your deployment with Okta and SSO.

1. Configure Okta as an Identity Provider
2. Configure a Single Sign-On Service Provider

Testing and Troubleshooting

- Testing
- Troubleshooting

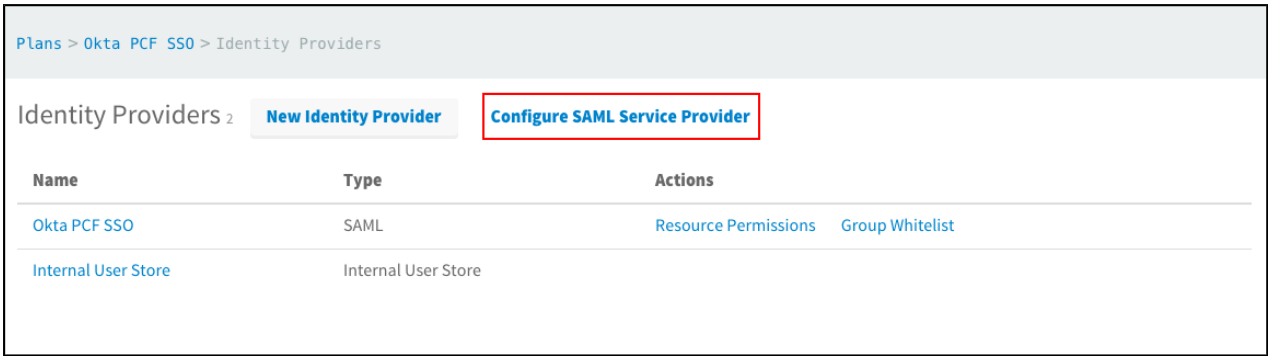
Configure Okta as an Identity Provider

This topic describes how to set up Okta as your identity provider by configuring SAML integration in both Pivotal Cloud Foundry (PCF) and Okta.

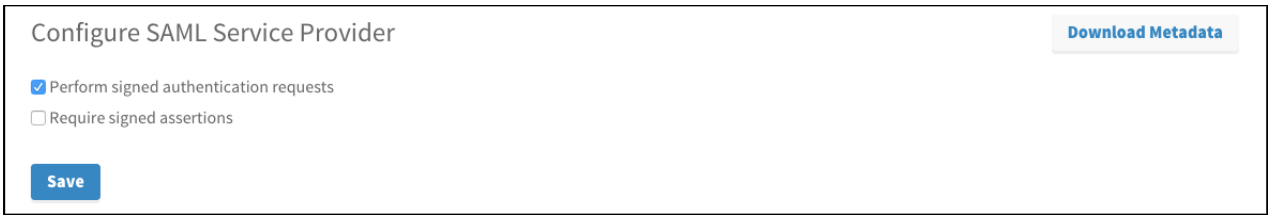
Set up SAML in PCF

1. Log into the Single Sign-On (SSO) dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.
2. Select your plan and click **Manage Identity Providers** on the dropdown menu.

3. Click **Configure SAML Service Provider**.



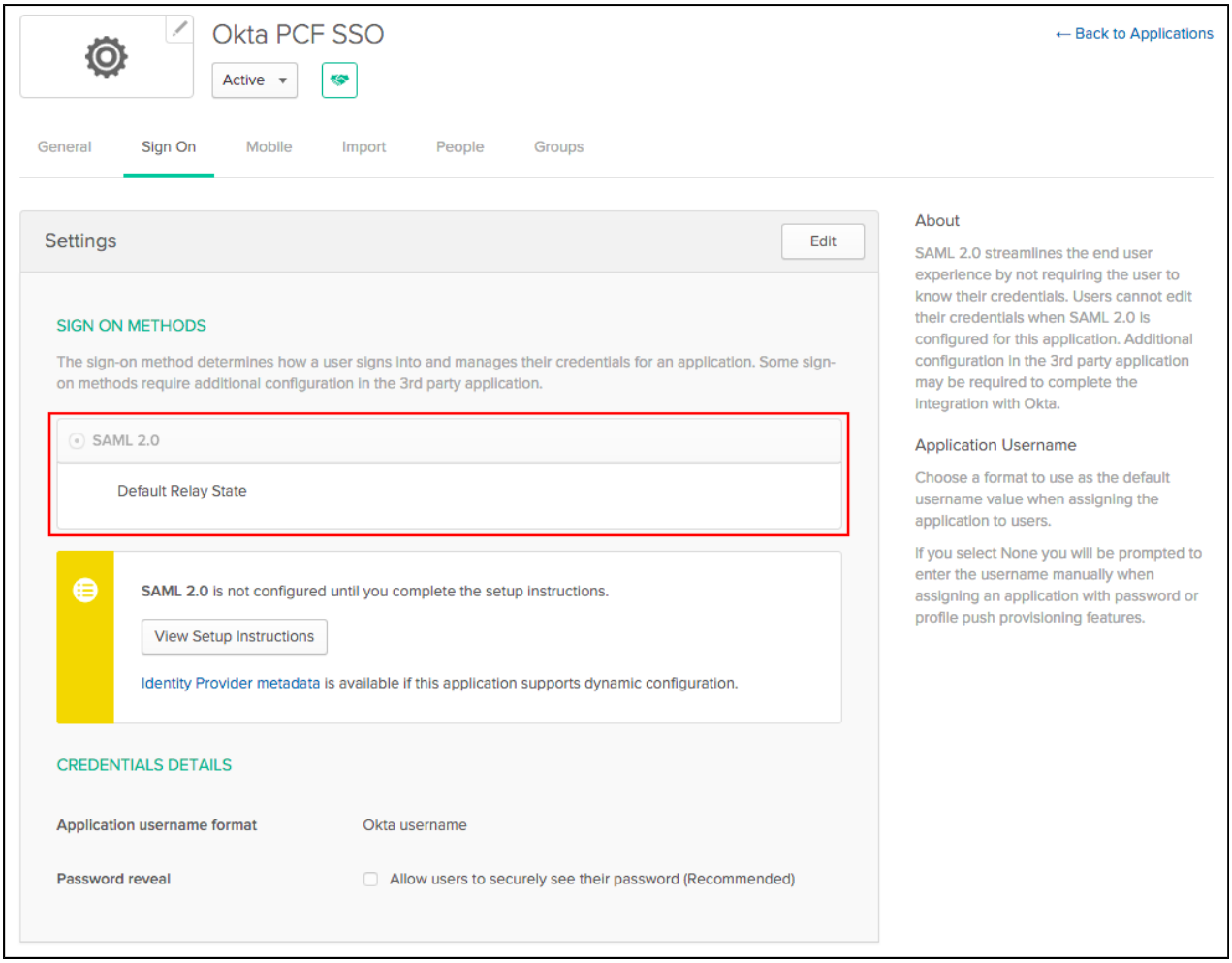
4. (Optional) Select **Perform signed authentication requests** to enforce SSO private key signature and identity provider validation.



5. (Optional) Select **Require signed assertions** to validate the origin of signed responses.
6. Click **Download Metadata** to download the service provider metadata.
7. Click **Save**.

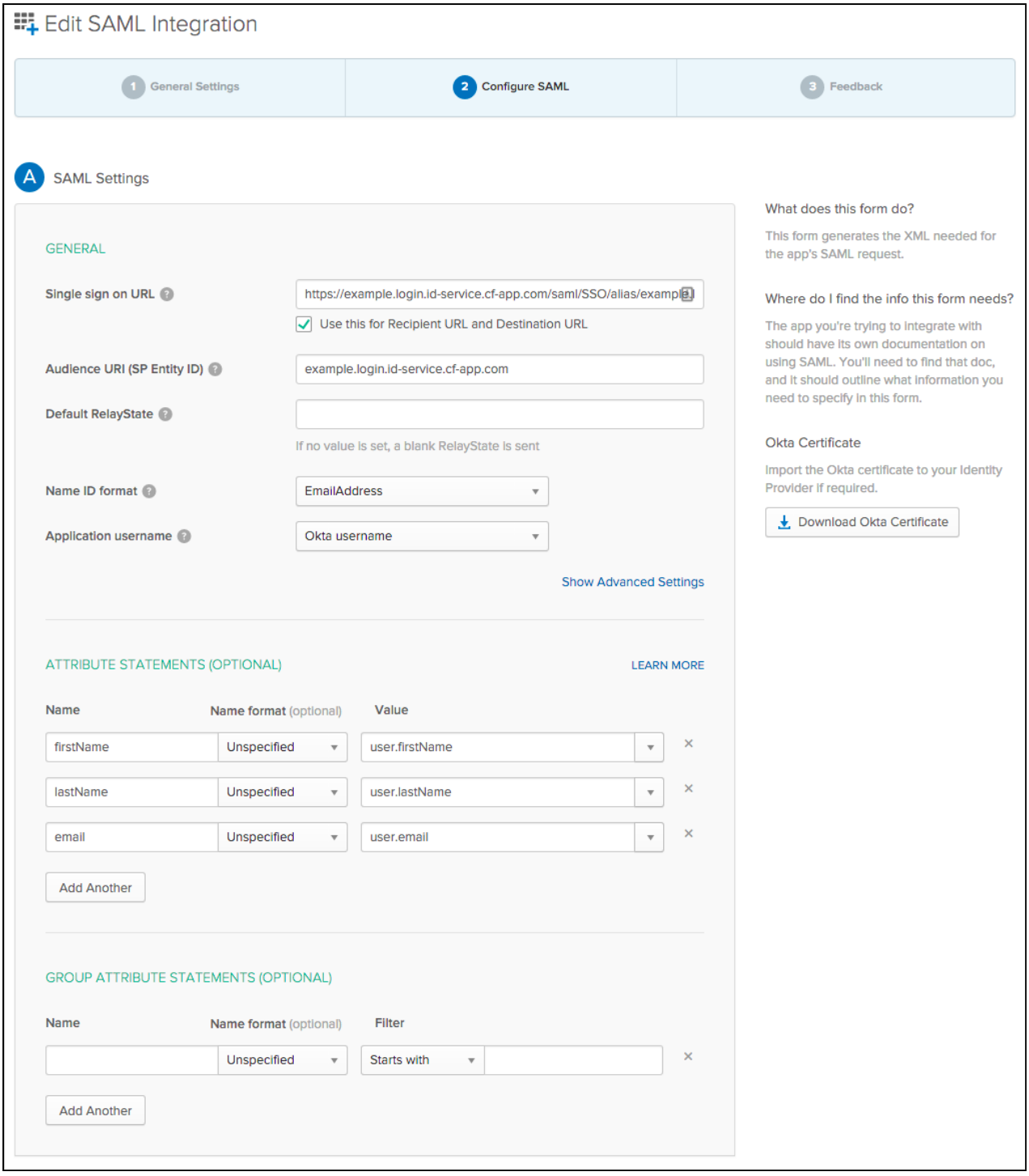
Set up SAML in Okta

1. Sign in as an Okta administrator.
2. Navigate to your application, then click the **Sign On** tab.
3. Under **Settings**, click **Edit**, and select **SAML 2.0**.



4. Click the **General** tab.

5. Under SAML Settings, click the **Edit** button followed by the **Next** button to configure SAML.



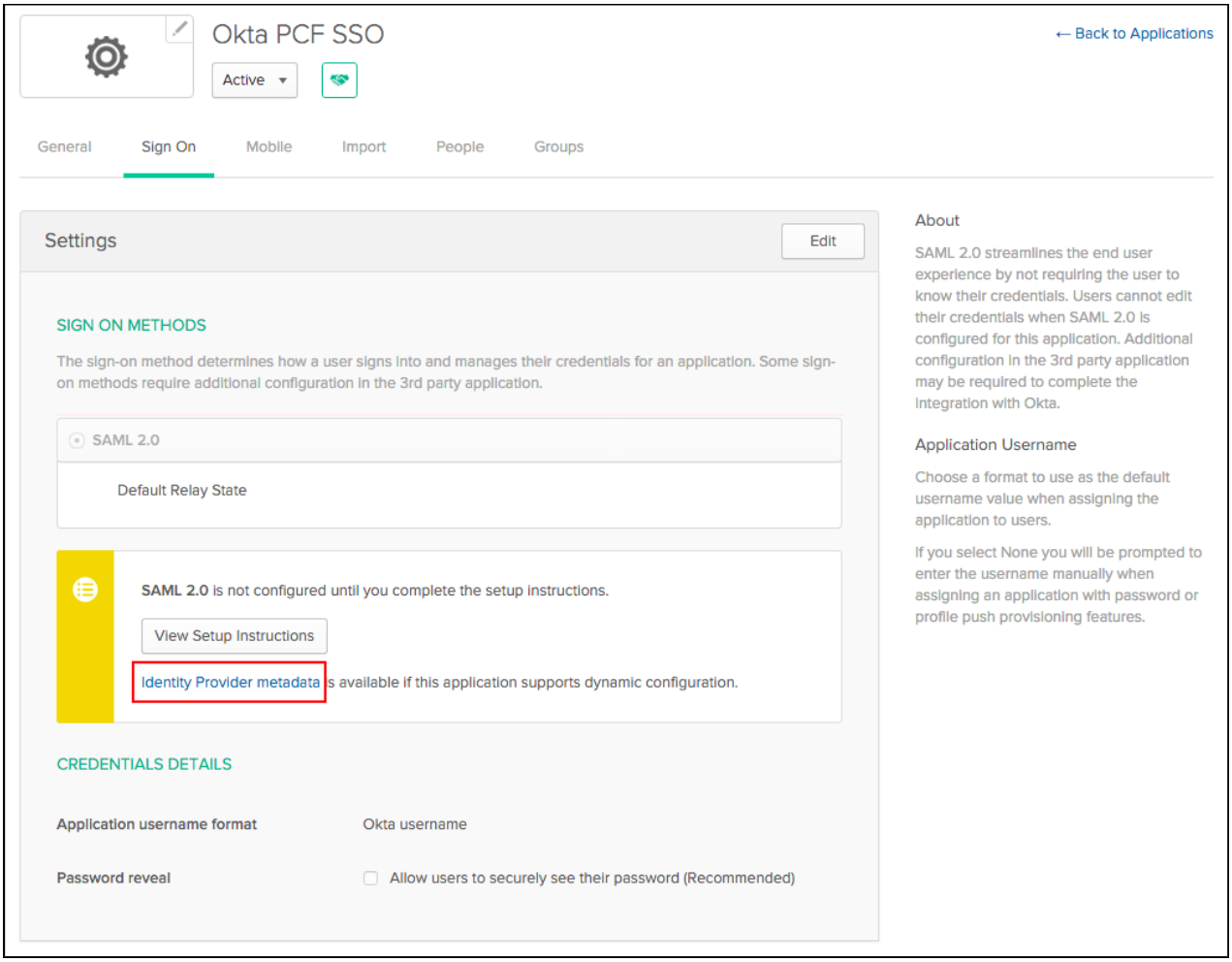
6. In the **SAML Settings** section, perform the following steps:

    a. Enter the `AssertionConsumerService Location URL` from your downloaded service provider metadata into **Single sign on URL**. For example, `https://AUTH-DOMAIN/saml/SSO/alias/AUTH-DOMAIN`.
    b. Enter your Auth Domain URL into **Audience URI (SP Entity ID)**. You can view the Auth Domain for a plan by logging into the SSO dashboard, clicking the name of your plan, and selecting **Edit Plan**. For example, `https://AUTH-DOMAIN.login.SYSTEM-DOMAIN`.
    c. Select a **Name ID format**.
    d. Select an **Application username**.

7. (Optional) To configure single logout, perform the following steps:

    a. Click **Show Advanced Settings**.
    b. For **Enable Single Logout**, select **Allow application** to initiate single logout.
    c. Enter the `SingleLogoutService Location URL` from your downloaded service provider metadata into **Single Logout URL**.
    d. Enter your `Auth Domain URL` into **SP Issuer**.
    e. Click **Upload Signature Certificate** to upload the signature certificate from your downloaded service provider metadata.

8. (Optional) Under **Attribute Statements (Optional)**, specify any attribute statements that you want to map to users in the ID token.
9. (Optional) Under **Group Attribute Statements (Optional)**, specify any group attribute statements that you want to map to users in the ID token. This is a group that users are in within Okta.
10. Click the **Next** button followed by the **Finish** button.
11. Click **Identity Provider metadata** to download the metadata, or copy and save the link address of the **Identity Provider metadata**.



Configure a Single Sign-On Service Provider

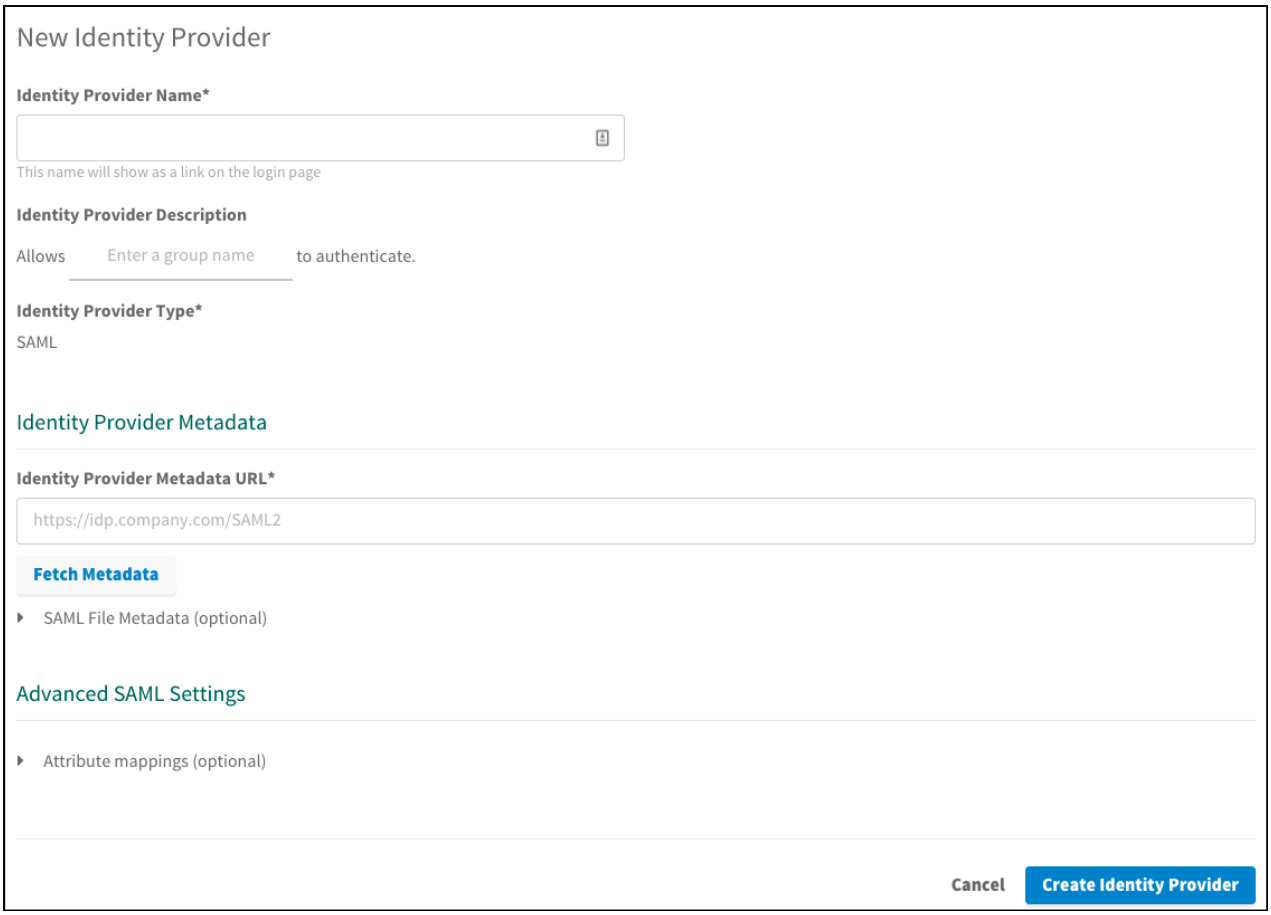
This topic describes how to add an external identity provider to your Pivotal Single Sign-On (SSO) service plan.

Setting up SAML

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.

2. Select your plan and click **Manage Identity Providers** on the dropdown menu.

3. Click **New Identity Provider** to create a new identity provider.



4. To create a new identity provider, perform the following steps:

    a. Enter an identity provider name into **Identity Provider Name**.
    b. (Optional) Enter a description into **Identity Provider Description**.
    c. Specify Identity Provider Metadata from Step 11 of the Configure Okta as an Identity Provider topic.

        i. Option 1: Enter your **Input Identity Provider Metadata URL** and **Fetch Metadata** to fetch your identity provider metadata from an endpoint.
        ii. Option 2: Click **SAML File Metadata (optional)** to upload your metadata XML manually.

    d. (Optional) Under **Advanced SAML Settings**, click **Attribute Mappings** to enter the mappings.

5. Click **Create Identity Provider**.

6. Click **Resource Permissions**.

7. Click **New Permissions Mapping** and perform the following steps:

    a. Enter a **Group Name**.
    b. For **Select Permissions**, select the permissions that the members of the group from the external identity provider should have access to.

8. Navigate to the identity provider list.

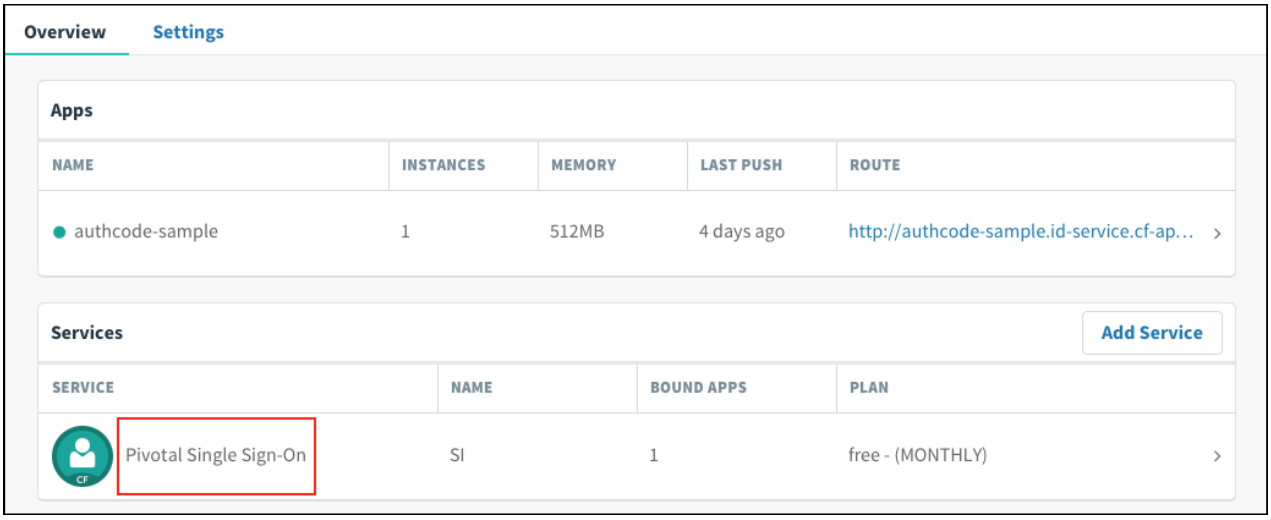
9. Click **Group Whitelist** and enter the group names from the external identity provider that should be propagated in the ID token.

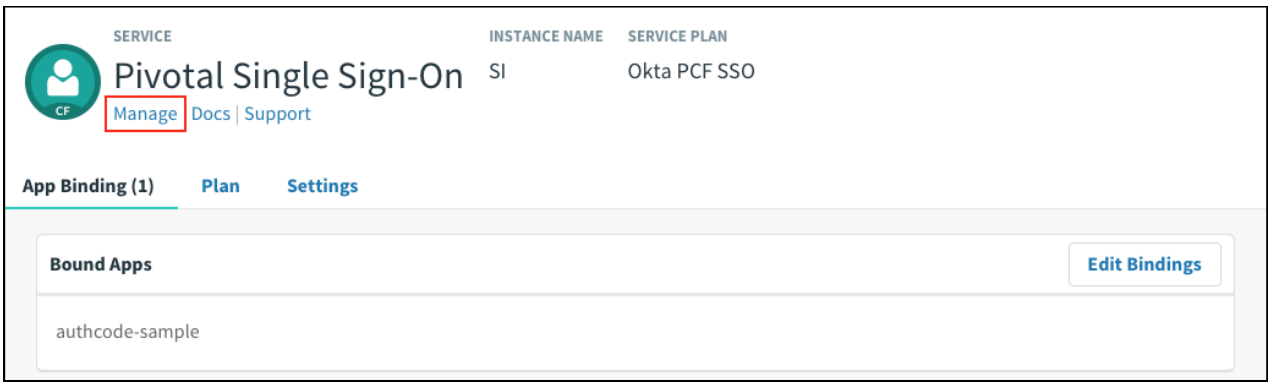
Testing

This topic describes how an administrator can test the connection between SSO and Okta services. An administrator can test both service provider and identity provider connections.

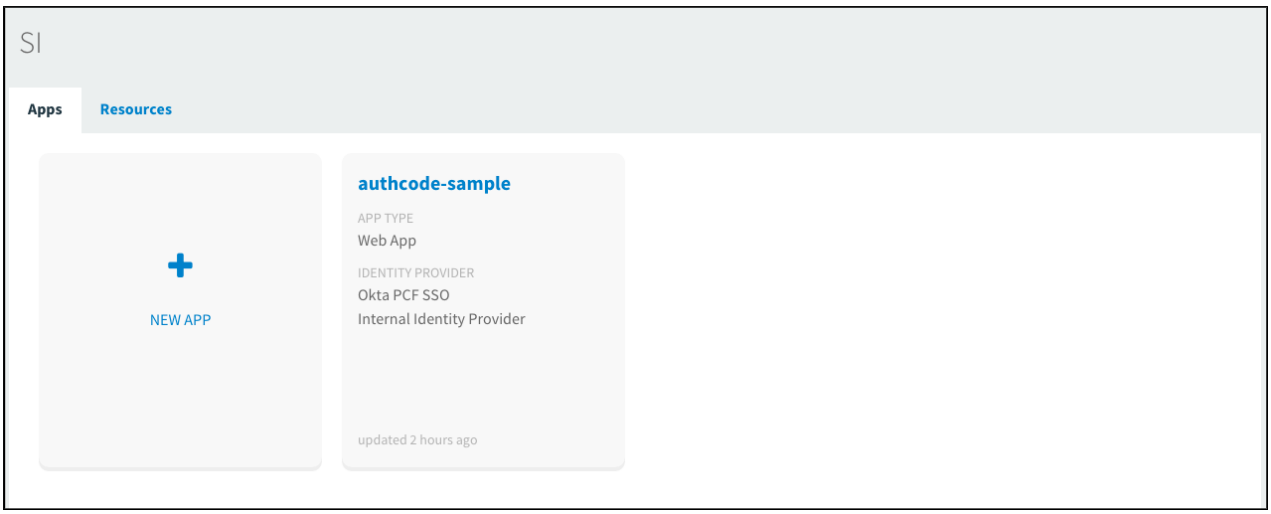
Test Your Service Provider Connection

1. Log in to Apps Manager at `https://apps.YOUR-SYSTEM-DOMAIN` and navigate to the organization and space where your application is located.
2. Under **Services**, locate the service instance of the Single Sign-On (SSO) plan bound to your application and click **Manage**.

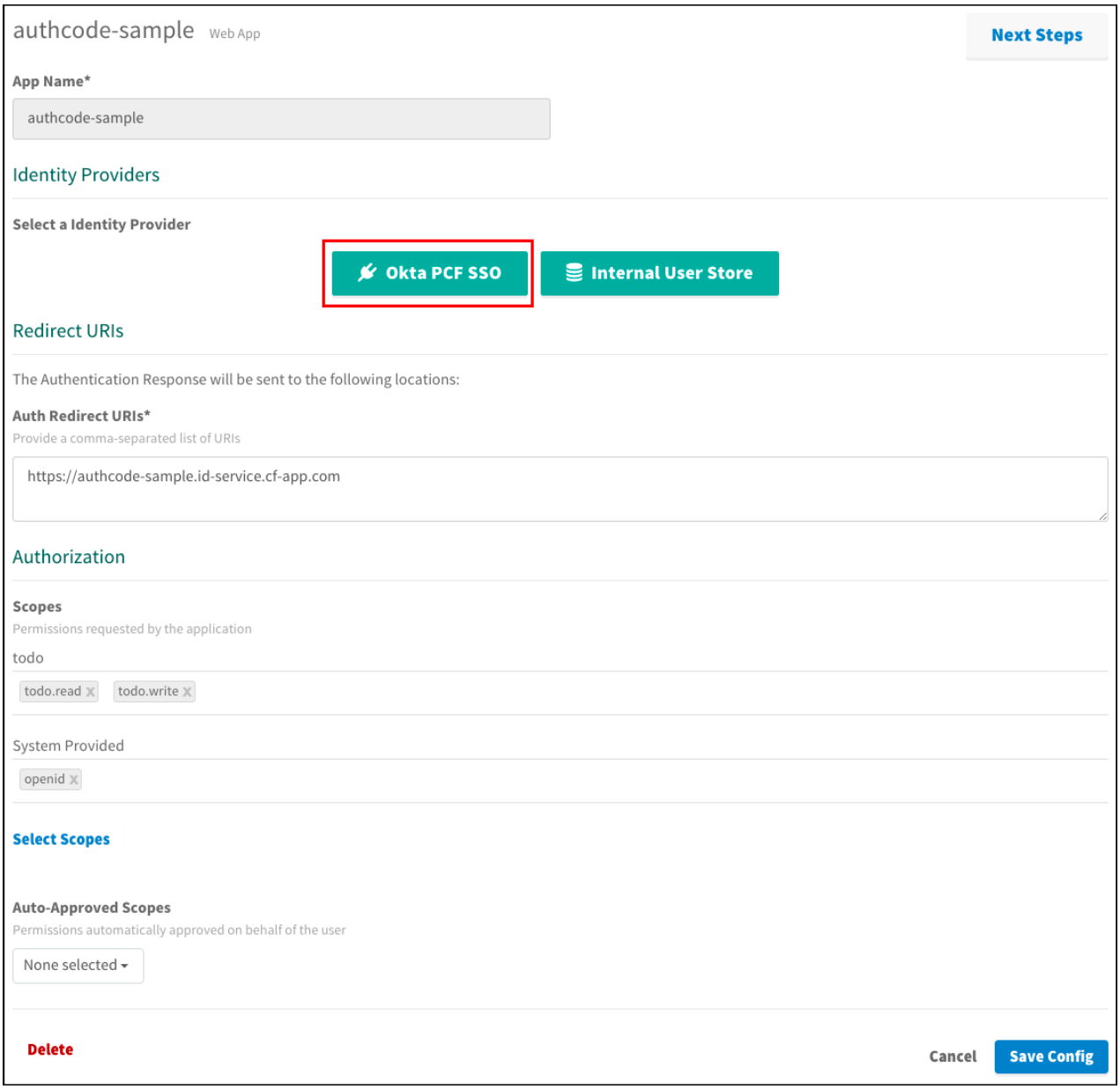




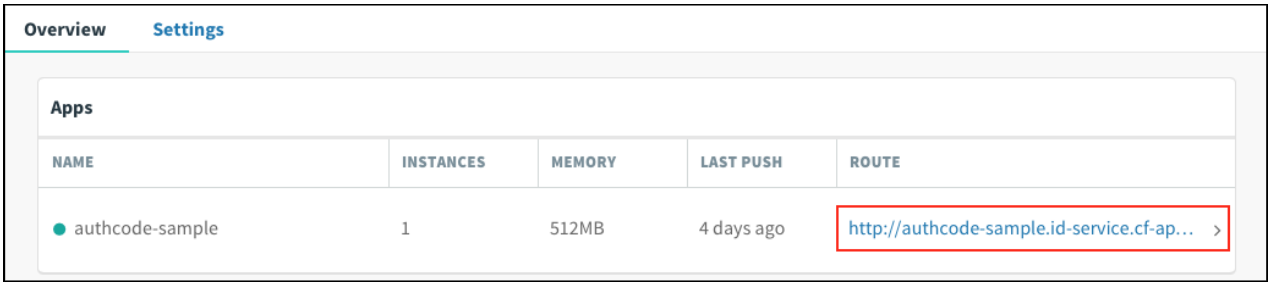
3. Under the **Apps** tab, click your application.



4. Under **Identity Providers**, select the Okta identity provider.

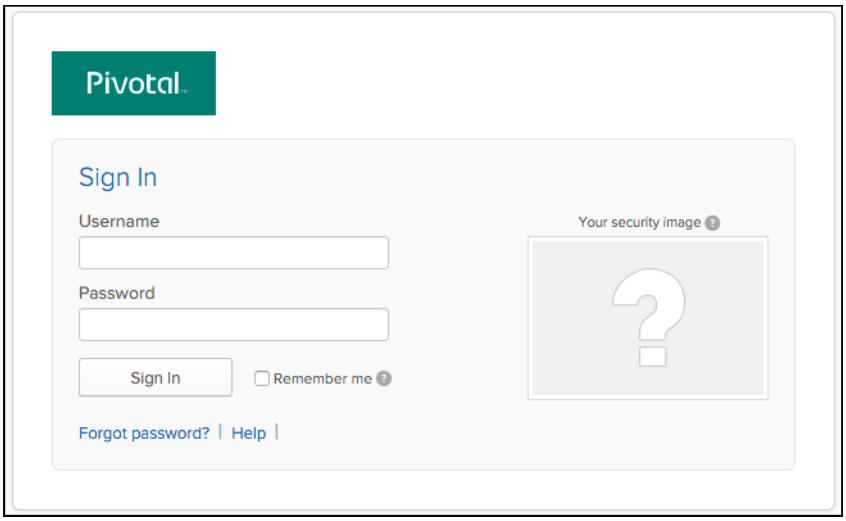


5. Return to Apps Manager and click on the URL below your application to be redirected to the identity provider to authenticate.

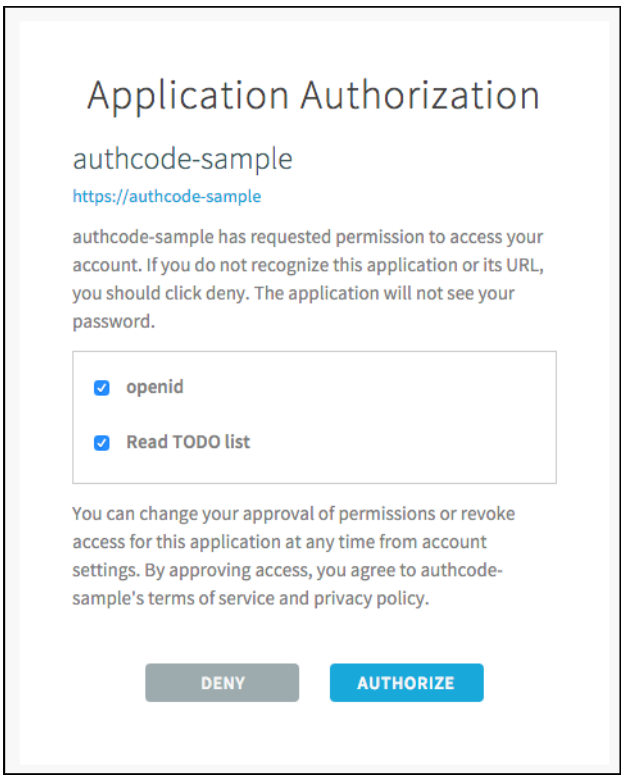


6. Click the link.

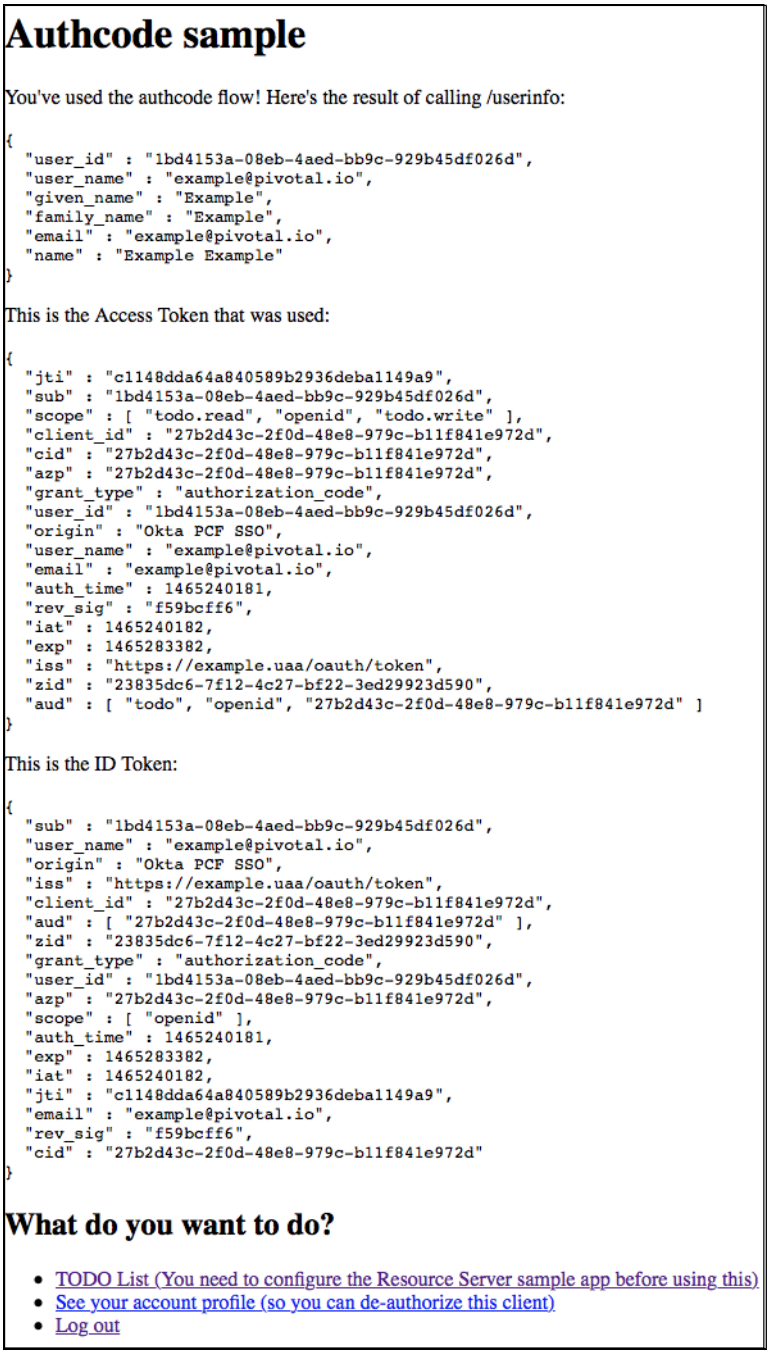
7. On the identity provider sign-in page, enter your credentials and click **Sign In**.



8. The application asks for authorization to the necessary scopes. Click **Authorize**.



9. The access token and ID token displays.



Authcode sample

You've used the authcode flow! Here's the result of calling /userinfo:


```
{
  "user_id" : "1bd4153a-08eb-4aed-bb9c-929b45df026d",
  "user_name" : "example@pivotal.io",
  "given_name" : "Example",
  "family_name" : "Example",
  "email" : "example@pivotal.io",
  "name" : "Example Example"
}
```


This is the Access Token that was used:


```
{
  "jti" : "c1148dda64a840589b2936deba1149a9",
  "sub" : "1bd4153a-08eb-4aed-bb9c-929b45df026d",
  "scope" : [ "todo.read", "openid", "todo.write" ],
  "client_id" : "27b2d43c-2f0d-48e8-979c-b11f841e972d",
  "cid" : "27b2d43c-2f0d-48e8-979c-b11f841e972d",
  "azp" : "27b2d43c-2f0d-48e8-979c-b11f841e972d",
  "grant_type" : "authorization_code",
  "user_id" : "1bd4153a-08eb-4aed-bb9c-929b45df026d",
  "origin" : "Okta PCF SSO",
  "user_name" : "example@pivotal.io",
  "email" : "example@pivotal.io",
  "auth_time" : 1465240181,
  "rev_sig" : "f59bcff6",
  "iat" : 1465240182,
  "exp" : 1465283382,
  "iss" : "https://example.uaa/oauth/token",
  "zid" : "23835dc6-7f12-4c27-bf22-3ed29923d590",
  "aud" : [ "todo", "openid", "27b2d43c-2f0d-48e8-979c-b11f841e972d" ]
}
```


This is the ID Token:


```
{
  "sub" : "1bd4153a-08eb-4aed-bb9c-929b45df026d",
  "user_name" : "example@pivotal.io",
  "origin" : "Okta PCF SSO",
  "iss" : "https://example.uaa/oauth/token",
  "client_id" : "27b2d43c-2f0d-48e8-979c-b11f841e972d",
  "aud" : [ "27b2d43c-2f0d-48e8-979c-b11f841e972d" ],
  "zid" : "23835dc6-7f12-4c27-bf22-3ed29923d590",
  "grant_type" : "authorization_code",
  "user_id" : "1bd4153a-08eb-4aed-bb9c-929b45df026d",
  "azp" : "27b2d43c-2f0d-48e8-979c-b11f841e972d",
  "scope" : [ "openid" ],
  "auth_time" : 1465240181,
  "exp" : 1465283382,
  "iat" : 1465240182,
  "jti" : "c1148dda64a840589b2936deba1149a9",
  "email" : "example@pivotal.io",
  "rev_sig" : "f59bcff6",
  "cid" : "27b2d43c-2f0d-48e8-979c-b11f841e972d"
}
```

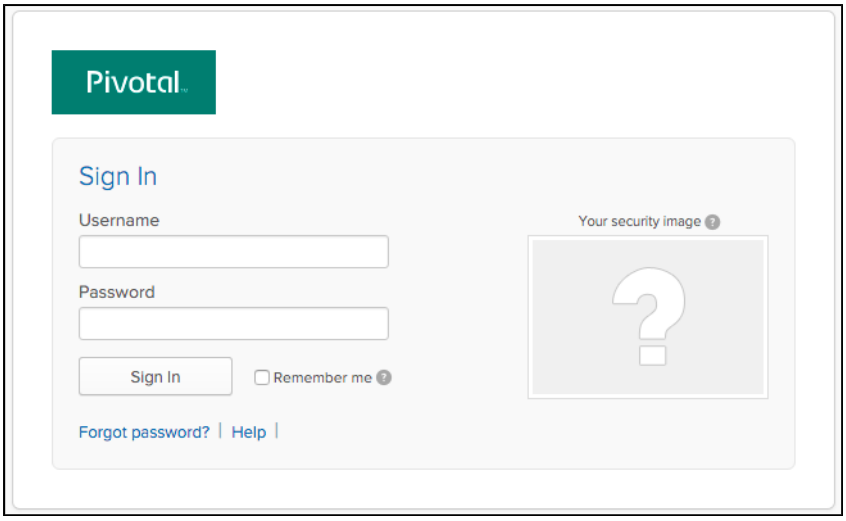

What do you want to do?

- TODO List (You need to configure the Resource Server sample app before using this)
- See your account profile (so you can de-authorize this client)
- Log out

Test Your Identity Provider Connection

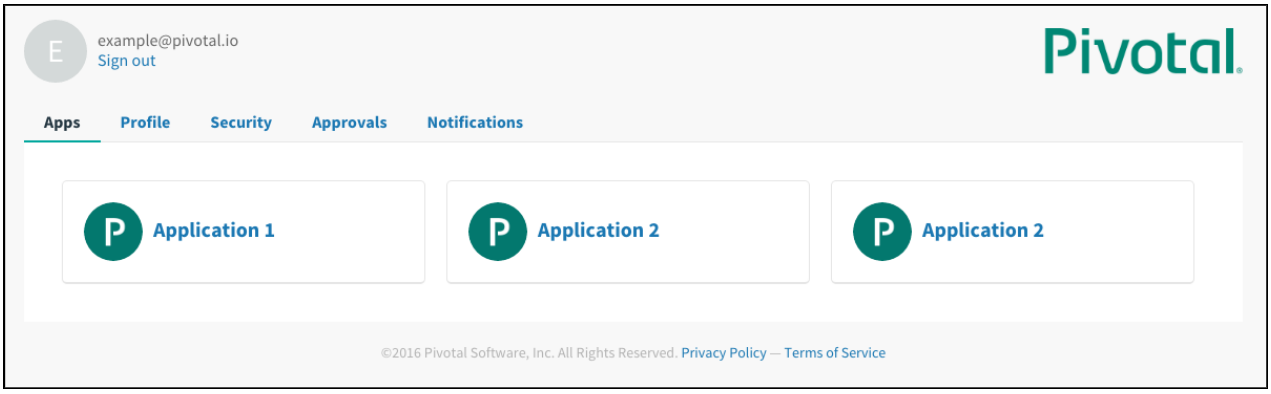
Note: SSO does not support identity provider-initiated flow into applications, but it does redirect the user to the User Account and Authentication (UAA) page to select applications assigned to the user.

1. Sign into Okta.



2. Navigate to the application tile and click it.

3. You are redirected to the page that lists applications you have access to.

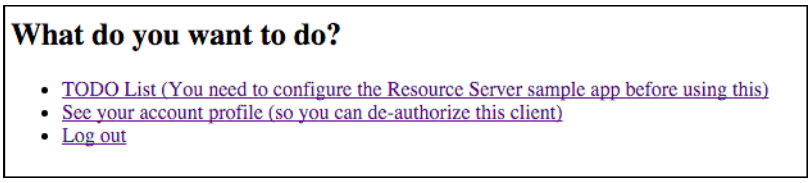


Test Your Single Sign-Off

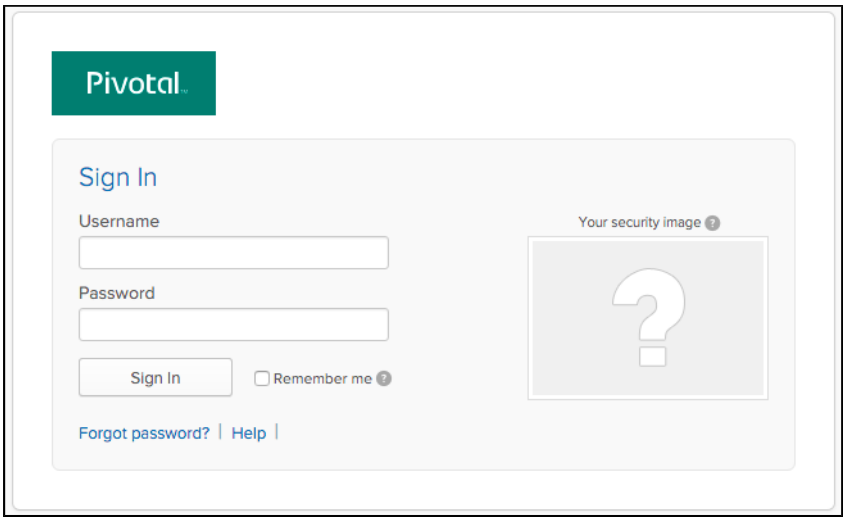
Test single sign-off to ensure that when users log out of the application, they are logged out of Okta as well.

1. Sign into the sample application. Information about the access and ID token displays, as well as the “What do you want to do?” section.

2. Under “What do you want to do?”, click **Log out**.



3. You are logged out and redirected to the Okta login page.

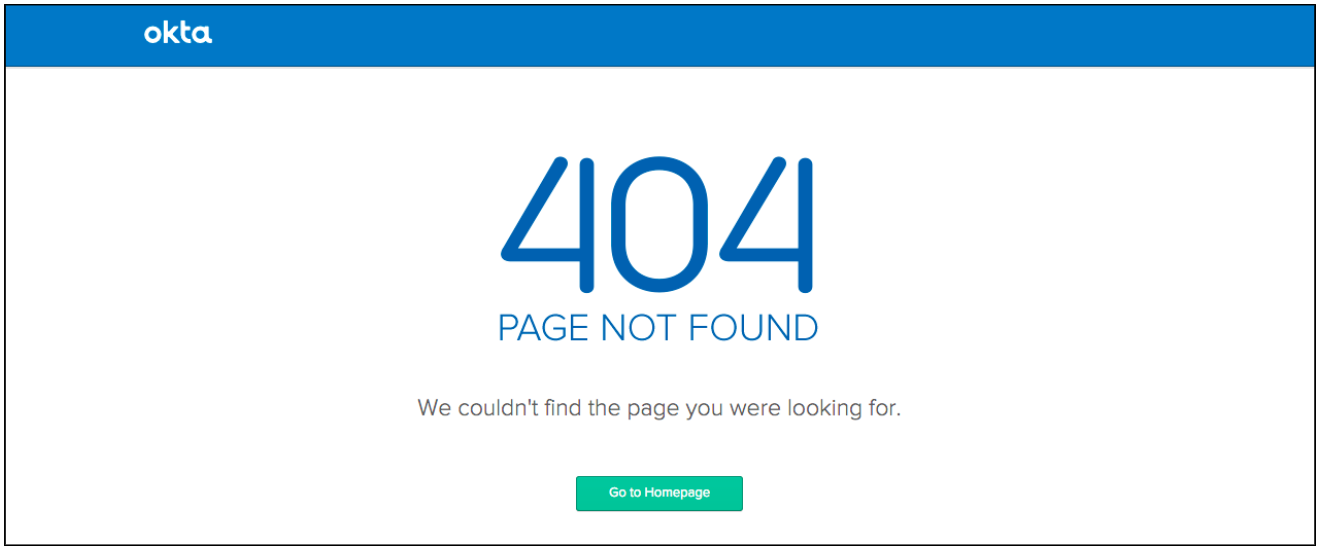


Troubleshooting

This topic describes how to resolve common errors that arise when configuring a single sign-on partnership between Okta and Pivotal Single Sign-On (SSO).

Page Not Found

Symptom:

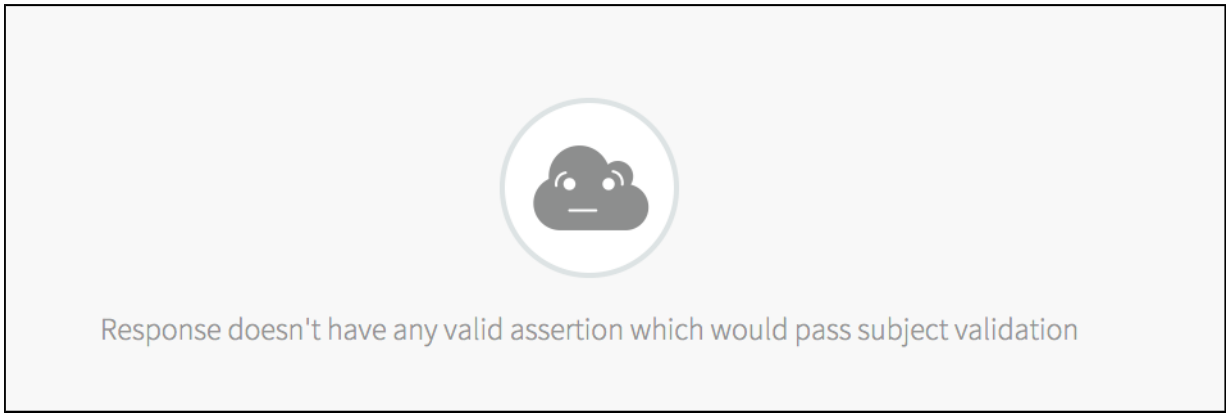


Explanations:

- The Okta instance is inactive.
- The Recipient URL is misconfigured in Okta.
- The identity provider SSO URL is misconfigured in the SSO plan settings.

No Valid Assertion

Symptom:

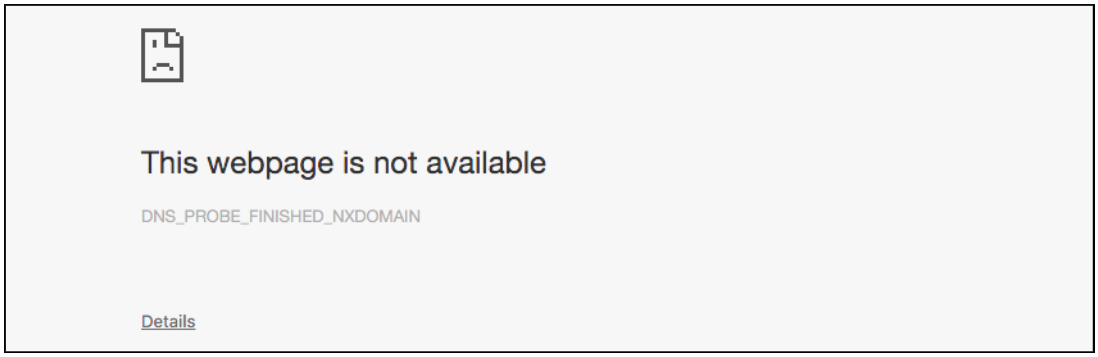


Explanations:

- The service provider Entity ID is misconfigured in Okta.
- The Destination URL is misconfigured in Okta.

Webpage Not Available

Symptom:

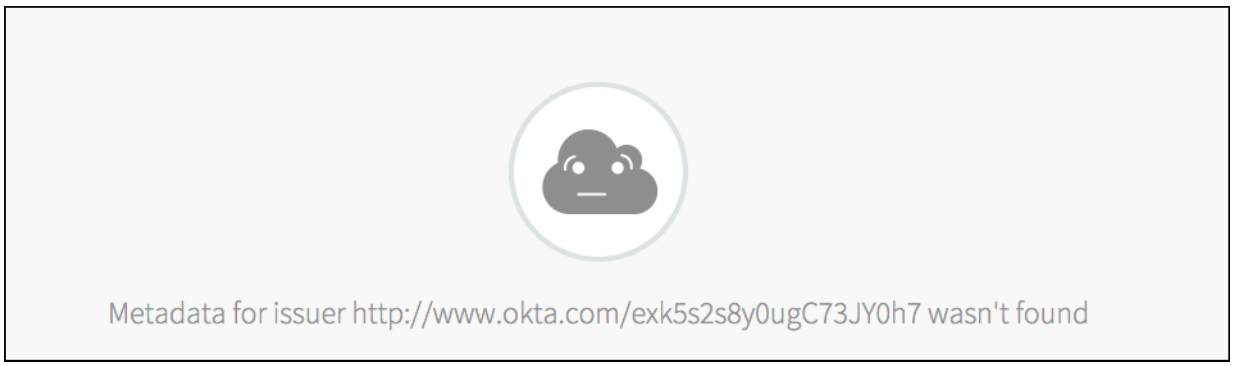


Explanation:

- The SSO URL is misconfigured in Okta.

Metadata Not Found

Symptom:



Explanation:

- The identity provider Entity ID is misconfigured in the SSO plan settings.

PingFederate Integration Guide Overview

PingFederate is a federation server that provides identity management, single sign-on, and API security for the enterprise. This documentation describes how to configure a single sign-on partnership between PingFederate as the Identity Provider (IdP) and the Single Sign-On Service (SSO) for Pivotal Cloud Foundry as the Service Provider (SP).

SSO supports service provider-initiated authentication flow and single logout. It does not support identity provider-initiated authentication flow. All SSO communication takes place over SSL.

Prerequisites

To integrate PingFederate with Pivotal Cloud Foundry (PCF), you need:

Pivotal

- PCF, version 1.7.0 or later.
- Single Sign-On, version 1.1.0 or later.

Ping

- PingFederate
- A user with Administrator privileges.

Note: To configure SAML, you must have the Single Sign-On service broker installed on your PCF deployment. You need to create a plan, grant any plan administrators, and specify any organizations this plan should be the authentication authority for. For help configuring plans, see the Manage Service Plans topic.

PingFederate Integration Guide

Configuring PingFederate with SSO

Complete both steps below to integrate your deployment with PingFederate and SSO.

1. Configure PingFederate as an Identity Provider
2. Configure a Single Sign-On Service Provider

Testing and Troubleshooting

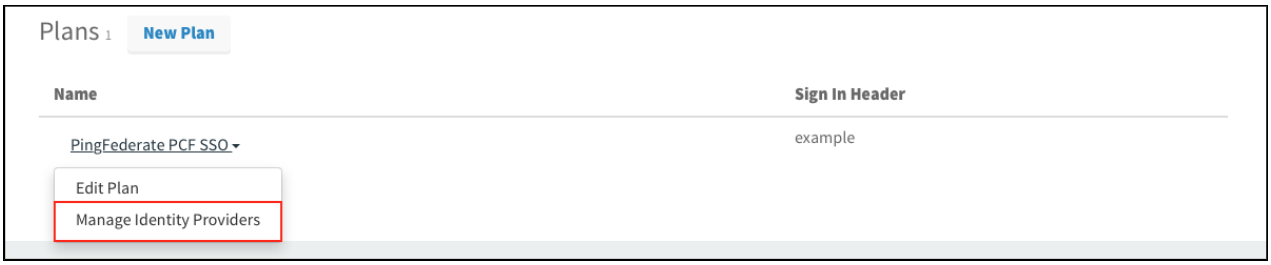
- Testing
- Troubleshooting

Configure PingFederate as an Identity Provider

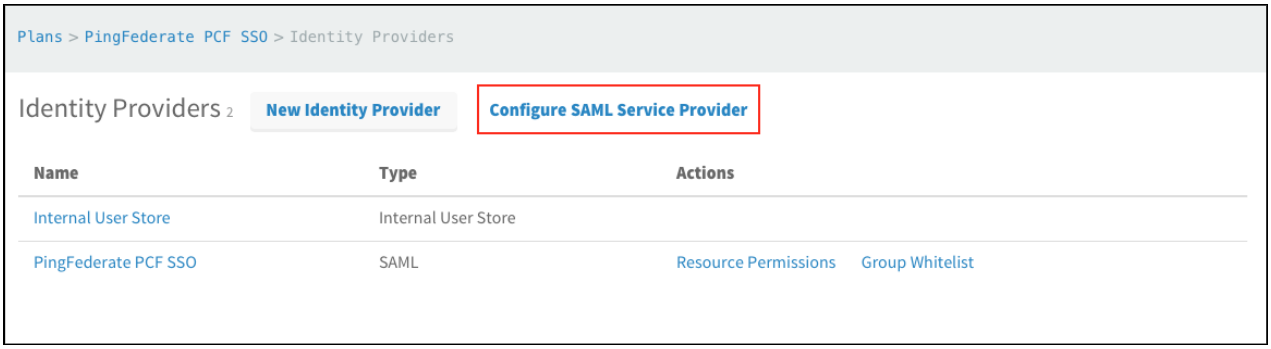
This topic describes how to set up PingFederate as your identity provider by configuring SAML integration in both Pivotal Cloud Foundry (PCF) and PingFederate.

Set up SAML in PCF

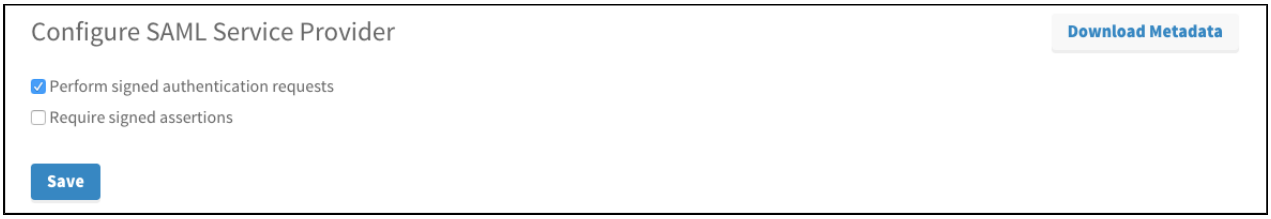
1. Log into the Single Sign-On (SSO) dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.
2. Select your plan and choose **Manage Identity Providers** from the dropdown menu.



3. Click **Configure SAML Service Provider**.



4. (Optional) Select **Perform signed authentication requests** to enforce SSO private key signature and identity provider validation.

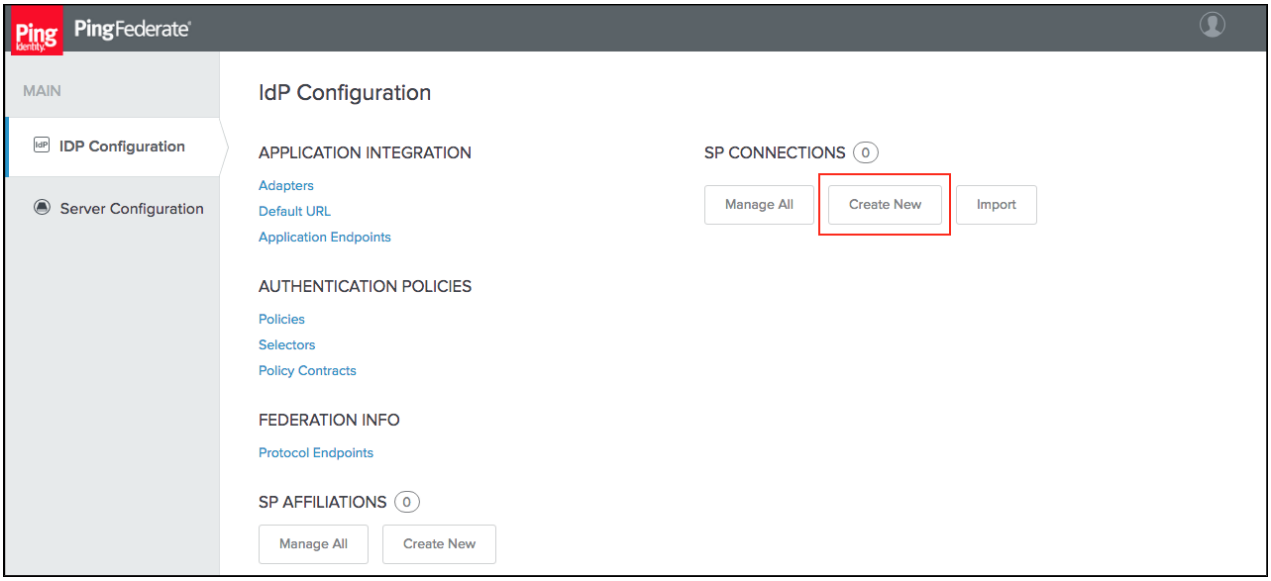


5. (Optional) Select **Require signed assertions** to validate the origin of signed responses.
6. Click **Download Metadata** to download the service provider metadata.
7. Click **Save**.

Set up SAML in PingFederate

Configure the Connection

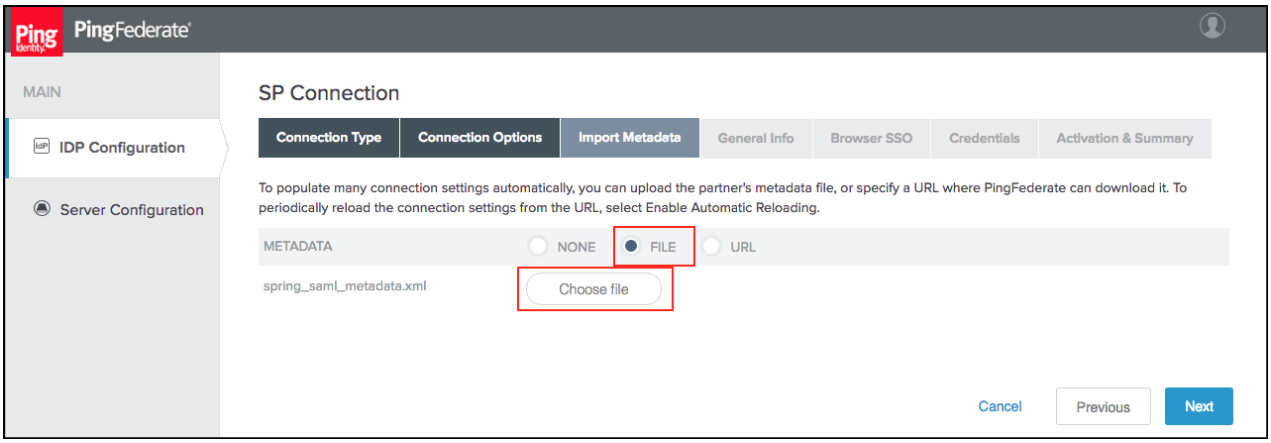
1. Sign in as a PingFederate administrator.
2. Navigate to your identity provider configurations by clicking on the **IDP Configuration** tab.
3. Under **SP Connections**, click the **Create New** button.



4. Select the **Browser SSO Profiles** connection template on the **Connection Type** tab and click **Next**.

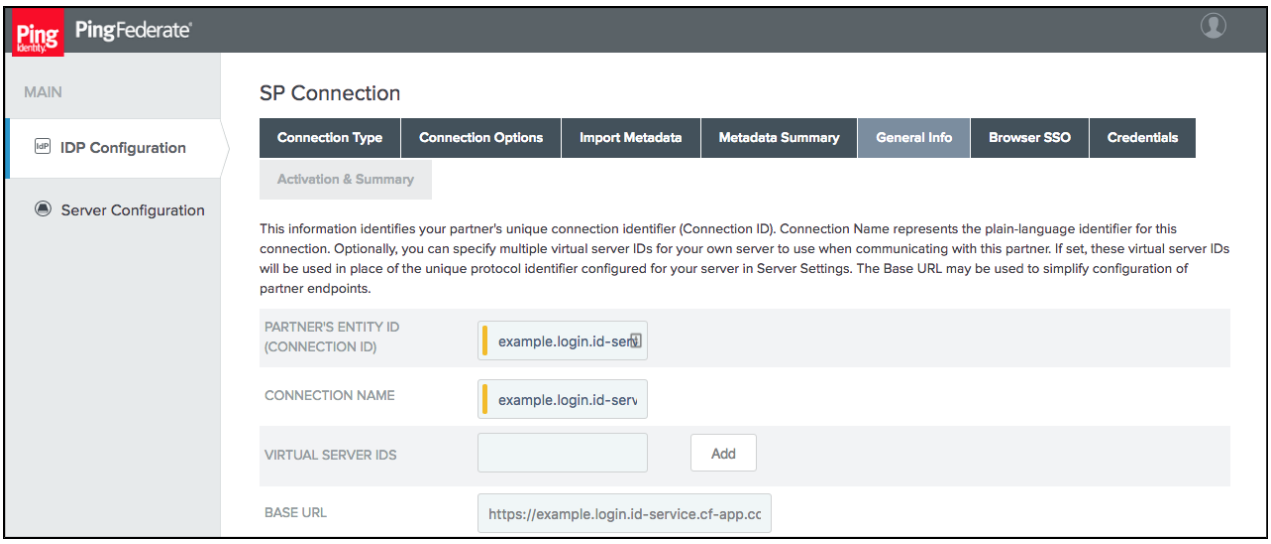
5. Select **Browser SSO** on the **Connection Options** tab and click **Next**.

6. Select **File** as the method for importing metadata and click **Choose file** to choose the SSO metadata on the **Import Metadata** tab. Click **Next**.



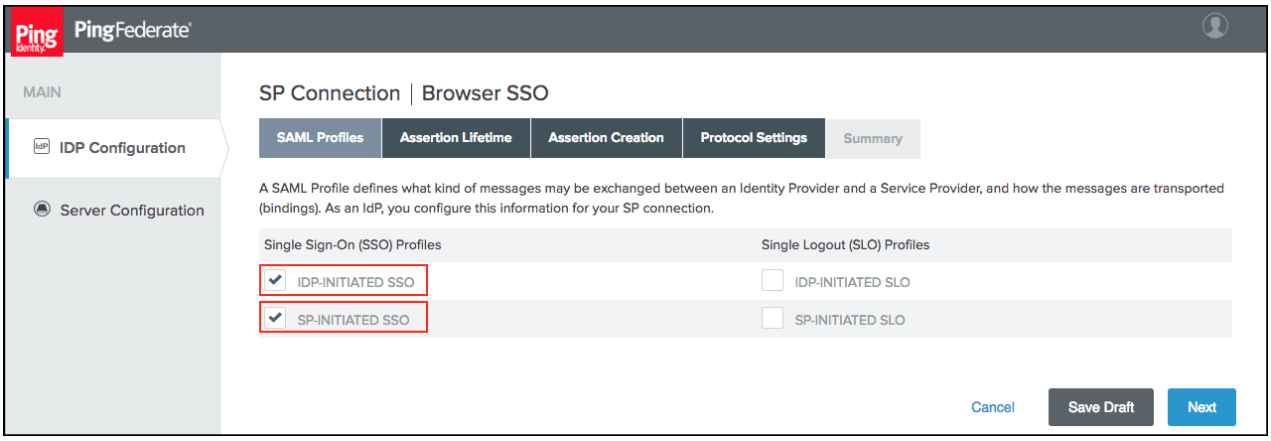
7. Review the information on the **Metadata Summary** tab and click **Next**.

8. Ensure that the **Partner's Entity ID**, **Connection Name**, and **Base URL** fields pre-populate based on the metadata. Click **Next**.



Configure Browser SSO

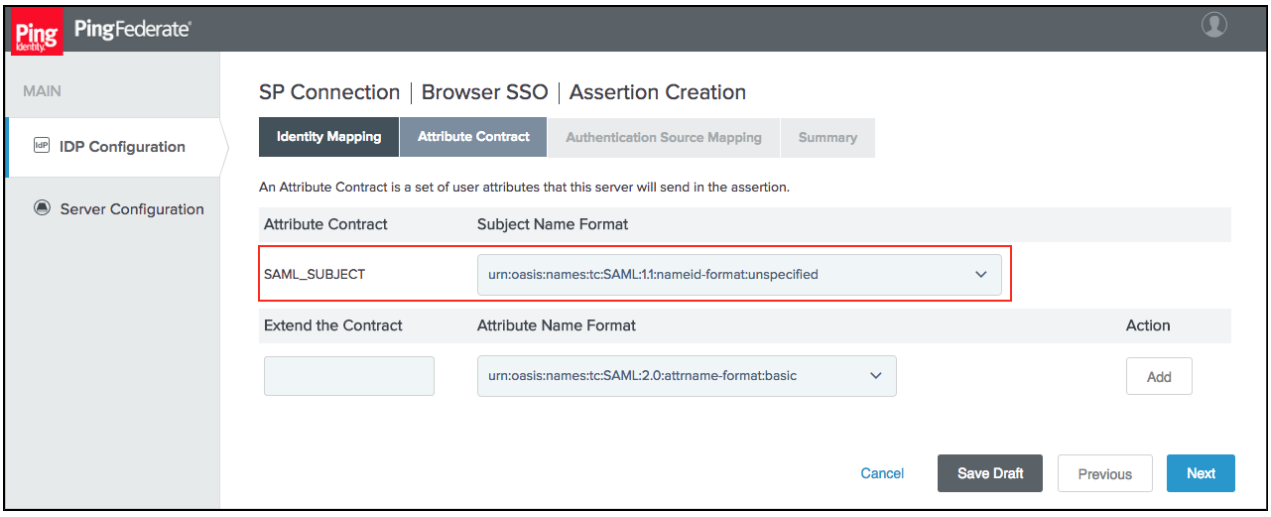
1. Click **Configure Browser SSO** on the **Browser SSO** tab.
2. Select the **IdP-Initiated SSO** and **SP-Initiated SSO** options on the **SAML Profiles** tab and click **Next**.



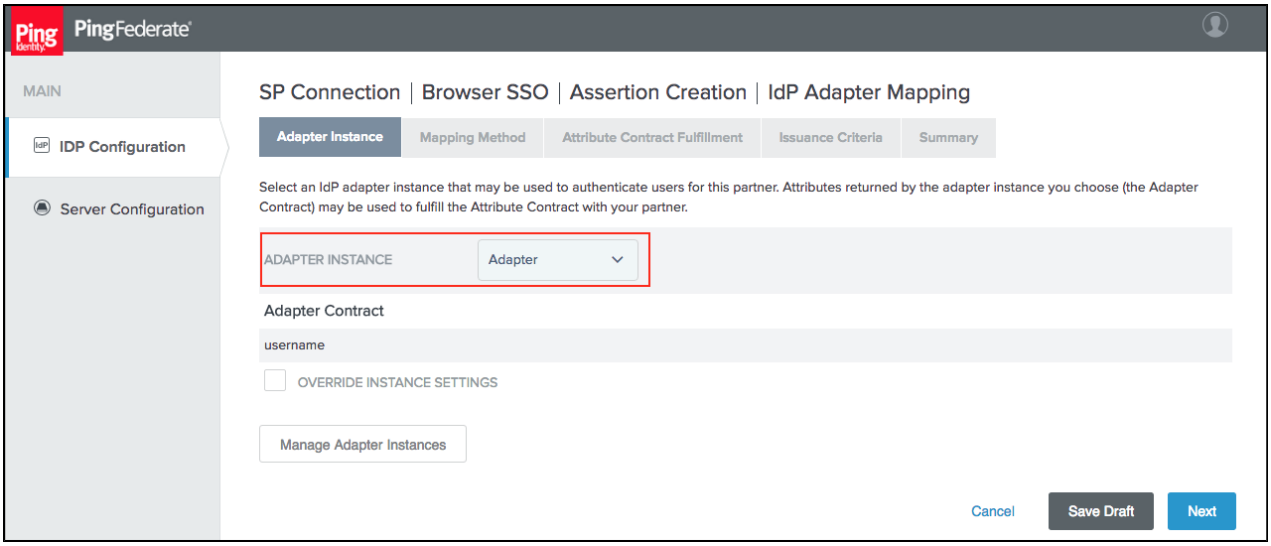
3. Enter your desired assertion validity time from on the **Assertion Lifetime** tab and click **Next**.

Assertion Creation

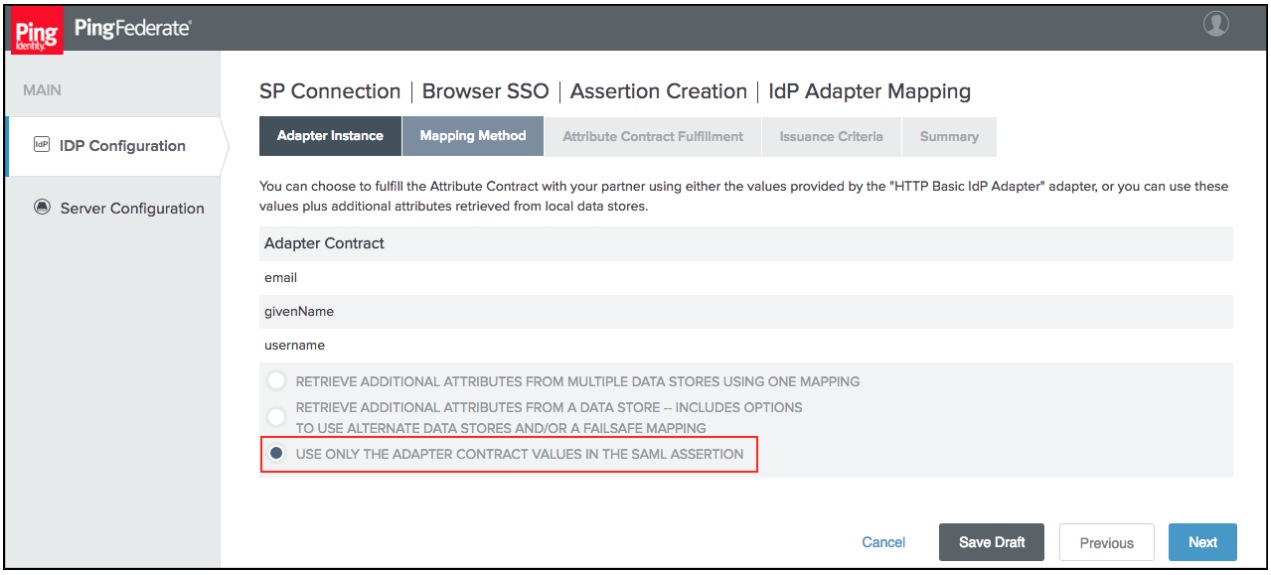
1. Click **Configure Assertion Creation** on the **Assertion Creation** tab.
2. Choose the **Standard** option on the **Identity Mapping** tab and click **Next**.
3. Select a **Subject Name Format** for the **SAML_SUBJECT** on the **Attribute Contract** tab and click **Next**.



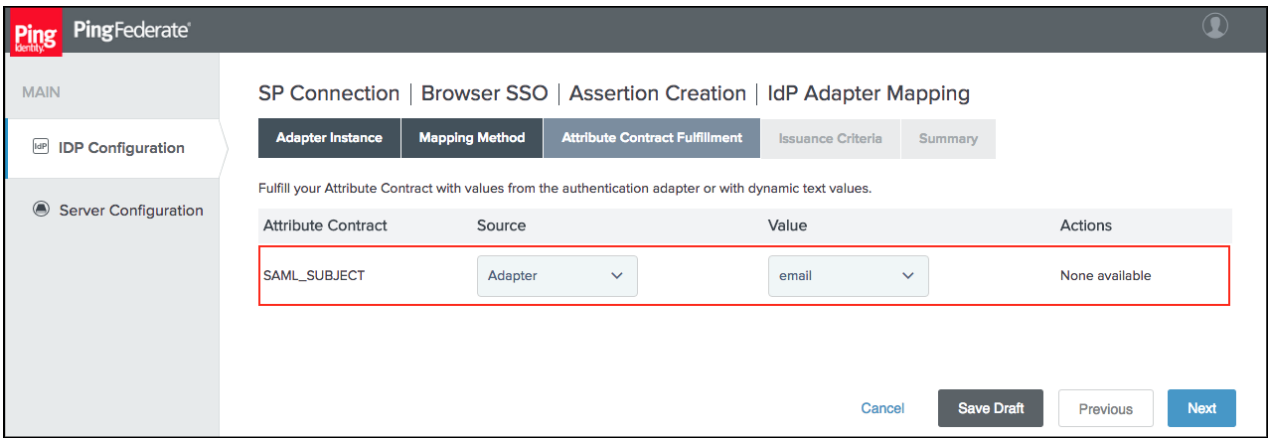
4. Click **Map New Adapter Instance** on the **Authentication Source Mapping** tab.
5. Select an **Adapter Instance** and click **Next**. The adapter must include the user's email address.



6. Select the **Use only the adapter contract values in the SAML assertion** option on the **Mapping Method** tab and click **Next**.



7. Select your adapter instance as the **Source** and the email as the **Value** on the **Attribute Contract Fullfillment** tab and click **Next**.

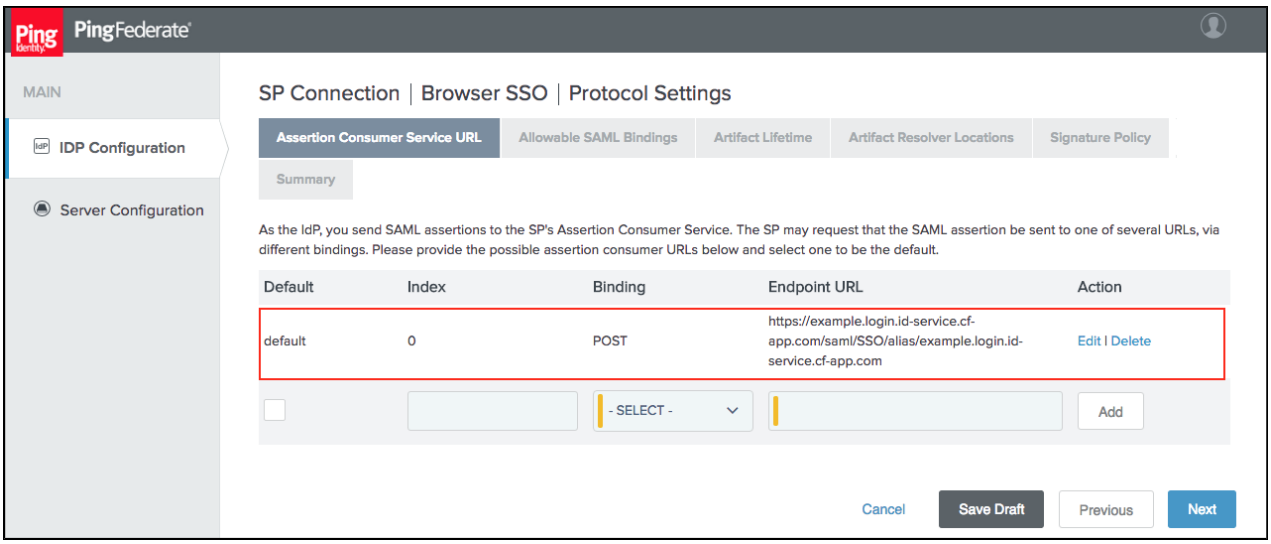


8. (Optional) Select any authorization conditions you would like on the **Issuance Criteria** tab and click **Next**.
9. Click **Done** on the **Summary** tab.
10. Click **Next** on the **Authentication Source Mapping** tab.
11. Click **Done** on the **Summary** tab.

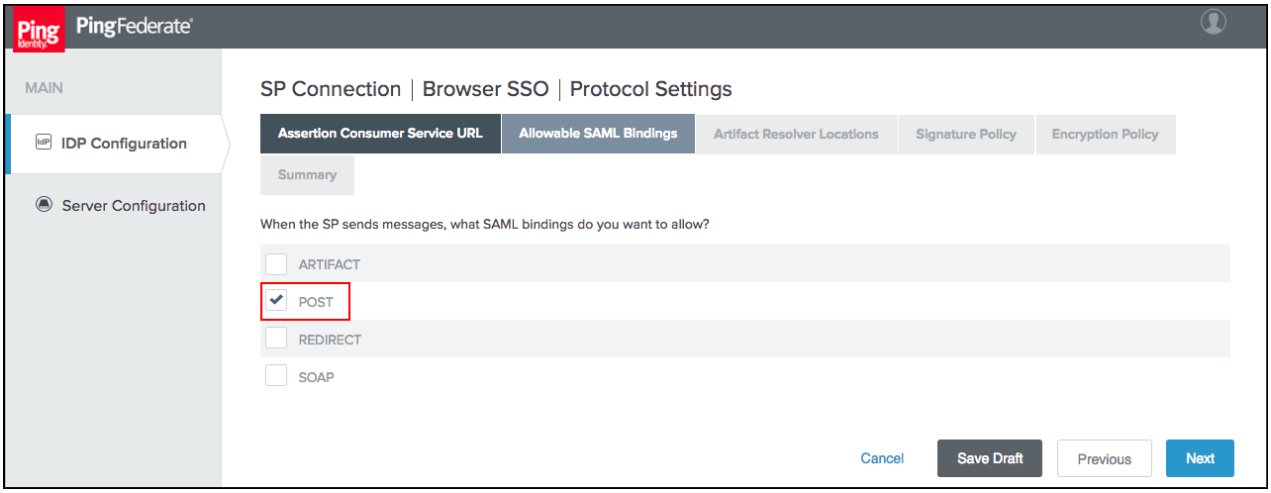
12. Click **Next** on the **Assertion Creation** tab.

Protocol Settings

1. Click **Configure Protocol Settings** on the **Protocol Settings** tab.
2. Select POST for **Binding** and specify the single sign-on endpoint url in the **Endpoint URL** field on the **Assertion Consumer Service URL** tab. Click **Next**



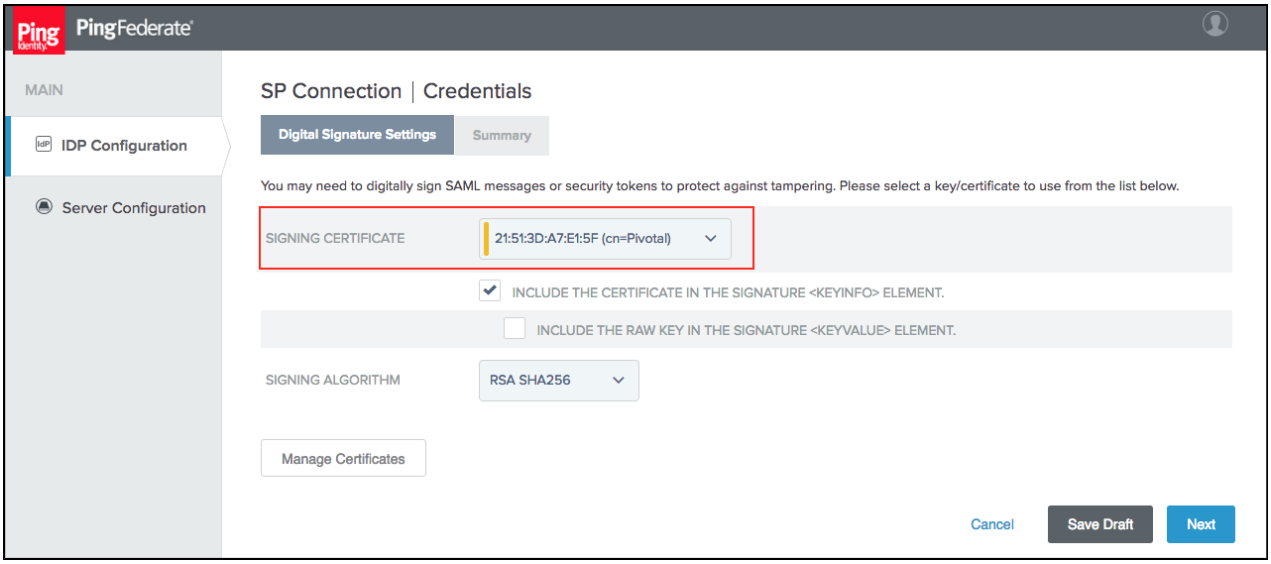
3. Select **POST** on the **Allowable SAML Bindings** tab and click **Next**.



4. Select your desired signature policies for assertions on the **Signature Policy** tab and click **Next**.
5. Select your desired encryption policy for assertions on the **Encryption Policy** tab and click **Next**.
6. Click **Done** on the **Protocol Settings Summary** tab.
7. Click **Done** on the **Browser SSO Summary** tab.

Configure Credentials

1. Click **Configure Credentials** on the **Credentials** tab.
2. Select the **Signing Certificate** to use with the Single Sign-On service and select **Include the certificate in the signature element**. Click **Next**.



3. Click **Done** on the **Summary** tab.
4. Click **Next** on the **Credentials** tab.
5. Select **Active** for the **Connection Status** on the **Activation & Summary** tab and click **Save**.
6. Click **Manage All** under **SP Connections**.
7. Click **Export Metadata** for the desired service provider connection.
8. Choose a **Signing Certificate** on the **Metadata Signing** tab and click **Next**.
9. Click **Export** on the **Export & Summary** tab and click **Done**.

Configure a Single Sign-On Service Provider

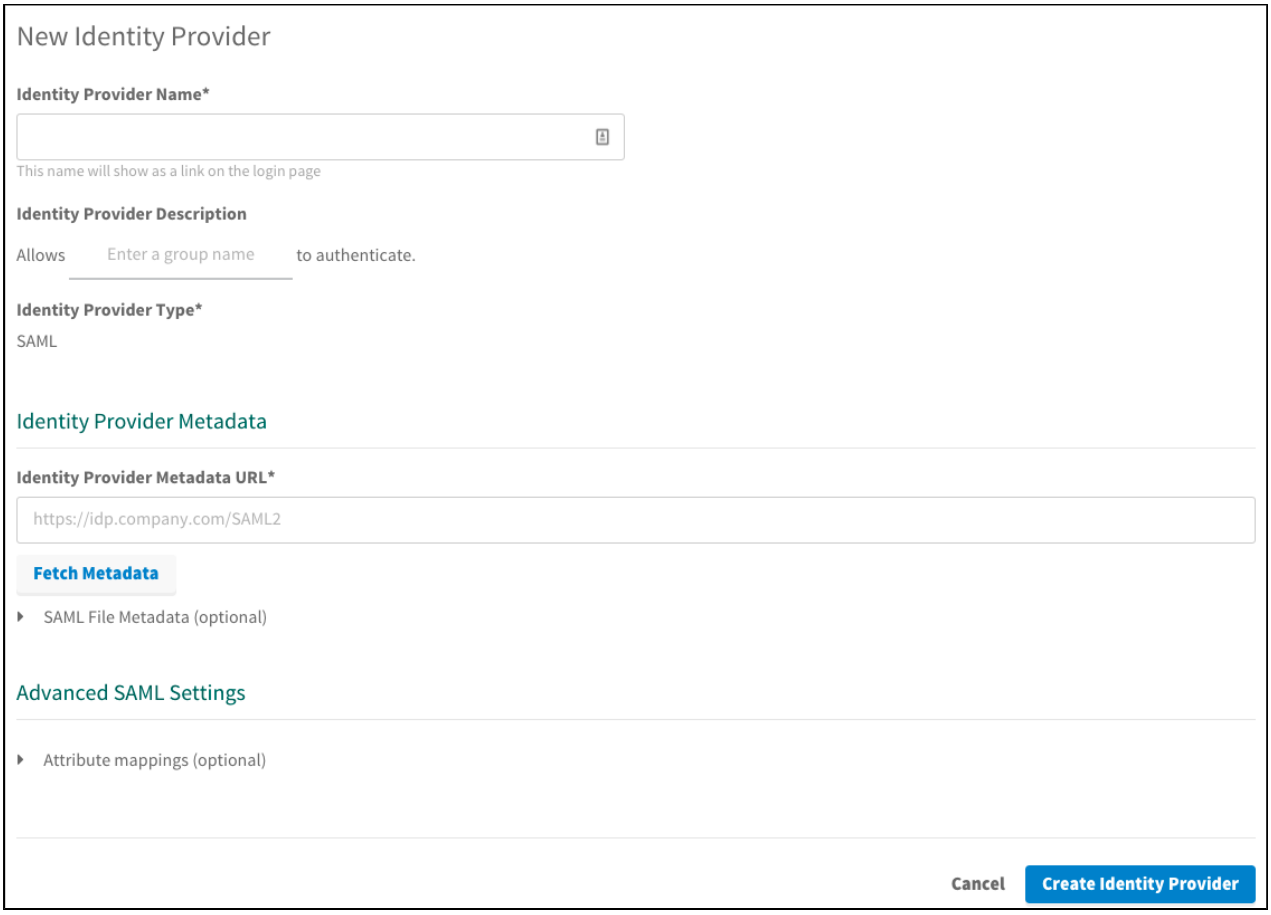
This topic describes how to add an external identity provider to your Pivotal Single Sign-On (SSO) service plan.

Setting up SAML

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.

2. Select your plan and choose **Manage Identity Providers** from the dropdown menu.

3. Click **New Identity Provider**.



4. To create a new identity provider, perform the following steps:

    a. Enter an identity provider name into **Identity Provider Name**.
    b. (Optional) Enter a description into **Identity Provider Description**.
    c. Click **SAML File Metadata (optional)**, then click the **Upload Identity Provider Metadata** button to upload your metadata XML.
    d. (Optional) Under **Advanced SAML Settings**, click **Attribute Mappings** to enter the mappings.

5. Click **Create Identity Provider**.

6. Click **Resource Permissions**.

7. Click **New Permissions Mapping** and perform the following steps:

    a. Enter a **Group Name**.
    b. For **Select Permissions**, select the permissions that the members of the group from the external identity provider should have access to.

8. Navigate to the identity provider list.

9. Click **Group Whitelist** and enter the group names from the external identity provider to propagate in the ID token when a user authenticates.

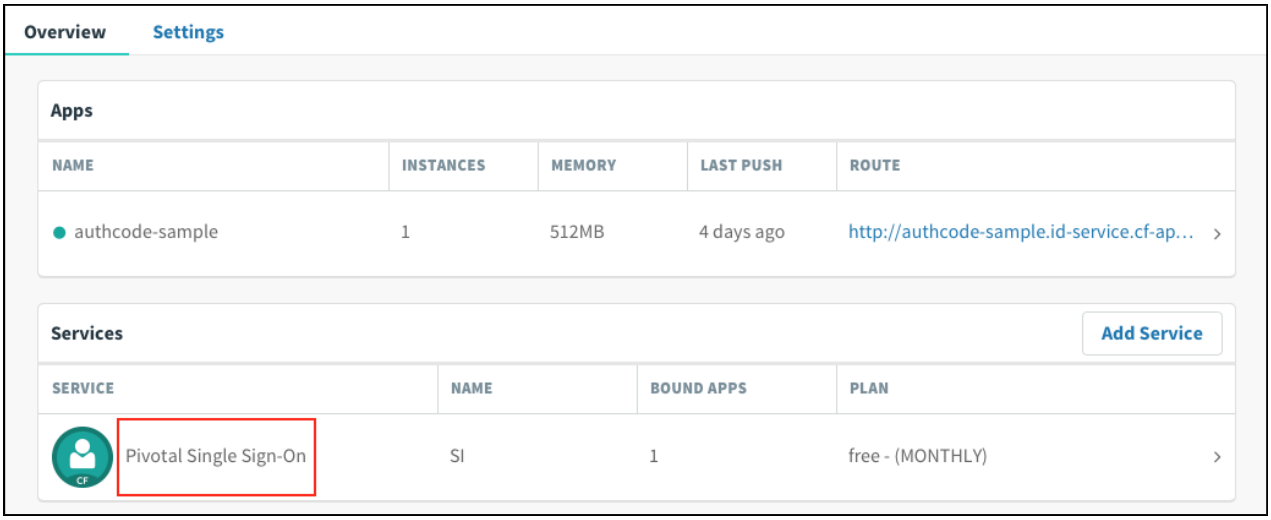
Testing

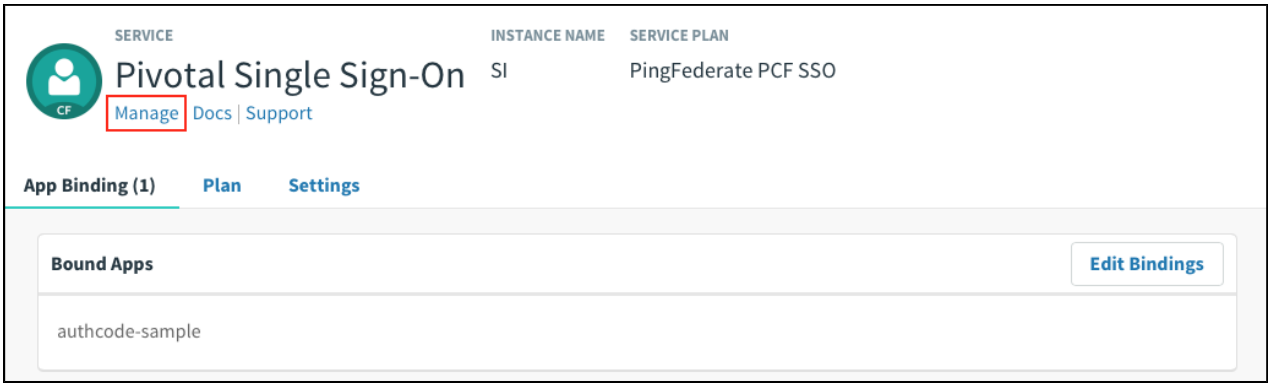
This topic describes how an administrator can test the connection between SSO and PingFederate. An administrator can test both service provider and identity provider connections.

Test Your Service Provider Connection

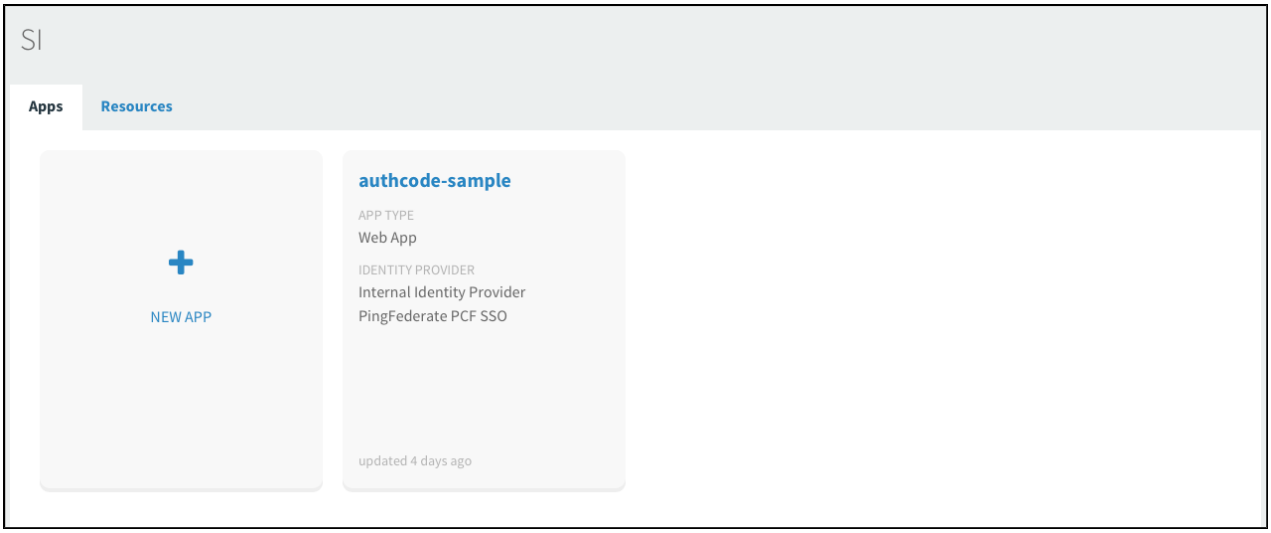
1. Log in to Apps Manager at `https://apps.YOUR-SYSTEM-DOMAIN` and navigate to the organization and space where your application is located.

2. Under **Services**, locate the service instance of the Single Sign-On (SSO) plan bound to your application. Click the service instance and then click **Manage**.

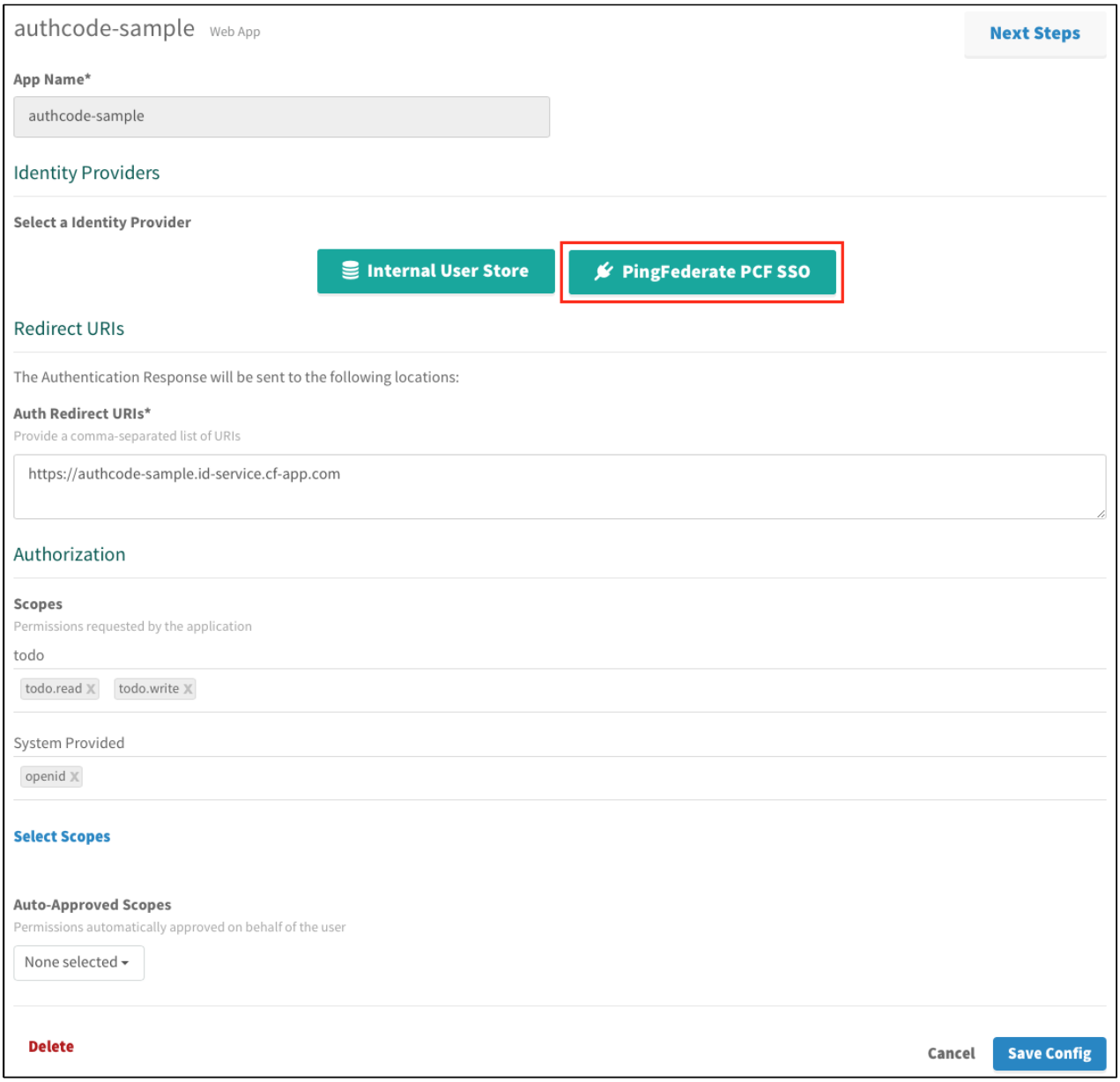




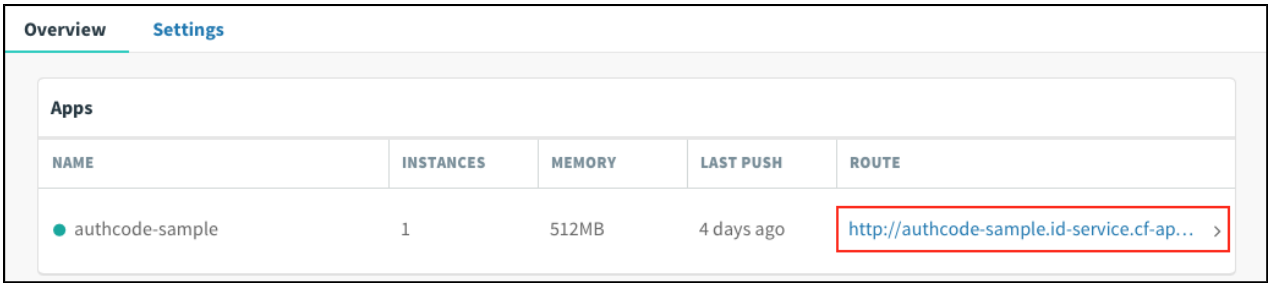
3. Under the **Apps** tab, click your application.



4. Under **Identity Providers**, select the PingFederate identity provider. a

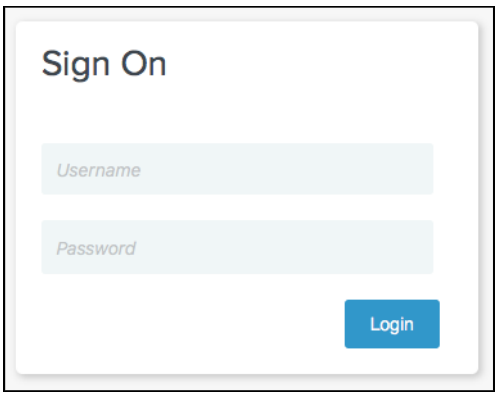


5. Return to Apps Manager and click the URL below your application to authenticate with the identity provider.

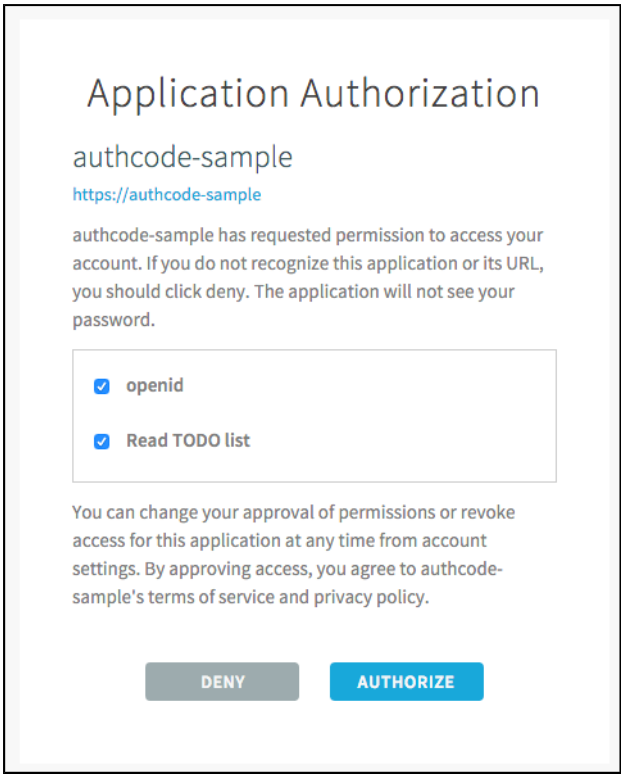


6. Click the link to **Log in via Auth Code Grant Type**.

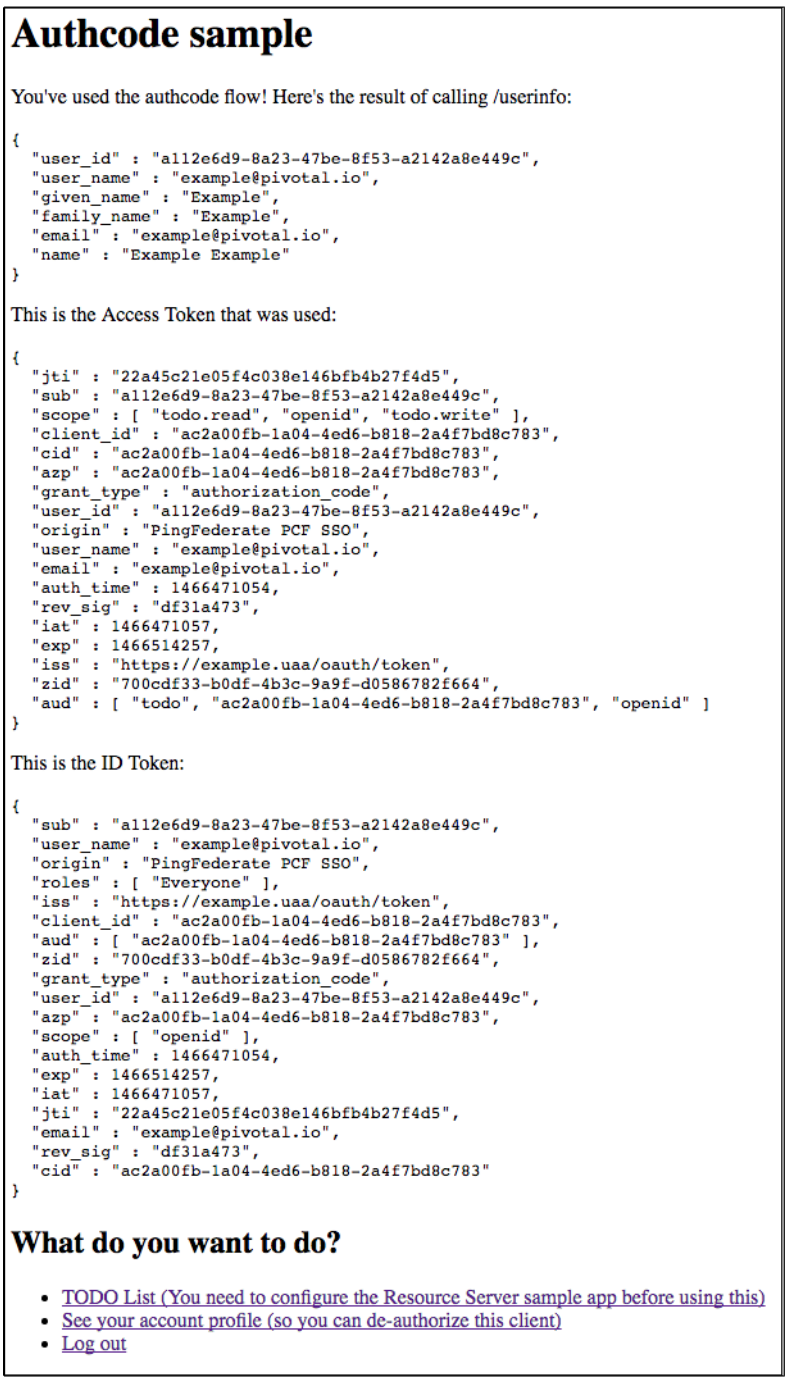
7. On the identity provider sign-in page, enter your credentials and click **Sign On**.



8. The application asks for authorization to the necessary scopes. Click **Authorize**.



9. View the access token and ID token.



Authcode sample

You've used the authcode flow! Here's the result of calling /userinfo:


```
{
  "user_id" : "a112e6d9-8a23-47be-8f53-a2142a8e449c",
  "user_name" : "example@pivotal.io",
  "given_name" : "Example",
  "family_name" : "Example",
  "email" : "example@pivotal.io",
  "name" : "Example Example"
}
```


This is the Access Token that was used:


```
{
  "jti" : "22a45c21e05f4c038e146bfb4b27f4d5",
  "sub" : "a112e6d9-8a23-47be-8f53-a2142a8e449c",
  "scope" : [ "todo.read", "openid", "todo.write" ],
  "client_id" : "ac2a00fb-1a04-4ed6-b818-2a4f7bd8c783",
  "cid" : "ac2a00fb-1a04-4ed6-b818-2a4f7bd8c783",
  "azp" : "ac2a00fb-1a04-4ed6-b818-2a4f7bd8c783",
  "grant_type" : "authorization_code",
  "user_id" : "a112e6d9-8a23-47be-8f53-a2142a8e449c",
  "origin" : "PingFederate PCF SSO",
  "user_name" : "example@pivotal.io",
  "email" : "example@pivotal.io",
  "auth_time" : 1466471054,
  "rev_sig" : "df31a473",
  "iat" : 1466471057,
  "exp" : 1466514257,
  "iss" : "https://example.uaa/oauth/token",
  "zid" : "700cdf33-b0df-4b3c-9a9f-d0586782f664",
  "aud" : [ "todo", "ac2a00fb-1a04-4ed6-b818-2a4f7bd8c783", "openid" ]
}
```


This is the ID Token:


```
{
  "sub" : "a112e6d9-8a23-47be-8f53-a2142a8e449c",
  "user_name" : "example@pivotal.io",
  "origin" : "PingFederate PCF SSO",
  "roles" : [ "Everyone" ],
  "iss" : "https://example.uaa/oauth/token",
  "client_id" : "ac2a00fb-1a04-4ed6-b818-2a4f7bd8c783",
  "aud" : [ "ac2a00fb-1a04-4ed6-b818-2a4f7bd8c783" ],
  "zid" : "700cdf33-b0df-4b3c-9a9f-d0586782f664",
  "grant_type" : "authorization_code",
  "user_id" : "a112e6d9-8a23-47be-8f53-a2142a8e449c",
  "azp" : "ac2a00fb-1a04-4ed6-b818-2a4f7bd8c783",
  "scope" : [ "openid" ],
  "auth_time" : 1466471054,
  "exp" : 1466514257,
  "iat" : 1466471057,
  "jti" : "22a45c21e05f4c038e146bfb4b27f4d5",
  "email" : "example@pivotal.io",
  "rev_sig" : "df31a473",
  "cid" : "ac2a00fb-1a04-4ed6-b818-2a4f7bd8c783"
}
```

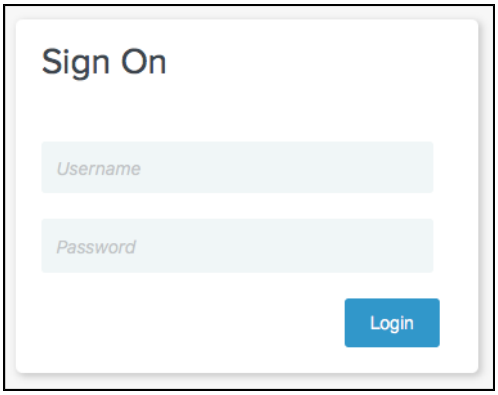

What do you want to do?

- TODO List (You need to configure the Resource Server sample app before using this)
- See your account profile (so you can de-authorize this client)
- Log out

Test Your Identity Provider Connection

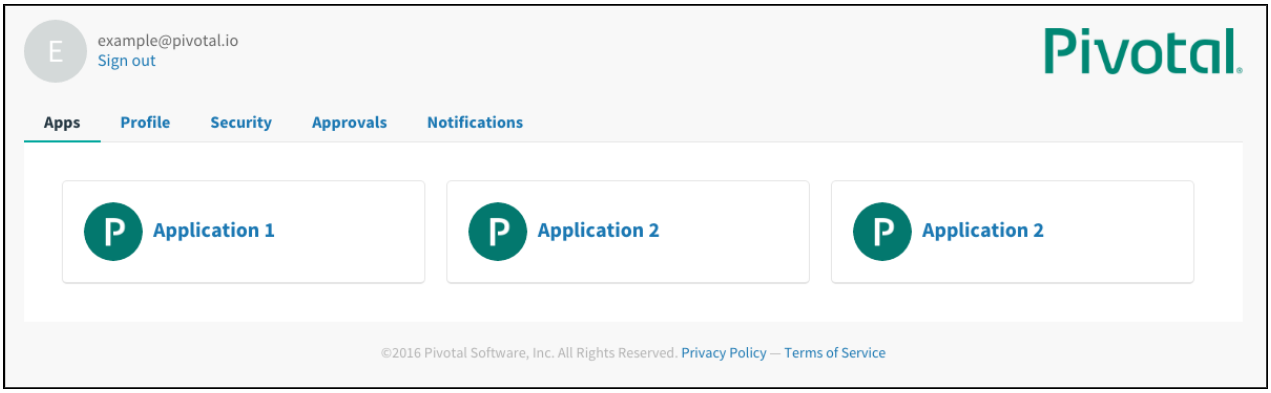
Note: SSO does not support identity provider-initiated flow into applications, but it does redirect the user to the User Account and Authentication (UAA) page to select applications assigned to the user.

1. Sign in to PingFederate.



2. Navigate to your application and click it.

3. View the list of applications you have access to.

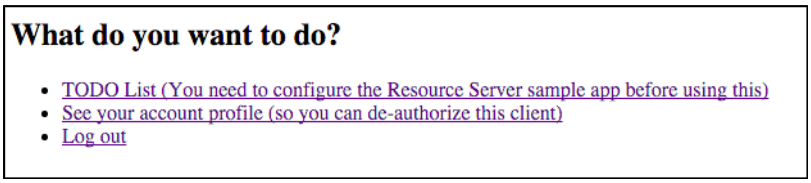


Test Your Single Sign-Off

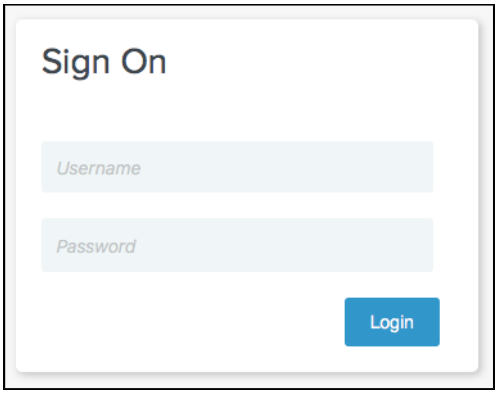
Test single sign-off to ensure that when users log out of the application, they are logged out of PingFederate as well.

1. Sign into the sample application. Information about the access and ID token displays, as well as the “What do you want to do?” section.

2. Under **What do you want to do?**, click **Log out**.



3. Ensure that you are logged out and redirected to the PingFederate login page.

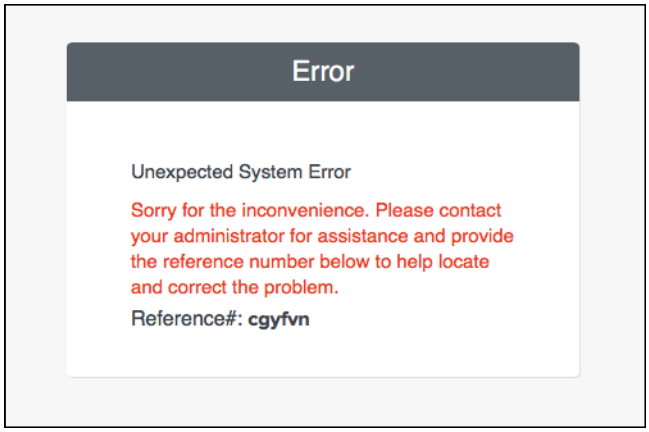


Troubleshooting

This topic describes how to resolve common errors that arise when configuring a single sign-on partnership between PingFederate and Pivotal Single Sign-On (SSO).

Error

Symptom:

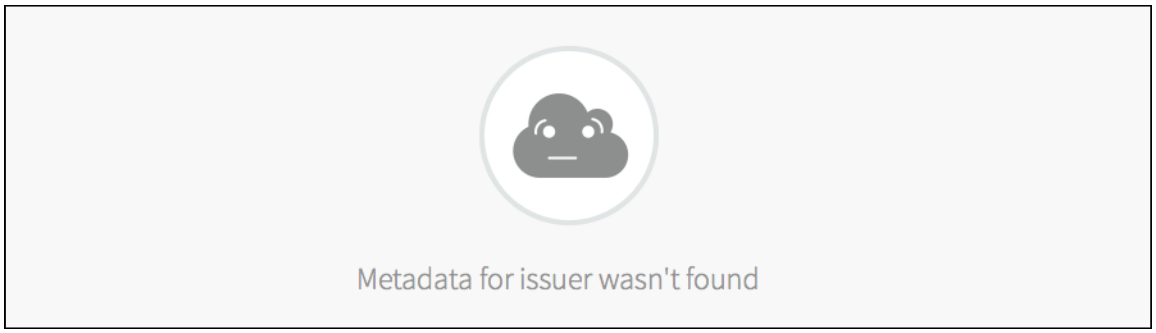


Explanations:

- Connection Status is disabled on PingFederate.
- The service provider Entity ID is misconfigured on PingFederate.
- The identity provider Single Sign-On URL is misconfigured in the SSO plan settings.

Metadata Not Found

Symptom:



Explanation:

- The identity provider Entity ID is misconfigured in the SSO plan settings.

PingOne Cloud Integration Guide Overview

PingOne Cloud is an identity-as-a-service solution that delivers secure single sign-on to SaaS. legacy and web applications. This documentation describes how to configure a single sign-on partnership between PingOne Cloud as the Identity Provider (IdP) and the Single Sign-On Service (SSO) for Pivotal Cloud Foundry as the Service Provider (SP).

SSO supports service provider-initiated authentication flow and single logout. It does not support identity provider-initiated authentication flow. All SSO communication takes place over SSL.

Prerequisites

To integrate PingOne Cloud with Pivotal Cloud Foundry (PCF), you need:

Pivotal

- PCF, version 1.7.0 or later.
- Single Sign-On, version 1.1.0 or later.

PingOne Cloud

- PingOne Cloud
- A user with Application Admin privileges.

Note: To configure SAML, you must have the Single Sign-On service broker installed on your PCF deployment. You need to create a plan, grant any plan administrators, and specify any organizations this plan should be the authentication authority for. For help configuring plans, see the Manage Service Plans topic..

PingOne Cloud Integration Guide

Configuring PingOne Cloud with SSO

Complete both steps below to integrate your deployment with PingOne Cloud and SSO.

1. Configure PingOne Cloud as an Identity Provider
2. Configure a Single Sign-On Service Provider

Testing and Troubleshooting

- Testing
- Troubleshooting

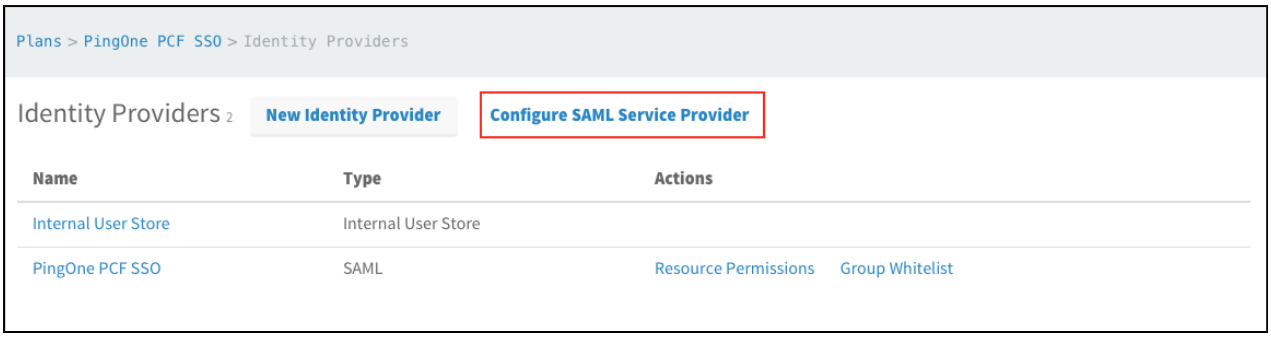
Configure PingOne Cloud as an Identity Provider

This topic describes how to set up PingOne Cloud as your identity provider by configuring SAML integration in both Pivotal Cloud Foundry (PCF) and PingOne Cloud.

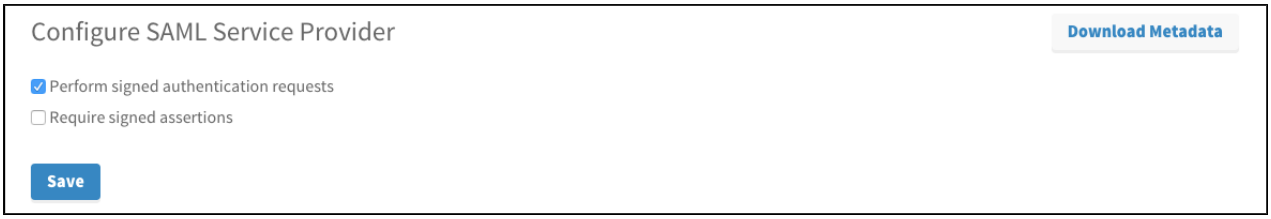
Set up SAML in PCF

1. Log into the Single Sign-On (SSO) dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.
2. Select your plan and click **Manage Identity Providers** on the dropdown menu.

3. Click **Configure SAML Service Provider**.



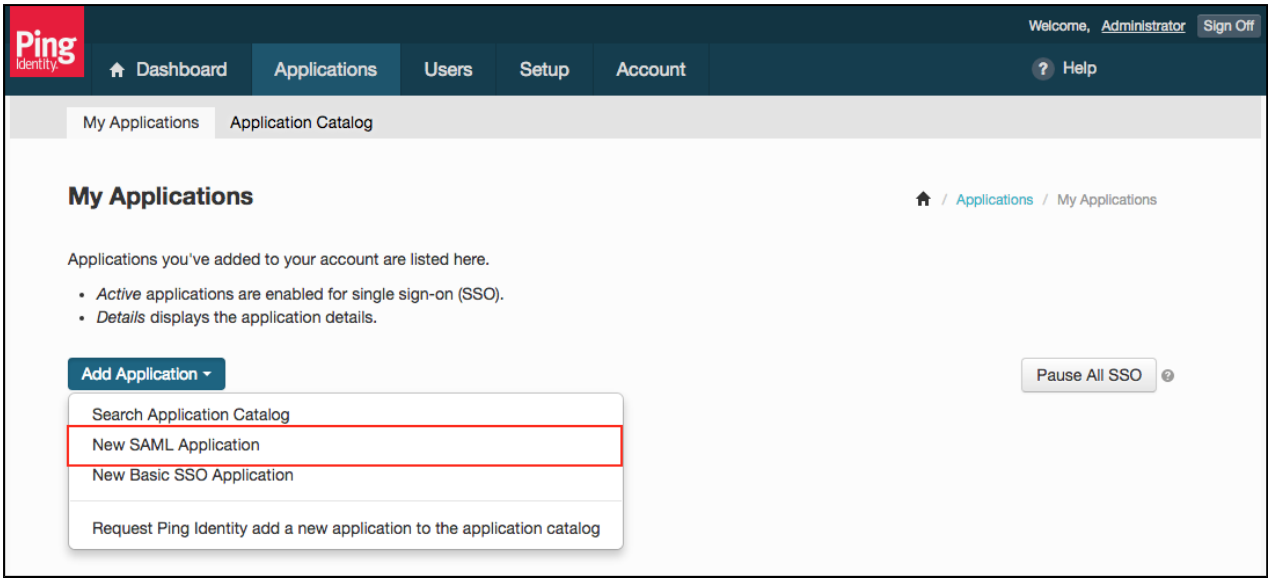
4. (Optional) Select **Perform signed authentication requests** to enforce SSO private key signature and identity provider validation.



5. (Optional) Select **Require signed assertions** to validate the origin of signed responses.
6. Click **Download Metadata** to download the service provider metadata.
7. Click **Save**.

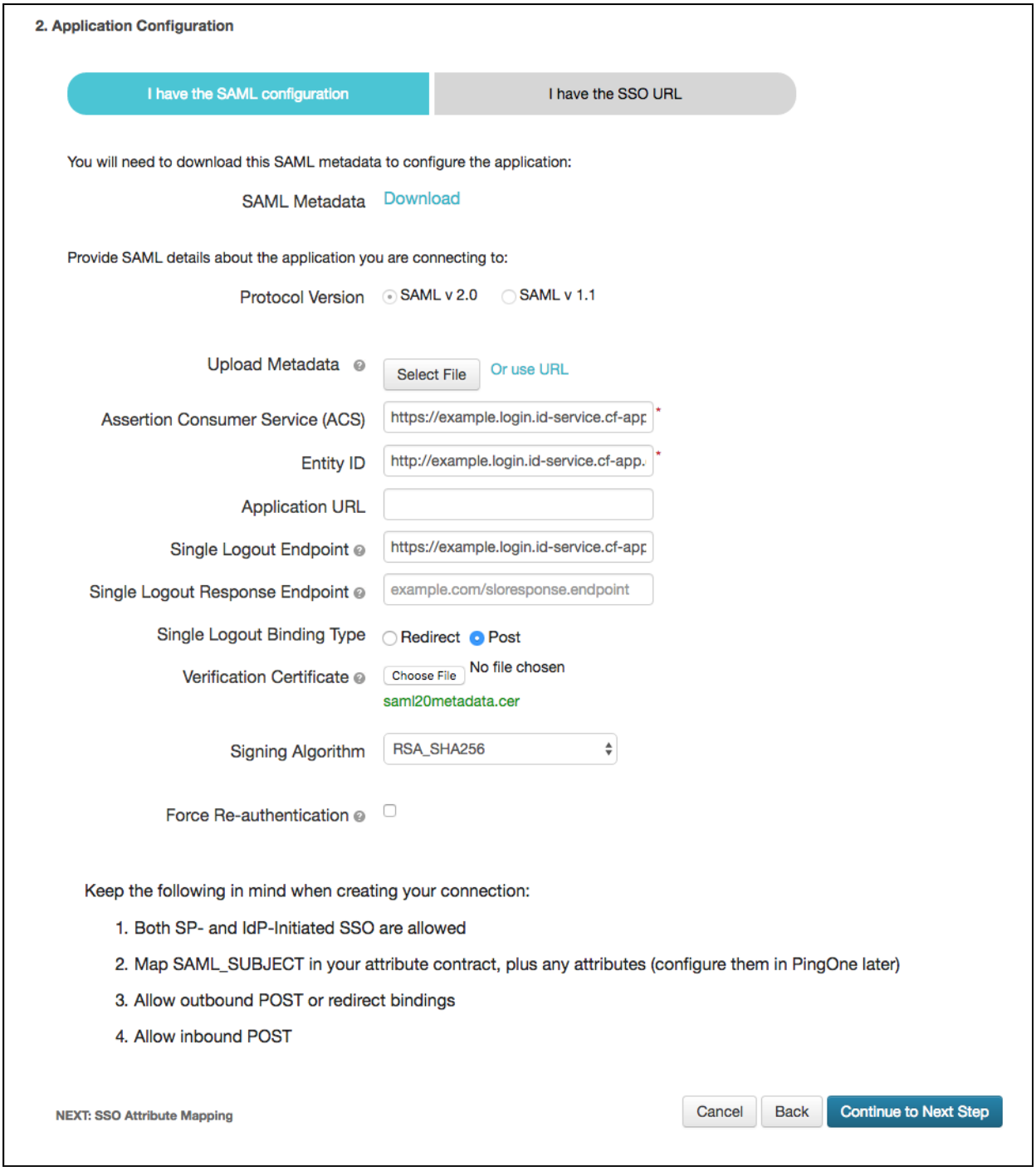
Set up SAML in PingOne Cloud

1. Sign in as a PingOne Cloud administrator.
2. Navigate to your application by clicking on the **Applications** tab.
3. Click the **Add Application** button and choose **New SAML Application**.



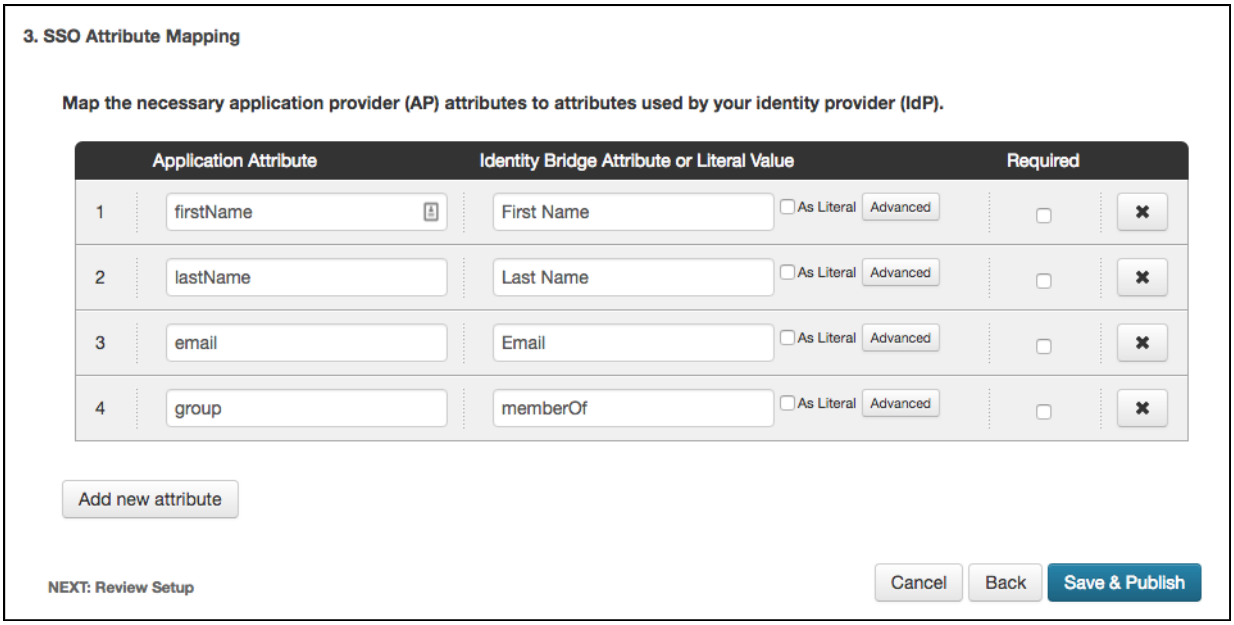
4. Enter the **Application Name**, **Application Description**, **Category** and any **Graphics**.

5. Click the **Continue to Next Step** button to configure SAML.



6. In the **Application Configuration** section, perform the following steps:
    a. Select **I have the SAML configuration**.
    b. For **SAML Metadata**, click **Download** to download the identity provider metadata.
    c. For **Protocal Version**, select **SAML v 2.0**.
    d. For **Upload Metadata**, click **Select File** and select the service provider metadata.
    e. Click the **Continue to Next Step** button.

7. (Optional) Under **SSO Attribute Mapping**, specify any application or group attributes that you want to map to users in the ID token.



8. Click the **Save & Publish** button followed by the **Finish** button.

Configure a Single Sign-On Service Provider

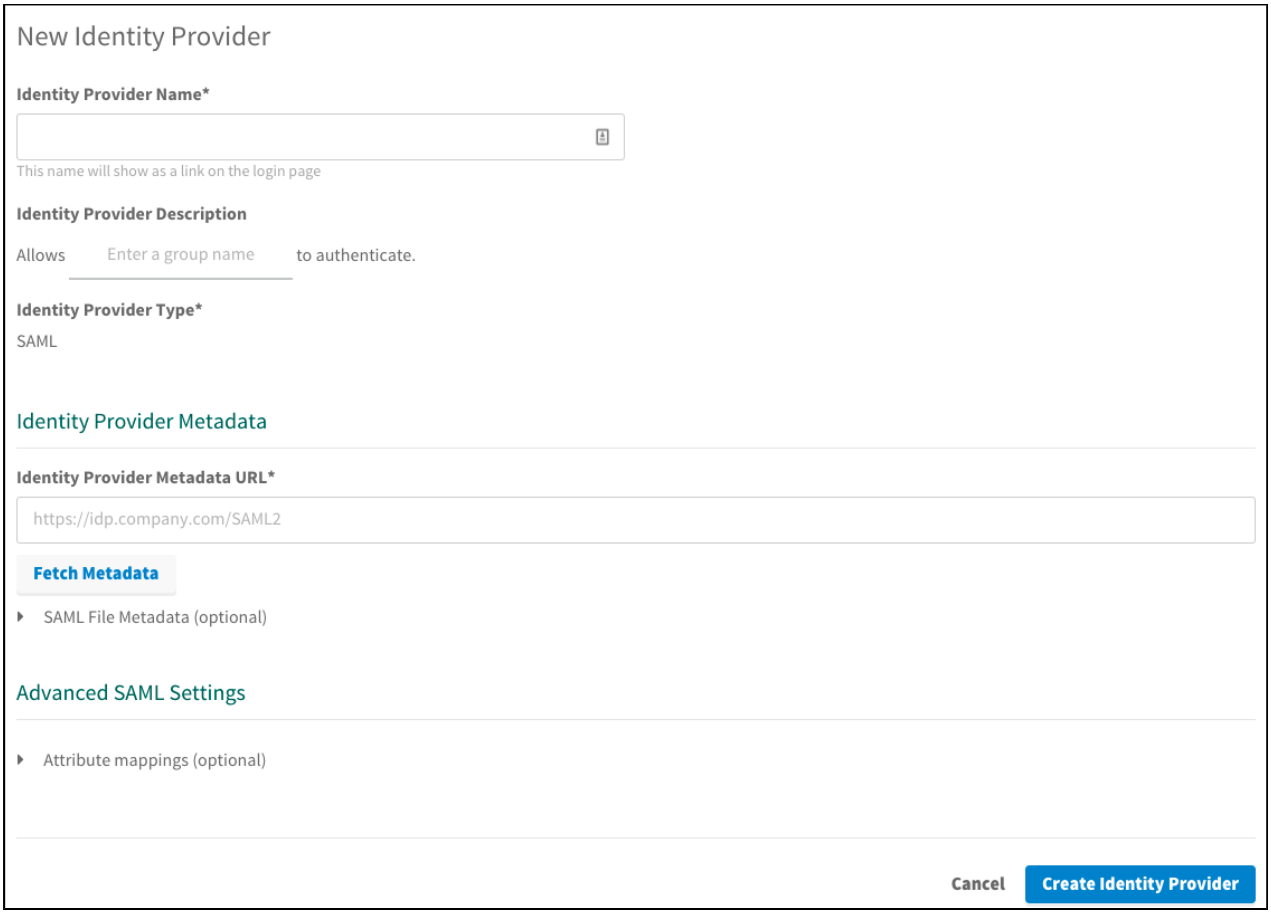
This topic describes how to add an external identity provider to your Pivotal Single Sign-On (SSO) service plan.

Setting up SAML

1. Log into the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` as a Plan Administrator.

2. Select your plan and click **Manage Identity Providers** on the dropdown menu.

3. Click **New Identity Provider** to create a new identity provider.



4. To create a new identity provider, perform the following steps:

    a. Enter an identity provider name into **Identity Provider Name**.
    b. (Optional) Enter a description into **Identity Provider Description**.
    c. Click **SAML File Metadata (optional)** follwed by clicking the **Upload Identity Provider Metadata** button to upload your metadata XML.
    d. (Optional) Under **Advanced SAML Settings**, click **Attribute Mappings** to enter the mappings.

5. Click **Create Identity Provider**.

6. Click **Resource Permissions**.

7. Click **New Permissions Mapping** and perform the following steps:

    a. Enter a **Group Name**.
    b. For **Select Permissions**, select the permissions that the members of the group from the external identity provider should have access to.

8. Navigate to the identity provider list.

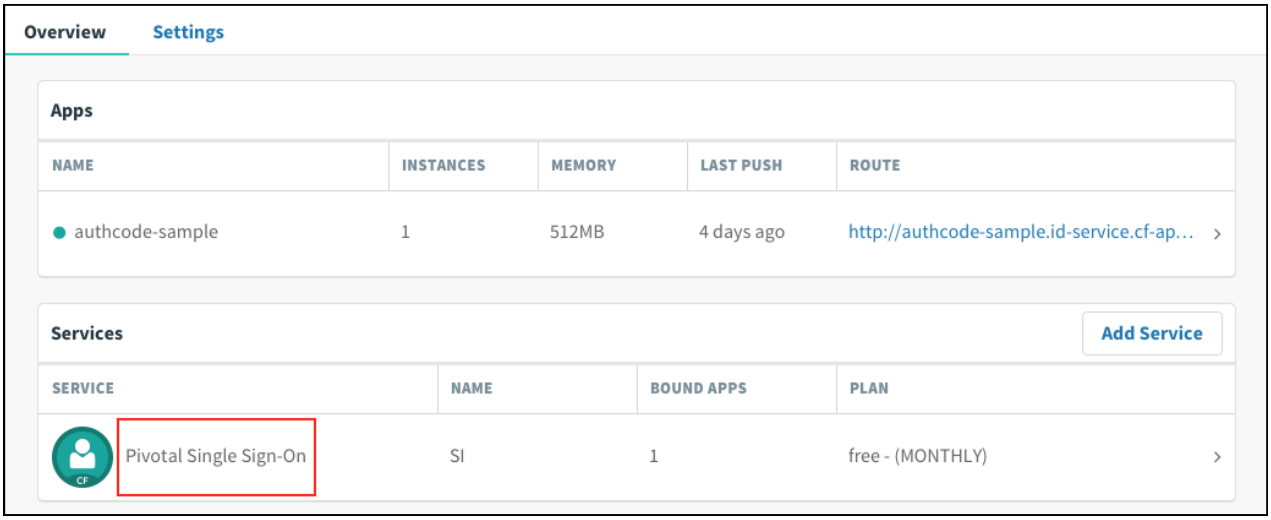
9. Click **Group Whitelist** and enter the group names from the external identity provider that should be propagated in the ID token.

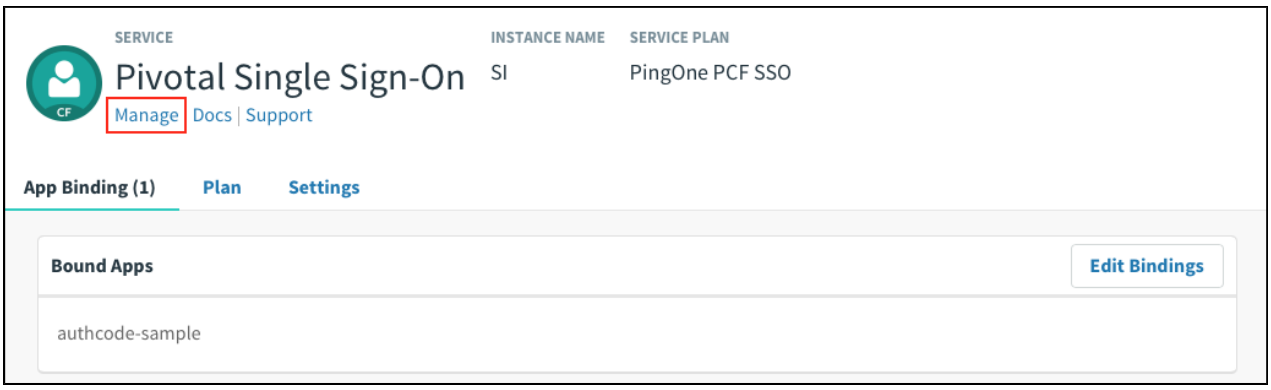
Testing

This topic describes how an administrator can test the connection between SSO and PingOne Cloud. An administrator can test both service provider and identity provider connections.

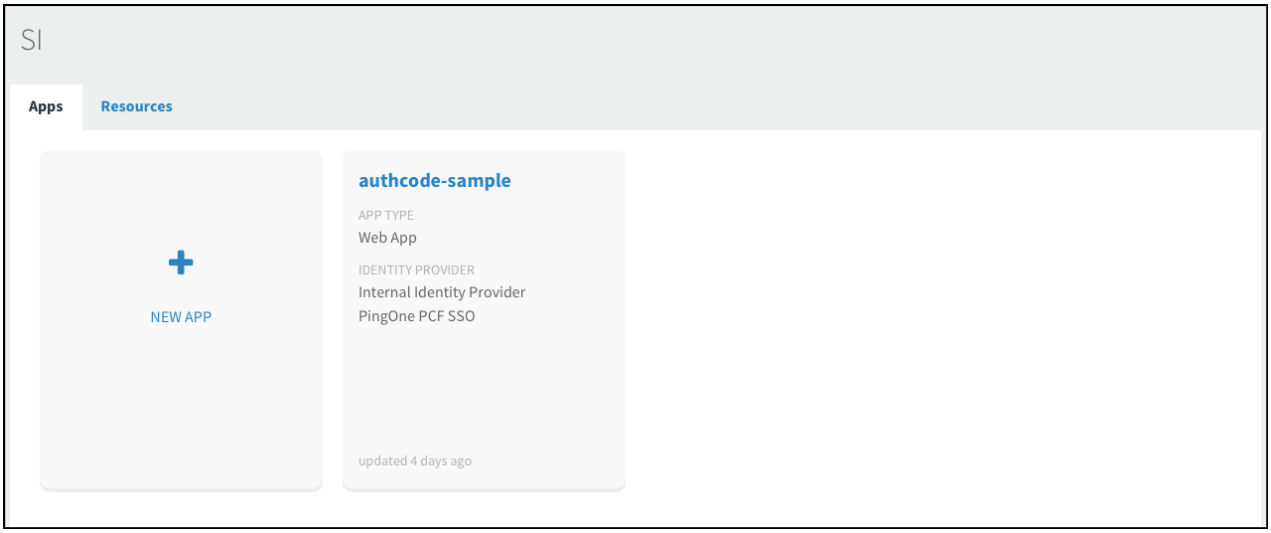
Test Your Service Provider Connection

1. Log in to Apps Manager at `https://apps.YOUR-SYSTEM-DOMAIN` and navigate to the organization and space where your application is located.
2. Under **Services**, locate the service instance of the Single Sign-On (SSO) plan bound to your application. Click on the service instance and click **Manage**.

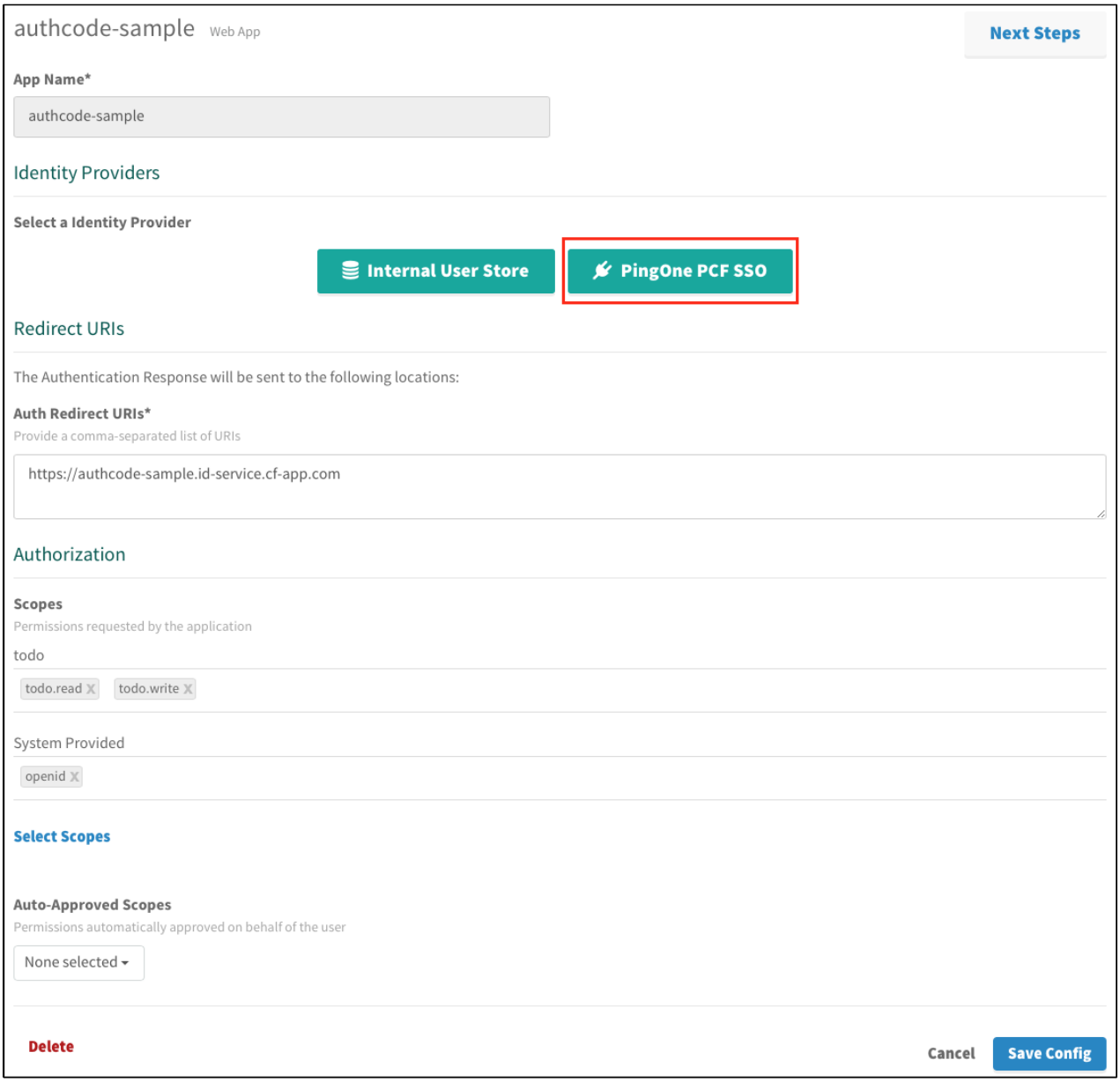




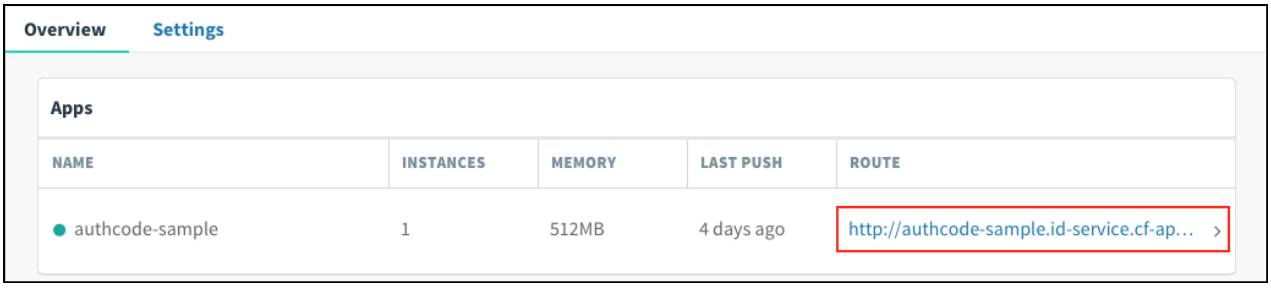
3. Under the **Apps** tab, click your application.



4. Under **Identity Providers**, select the PingOne identity provider.

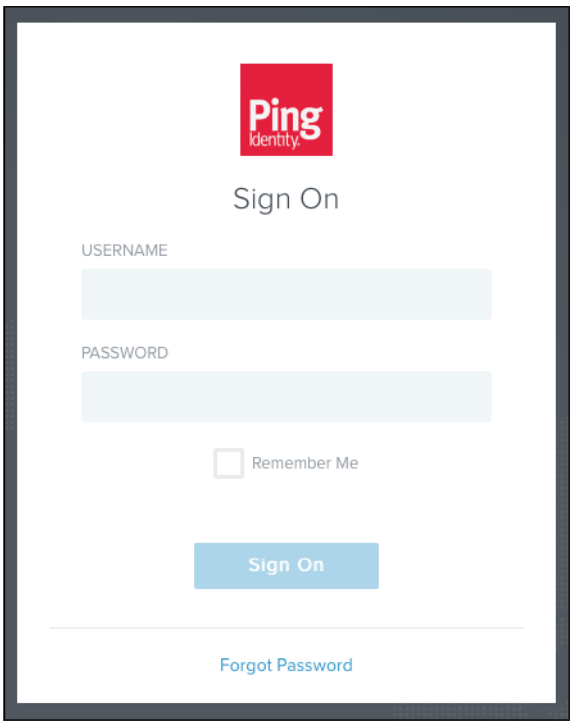


5. Return to Apps Manager and click on the URL below your application to be redirected to the identity provider to authenticate.

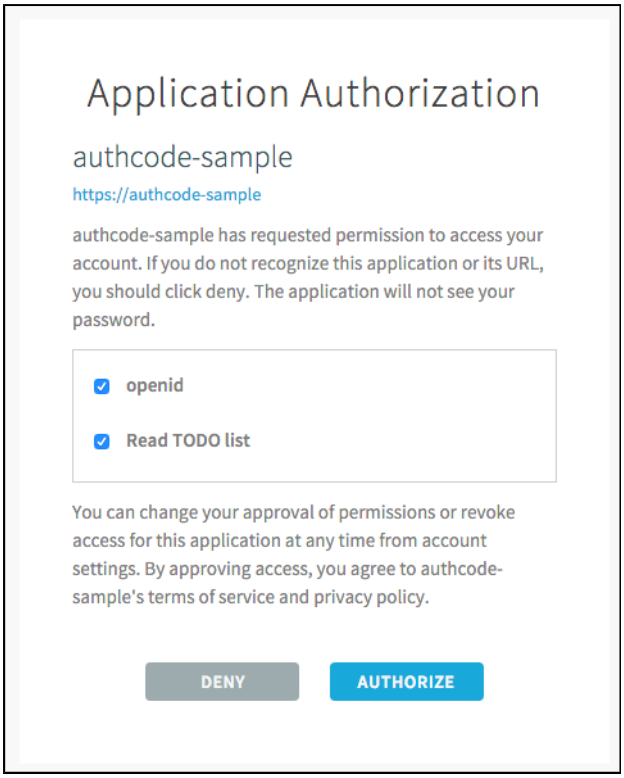


6. Click the link.

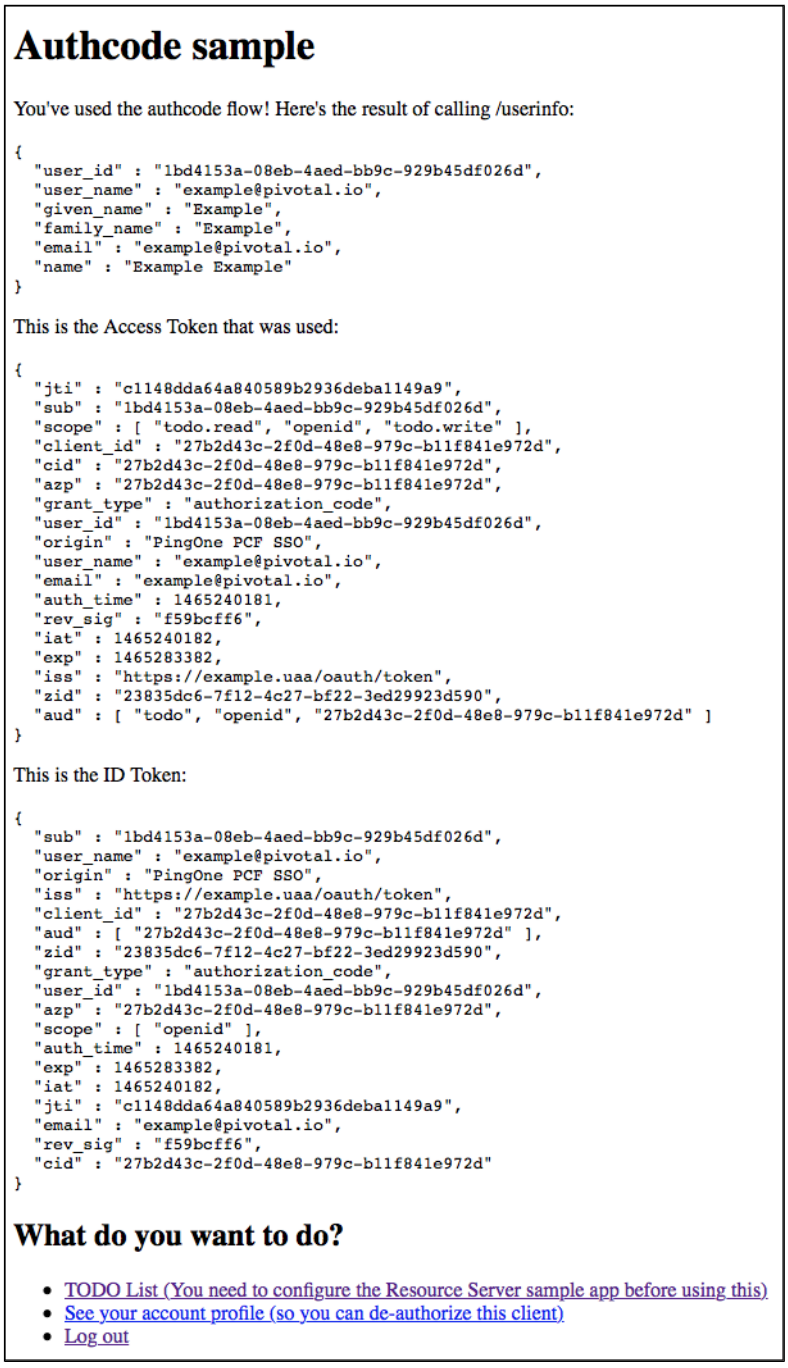
7. On the identity provider sign-in page, enter your credentials and click **Sign On**.



8. The application asks for authorization to the necessary scopes. Click **Authorize**.



9. The access token and ID token displays.



Authcode sample

You've used the authcode flow! Here's the result of calling /userinfo:


```
{
  "user_id" : "1bd4153a-08eb-4aed-bb9c-929b45df026d",
  "user_name" : "example@pivotal.io",
  "given_name" : "Example",
  "family_name" : "Example",
  "email" : "example@pivotal.io",
  "name" : "Example Example"
}
```


This is the Access Token that was used:


```
{
  "jti" : "c1148dda64a840589b2936deba1149a9",
  "sub" : "1bd4153a-08eb-4aed-bb9c-929b45df026d",
  "scope" : [ "todo.read", "openid", "todo.write" ],
  "client_id" : "27b2d43c-2f0d-48e8-979c-b11f841e972d",
  "cid" : "27b2d43c-2f0d-48e8-979c-b11f841e972d",
  "azp" : "27b2d43c-2f0d-48e8-979c-b11f841e972d",
  "grant_type" : "authorization_code",
  "user_id" : "1bd4153a-08eb-4aed-bb9c-929b45df026d",
  "origin" : "PingOne PCF SSO",
  "user_name" : "example@pivotal.io",
  "email" : "example@pivotal.io",
  "auth_time" : 1465240181,
  "rev_sig" : "f59bcff6",
  "iat" : 1465240182,
  "exp" : 1465283382,
  "iss" : "https://example.uaa/oauth/token",
  "zid" : "23835dc6-7f12-4c27-bf22-3ed29923d590",
  "aud" : [ "todo", "openid", "27b2d43c-2f0d-48e8-979c-b11f841e972d" ]
}
```


This is the ID Token:


```
{
  "sub" : "1bd4153a-08eb-4aed-bb9c-929b45df026d",
  "user_name" : "example@pivotal.io",
  "origin" : "PingOne PCF SSO",
  "iss" : "https://example.uaa/oauth/token",
  "client_id" : "27b2d43c-2f0d-48e8-979c-b11f841e972d",
  "aud" : [ "27b2d43c-2f0d-48e8-979c-b11f841e972d" ],
  "zid" : "23835dc6-7f12-4c27-bf22-3ed29923d590",
  "grant_type" : "authorization_code",
  "user_id" : "1bd4153a-08eb-4aed-bb9c-929b45df026d",
  "azp" : "27b2d43c-2f0d-48e8-979c-b11f841e972d",
  "scope" : [ "openid" ],
  "auth_time" : 1465240181,
  "exp" : 1465283382,
  "iat" : 1465240182,
  "jti" : "c1148dda64a840589b2936deba1149a9",
  "email" : "example@pivotal.io",
  "rev_sig" : "f59bcff6",
  "cid" : "27b2d43c-2f0d-48e8-979c-b11f841e972d"
}
```

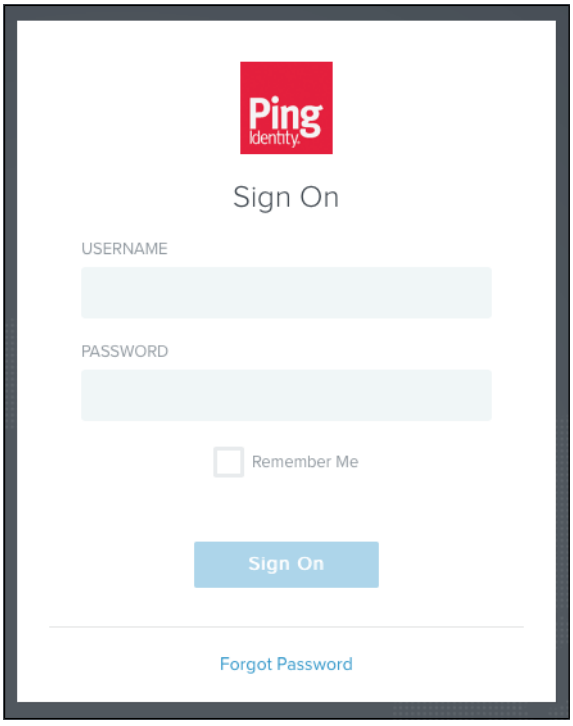

What do you want to do?

- TODO List (You need to configure the Resource Server sample app before using this)
- See your account profile (so you can de-authorize this client)
- Log out

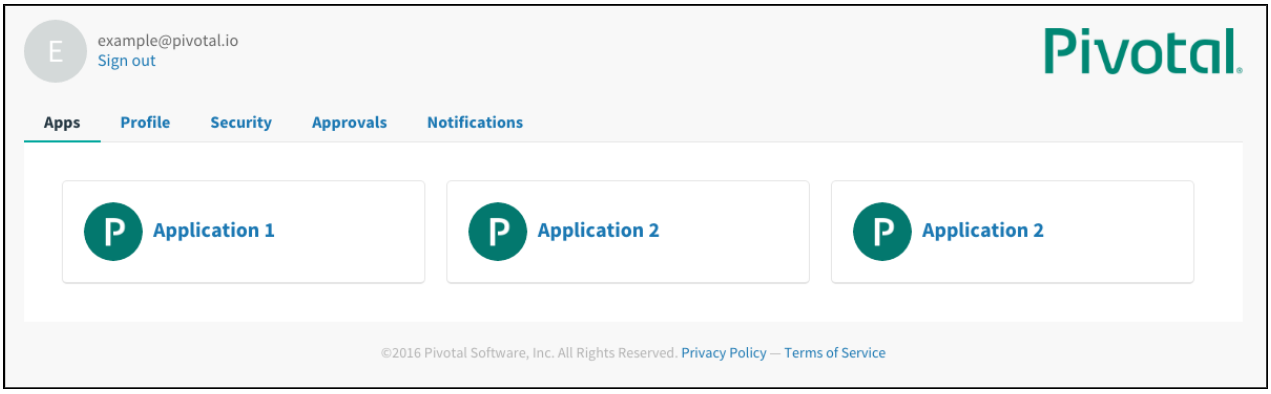
Test Your Identity Provider Connection

Note: SSO does not support identity provider-initiated flow into applications, but it does redirect the user to the User Account and Authentication (UAA) page to select applications assigned to the user.

1. Sign in to PingOne.



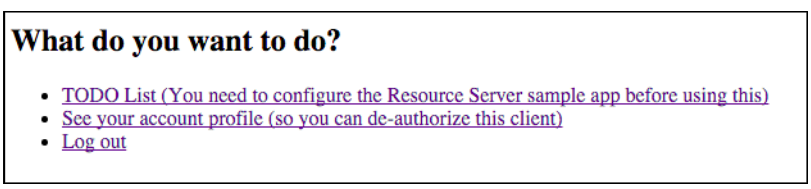
2. Navigate to your application and click it.
3. You are redirected to the page that lists applications you have access to.



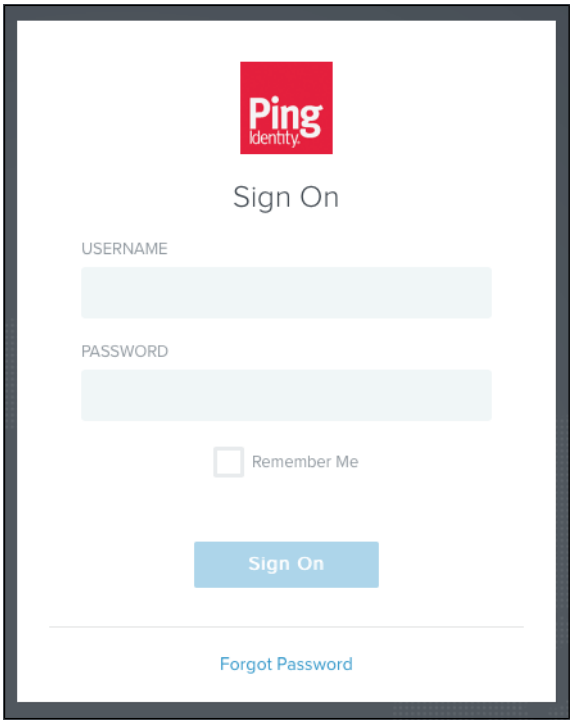
Test Your Single Sign-Off

Test single sign-off to ensure that when users log out of the application, they are logged out of PingOne as well.

1. Sign into the sample application. Information about the access and ID token displays, as well as the “What do you want to do?” section.
2. Under “What do you want to do?”, click **Log out**.



3. You are logged out and redirected to the PingOne login page.

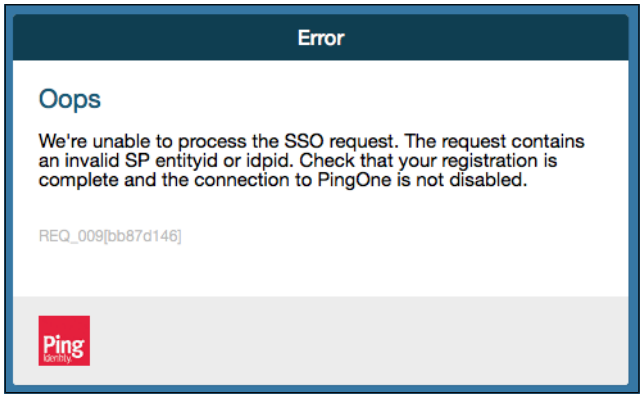


Troubleshooting

This topic describes how to resolve common errors that arise when configuring a single sign-on partnership between PingOne Cloud and Pivotal Single Sign-On (SSO).

Error

Symptom:

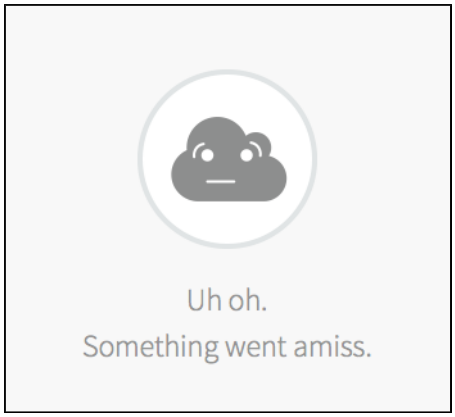


Explanations:

- Single Sign-On is disabled on PingOne.
- The service provider Entity ID is misconfigured on PingOne.
- The identity provider Single Sign-On URL is misconfigured in the SSO plan settings.

Something went amiss

Symptom:



Explanation:

- The service provider Assertion Consumer Service (ACS) is misconfigured on PingOne.

Metadata Not Found

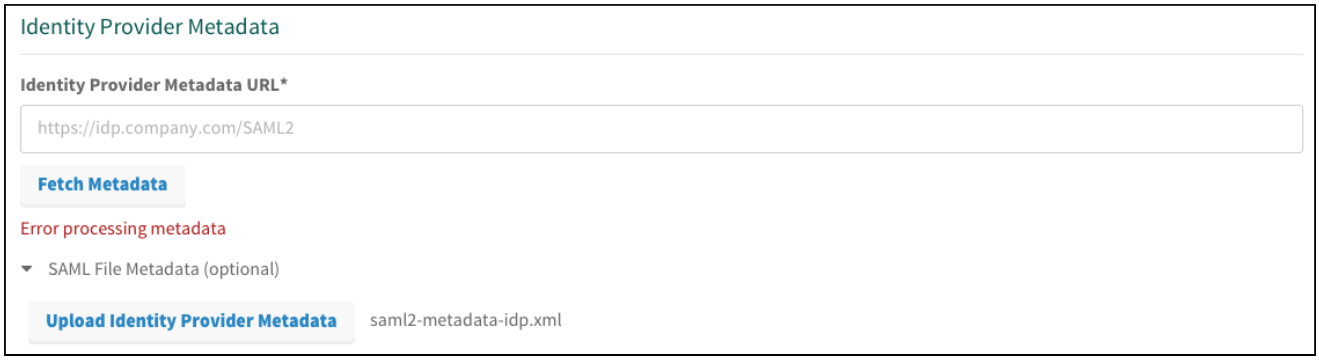
Symptom:

Explanation:

- The identity provider Entity ID is misconfigured in the SSO plan settings.

Missing Name ID

Symptom:



Explanation:

- The identity provider metadata is missing configurations for Name ID. See Configure Identity Provider Metadata.

Release Notes

View Release Notes for Another Version

To view the release notes for another product version, select the version from the drop-down list at the top of this page.

v1.2.x

v1.2.2

Release date: 14 October 2016

- PCF updated stemcell to 3263.7. This release bumps the Ubuntu stemcell for USN-3099-2: Linux kernel (Xenial HWE) vulnerabilities.

v1.2.1

Release date: 20 September 2016

- PCF updated stemcell to 3263. This is a security upgrade to patch CVE.

v1.2.0

Release date: 16 September 2016

What's New

Note: The Single Sign-On service tile works with the current and future versions of Pivotal Elastic Runtime.

- The SSO v1.2.x tiles are compatible with PCF v1.8.x or greater.

- Single Sign-On (SSO) for Pivotal Cloud Foundry® (PCF) provides the ability to create admin clients. Admin Clients can be used to:
  - Create, modify and delete identity providers
  - Create, modify and delete clients
  - Create, modify and delete users
  - Create, modify and delete groups/resources
- SSO provides the ability for administrators to disable internal authentication.
- SSO provides the ability for administrators to prevent users from creating new accounts and resetting their passwords.
- SSO provides the ability for administrators to specify zone token expiry.
- SSO provides the ability for developers to configure Application Settings including App Launch URL, App Icon and Show on homepage.
- SSO provides the ability for developers to select identity providers when binding an application.
- SSO introduces whitelabeling support for the following properties set in Operations Manager:
  - Logo
  - Header accent color
  - Footer text
  - Footer links