

Table of Contents

Table of Contents	1
Getting Started with Pivotal Single Sign-On	2
Adding Users to a Single Sign-On Service Plan	5
Administering Pivotal Single Sign-On	7
Choosing an Application Type	10

Getting Started with Pivotal Single Sign-On

This documentation describes installing and configuring the [Pivotal Single Sign-On](#) service.

Product Snapshot

Current [Pivotal Single Sign-On](#) for Pivotal Cloud Foundry® Details

Version: 1.0.10

Release Date: 2016-03-17

Software component version: Pivotal Single Sign-On 1.0.10 Installer based on Elastic Runtime 1.6.x

Compatible Ops Manager Version(s): 1.6.x

Compatible Elastic Runtime Version(s): 1.6.x

vSphere support? Yes

AWS support? Yes

OpenStack support? No

Step 1: Install the Service Tile in Ops Manager

1. Download the Pivotal Single Sign-On product tile from [Pivotal Network](#).
2. Upload the product tile to your Ops Manager installation. See [Adding and Deleting Products](#).
3. Click **Add** next to the uploaded product description in the Available Products view to add this product to your staging area.
4. Click **Apply Changes** to deploy the service.

Step 2: Login to the SSO Dashboard

1. After Ops Manager has finished applying your changes, login to the SSO dashboard at `https://p-identity.YOUR-SYSTEM-DOMAIN` using your UAA administrator credentials. You can find those credentials in your **Pivotal Elastic Runtime** tile under the **Credentials** tab.
2. From the SSO dashboard, create a new Single Sign-On Service Plan by clicking **New Plan**.

P
Pivotal Single Sign-On

Plans > Test

Test

Plan Name*

Description*

This will appear as a plan feature in the Apps Manager Marketplace

Auth Domain* `https://appcom.login.sys.poodle.wild.cf-app.com`

Instance Name*

This will appear as a title on the Sign In page

Org Visibility
☒ sandbox

Cancel Save Plan

- Choose a **Plan Name**, and enter a **Description**.
- Enter a **Auth Domain**, which is the URL your users will enter to access their Single Sign-On Service Plan.
- Choose an **Instance Name**. The instance name appears on the login page and in other user-facing content, such as email communications.
- Under **Org Visibility**, select which organizations should have access to your Single Sign-On Service Plan. If you do not select any orgs, that plan will be completely hidden in the Services Marketplace.
- Click **Save Plan**. Your new plan will appear in the Apps Manager Marketplace to users in the organizations you have selected. You can also access the Services Marketplace by entering `cf marketplace` in a terminal window.



Note: You can create and configure service plans at any time. However, you must create at least one plan in order to register and use the Service with applications on your [Pivotal Cloud Foundry®](#) installation.

Step 3: Update SSL Certificate

As described in the above section, every service plan corresponds to a tenant. Each service plan is accessible at a tenant-specific URL in the format `https://<AUTH-DOMAIN>.login.<SYSTEM-DOMAIN>`.

You must update the SSL Certificate for the domains listed below. In addition, depending on your infrastructure and type of load balancer, update your load balancer configuration for the following domains:

- *.<SYSTEM-DOMAIN>
- *.<APPS-DOMAIN>
- *.login.<SYSTEM-DOMAIN>
- .uaa.<SYSTEM-DOMAIN>

Step 4: Create Service Instances

After you have made the plans visible in the relevant PCF organizations, create a service instance to use the service plan within a space. Refer to the [Manage Service Instances](#) section of the *Administering Pivotal Single Sign-On* topic for more information about how to create a service instance within orgs and spaces.

While service instances are accessible only within a space, service plans are visible across spaces and orgs. Every plan needs a service instance in every space containing applications it will access.

If you have a plan that allows Single Sign-On to applications in multiple spaces, you must create a service instance for the plan in each of the spaces.

Step 5: Configure External Identity Providers

Configure the external identity providers for your PCF platform. External identity providers must be SAML 2.0 compliant. Refer to [Administering Pivotal Single Sign-On](#) for more information about configuring External Identity Providers.

Step 6: Register your Applications

Secure your applications in one of the following ways:

- Bind your application to a service instance from the Apps Manager Console.
- Register your application from the Pivotal Single Sign-On service dashboard.

When you bind an application with the Pivotal Single Sign-On service, an OAuth Client is registered with the Single Sign-On server. This OAuth Client acts as an OAuth 2.0 Authorization Server and issues tokens.

If you bind your application from the Apps Manager Console, then you are choosing the following defaults:

- The application type is Web App.
- The User Store Connection is an Internal User Store. This means that users are stored in the user store provided by Pivotal Single Sign-On.

If you register your application with the Pivotal Single Sign-On service dashboard, you gain access to a larger selection of application types. You can also configure the User Store Connection to an external identity provider with SAML.

Step 7: Enable OAuth 2.0 for your applications

Your application must be able to request and validate an OAuth or OpenID Connect token. Refer to the [Application Integration](#) section for more information.

Adding Users to a Single Sign-On Service Plan


You cannot add users to Service Plans from the Single Sign-On 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 your Service Plans.

Step 1: Client Registration

1. Target your system domain.

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

 **Note:** If you do not have SSL configured, you can turn off SSL validation with `--skip-ssl-validation`

2. Fetch your admin client token.

```
$ uaac token client get admin
Client secret:
```

3. When prompted with `Client secret`, enter your **Admin Client Credentials** located in the **Credentials** tab of your **Pivotal Elastic Runtime** tile.

4. Update client registrations for `identity` and add `password` as a supported authorization grant type.

```
$ uaac client update identity --authorized_grant_types "refresh_token,password,client_credentials,authorization_code"
```

Step 2: Client Creation

1. Target your system domain.

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

2. Use the identity client and the administrator user credentials to retrieve a token which allows client creation in a given identity zone.

```
$ uaac token owner get
```

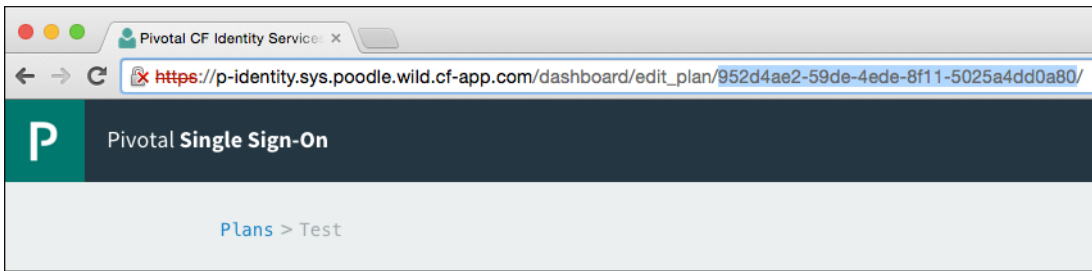
3. When prompted with `Client ID`, type `identity` and press enter.

4. For `Client secret`, enter your **Identity Client Credentials**, located in the **Credentials** tab of your **Pivotal Elastic Runtime** tile.

5. For `User name`, enter `Admin`.

6. Enter your `Password`, which is located in the **Admin Credentials** under the **Credentials** tab of your **Pivotal Elastic Runtime** tile.

7. Find the Identity Zone ID of your Service Plan by logging into the SSO dashboard, selecting **Edit Plan**, and copying the Identity Zone ID from the URL.



The

highlighted text in the example URL above is the Identity Zone ID of the Service Plan.

- Execute the following command, replacing `YOUR-IDENTITY-ZONE-ID` with the Identity Zone ID of your Service Plan. You can also replace `service_plan_admin_client` with a different name for your client, and `secret` with a different client secret.

```
uaac curl -k -H"Accept:application/json" -H"Content-Type:application/json" -H"X-Identity-Zone-Id:YOUR-IDENTITY-ZONE-ID"
```

Step 3: User Creation

- Target the `auth domain` of your Single Sign-On 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
```

- Fetch the token for your Service Plan admin client.

```
$ uaac token client get service_plan_admin_client
Client secret:
```

- When prompted with `Client secret`, enter the `secret` from the `uaac curl` command above.
- 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
```

- (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
```

Administering Pivotal Single Sign-On

Manage Service Plans

Pivotal Single Sign-On is a multi-tenant service. A Pivotal Single Sign-On tenant corresponds to a service plan. As an enterprise adopting Pivotal Single Sign-On, you might want to segregate your tenants into 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

You can create new Single Sign-On Service Plans at any time from the SSO dashboard by logging in with your administrator credentials to `p-identity.YOUR-SYSTEM-DOMAIN`.

Manage Service Instances

Service instances are created based on the service plan and are visible within the confines of a space.


To create a new service instance:

1. Log in to your Apps Manager instance at `https://console.YOUR-SYSTEM-DOMAIN` as a Space Developer.
2. Navigate to the Marketplace and select **Pivotal Single Sign-On**.
3. Select your service plan.
4. Enter an **Instance Name**.
5. Choose a space for the instance from the **Add to Space** dropdown menu. The default is 'development'.
6. Choose an application to bind the service instance to from the **Bind to App** dropdown menu. This option defaults to [do not bind].

Manage User Stores

Follow the steps below to configure an External Identity Provider based on SAML 2.0:

1. Navigate to the Plan Administration User Interface at `https://p-identity.YOUR-SYSTEM-DOMAIN`.
2. Log in with your administrator credentials.
3. Navigate to the Service Plan and select **Manage User Stores** from the menu.
4. Click **Service Provider Metadata** to download the SAML Service Provider metadata for the selected Service Plan.
5. Import the Service Provider Metadata on the Identity Provider console.
6. Configure the Identity Provider. These steps vary based on the Identity Provider being configured.
 - a. Create the Remote Service Provider Entity Object. This can be done by importing the Service Provider Metadata from Step 4.
 - b. Set the NameID format as email address and map the value as the user's email address.

 **Note:** Pivotal Single Sign-On service only supports email as the NameID.

- c. Set the SAML SSO Binding Profile as `HTTP POST`.
- d. Only SP-Initiated SSO Transactions are allowed. Configure the same on the provider.
- e. Download the Identity Provider Metadata. If the Identity Provider exposes a URL for downloading the Metadata,

the same can be used.

7. Click **New User Store**.
8. Enter a User Store Name. This is visible on the application login page if more than one Identity Provider is associated with an application.
9. Supply the SAML Identity Provider metadata in one of the following ways:
 - a. Provide the metadata URL and click Fetch Metadata.
 - b. Click Upload Identity Provider Metadata to upload the XML metadata of your identity provider.

After you complete the steps above, the Entity ID, SSO URL, and NameID of your new user store are visible. In order to add users to the Internal User Store for a given Service Plan, follow the steps outlined in [Adding Users to a Single Sign-On Service Plan](#).

Manage Applications

Binding Applications running on Pivotal Cloud Foundry


1. Log in as a user with the role of a Space Developer.
2. Set the GRANT_TYPE Environment Variable for the Application depending on the Application Type. Refer to [Choosing an Application Type for Pivotal Single Sign-On](#).

Application Type	Grant Type Env Value
Web App	authorization_code
Native Mobile App	password
Single Page JavaScript App	client_credentials
Service-to-Service App	implicit

3. Bind the Application with the Service Instance.
4. Binding the Application creates an OAuth Client based on the Grant Type set.
5. Click the **Manage** link under the Pivotal Single Sign-On service instance to launch the service dashboard.
6. Navigate to your Application.
7. Under User Store Connections, Internal User Store is selected by default. Switch it to use an External Identity provider for SSO and save the configuration.
8. Refer to the [Application Integration](#) section on how to enable Single Sign-On on the application side.

Registering External Applications (Applications not hosted on Pivotal Cloud Foundry)

1. Click the **Manage** link under the Pivotal Single Sign-On service instance to launch the service dashboard.
2. Click **New App**.
3. Enter an **App Name**.
4. Select an **Application Type**. Refer to [Choosing an Application Type for Pivotal Single Sign-On](#) to determine the application type you should choose.
5. User Store Connections defaults to `Internal User Store`, but you can choose to use one or more SAML User Store Connections. This option governs the users who are allowed to authenticate to the application.

 **Note:** This option is available for all application types except the Service-to-Service App.


6. If your Application Type is 'Web App', enter a whitelist of valid Auth Redirect URIs beneath Redirect URIs. The redirect query parameter specified on the OAuth request must match the URIs specified in this list. Otherwise, the Pivotal Single Sign-On service rejects the request.
7. Enter the **Scopes** that this application can request. This field defaults to `openid`. Scopes are permissions that the application requests on the user's behalf. 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.
8. Select scopes pertaining to company owned and managed applications as **Auto-Approved Scopes**. All defined scopes in the **Scopes** field are available for selection as Auto-Approved Scopes. Scopes selected as Auto-Approved do not require explicit authorization from the end user. To determine whether a scope should be 'Auto-Approved', decide whether the application you are binding is trusted. Company owned and managed applications fall under this category. Scopes that pertain to permission or actions on resources from applications external to PCF are not candidates for auto-approval.
9. After you click **Create App**, the **Next Steps** view displays. These steps describe the important endpoints required for application integration. Refer to the [Application Integration](#) section for more details.

Application Integration


The Pivotal Single Sign-On service is based on the OAuth protocol. Ensure that the applications you intend to secure with Pivotal Single Sign-On are OAuth aware.

Java-based sample applications, created using [Spring Boot](#), are available for all four application types. You can find information about configuring and running these sample applications [here](#). These applications use the SSO Service Connector, which auto configures the application for OAuth. The only step involved after binding the application with the SSO Service is to restart the application for the new SSO configuration to take effect.

For non-Java applications, the applications need to be made OAuth aware. The following information needs to be supplied to the application after the SSO service bind.

 **Note:** All the information above is available on the Next Steps page.

- App ID, also known as OAuth Client ID
- App Secret, also known as OAuth Client Secret
- OAuth Authorization URL: This is the endpoint for Client Authorization
- OAuth Token URL: This is the end point for token retrieval

 **Note:** The **Application ID** and **Application Secret** are both available on the Next Steps page of the Pivotal Single Sign-On service dashboard.

Validation of the token requires that you 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 end point specified on the Next Steps Page.
2. The value of `aud` in the token matches your Application ID.
3. The value of `iss` matches `https://AUTH-DOMAIN.uaa.YOUR-SYSTEM-DOMAIN/oauth/token`.
4. The expiry time (`exp`) of the token has not passed.

Choosing an Application Type

The **Application Type** you select in the Pivotal Single Sign-On dashboard depends on the type of application and the type of user your application authenticates.

- If your application is authenticating end users, refer to the table below. Find your application type in the first column to determine which **Application Type** to select in the Pivotal SSO dashboard.

APPLICATION TYPE	PIVOTAL SSO APPLICATION TYPE	OAUTH GRANT TYPE EQUIVALENT
Web	Web App	Authorization code
Native Mobile, Desktop, or Command Line	Native Mobile App	Resource Owner Password
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.

- If your application is trying to access other services or APIs on its own behalf rather than on the user's behalf, the application must authenticate itself using the Single Sign-On service. Choose the **Service-to-Service App** type. This corresponds to the "Client Credentials" grant type in OAuth.