Security and Pivotal Single Sign-On

Goal

Set up a SSO service plan and obtain and access token. Secure an application with the access token.

Approximate time: 20 minutes

Exercises

Create a SSO Service Plan/UAA Tenant

1. In Ops Manager, verify that the SSO Service is installed. There should be a Pivotal Single Sign-On tile. If not, you will need to install it before starting this lab.
2. In Apps Manager, verify that you have a space in the `system` org named `identity-service-space`. Click on the link to the `identity-service-broker`. The link should be `p-identity.[system_domain]`.
3. Log in with Ops Manager > Pivotal Elastic Runtime > Credentials > UAA Admin Credentials. This brings up the dashboard for the Pivotal Single Sign-On (SSO) service.
4. Click on `New Plan` to create a SSO service plan/tenant. Name the plan `CustomerAppSSOPlan1`. If you are sharing a Cloud Foundry installation, append your initials to the plan name.
5. For Description, enter `Customer apps SSO 1`.
6. For Auth Domain, enter 'sso1'. If you are sharing a Cloud Foundry installation, append your initials to the Auth Domain name.

7. For Instance Name, enter `Customer App SSO Plan 1`.
8. Under `Org Visibility`, select your org. You could also use `cf enable-service-access` to make the SSO Service available to your org.
9. Click `Create Plan`. You have now created a tenant on the UAA/Pivotal Single Sign-On service.
10. Click on the plan, then `Manage User Stores`. Notice that you are using the Internal User Store.

Congratulations, you have created a SSO service plan/UAA tenant and made it available to your org.

Create a Service Instance

1. In Apps Manager, click on the Marketplace. Notice that your plan is available under Pivotal Single Sign-On. Notice that the first item under Plan Features is what you typed in `Description` when creating the plan.
2. Click `Select this plan` Create a SSO Service instance in your development space. Name the instance `SSO instance 1`. You do not need to bind it to an application yet. Click `Add` to bind the service to your space.

Congratulations, you have created an SSO service instance in your development space.

Create an OAuth Client on the Service Plan/UAA Tenant

1. Click `Manage` on the SSO Service instance to bring up the dashboard for the service instance.
2. Click on `New App` to begin the process of creating an OAuth Client on the UAA tenant that you just created.
3. Name the app `client_credentials OAuth Client 1`.
4. Click on the `Service-to-Service App` circle. This will create an OAuth client on the UAA with grant_type="client_credentials".
5. Click `Create App` to create an OAuth client of type client_credentials on your new tenant of the UAA.

6. The `App ID` textbox contains the ClientID of the OAuth client on the UAA tenant. Your OAuth-aware application will need this ID.
7. The `App Secret` textbox contains the ClientSecret, which is also needed by your OAuth-aware application. Notice that the secret is only displayed once.
8. Copy the App ID and App Secret into a text document. Separate the two values with a colon.
9. Convert the combined values into a base64-encoded value using https://www.base64encode.org/ This is what you will use in the curl statement below. Substitute [base64 encoded value] with this value.
10. You will also use the [OAuth Token URL] in the curl statement below.
11. As a test of the OAuth client on the UAA tenant, retrieve a token using the following curl command **make sure that there are not extra characters in the request**:

```
curl -i -H 'Content-Type: application/x-www-form-urlencoded' -X POST '[OAuth Token URL]' -d
'grant_type=client_credentials' -H "Authorization: Basic [base64 encoded value]" -k
```

For example:

```
curl -i -H 'Content-Type: application/x-www-form-urlencoded' -X POST
'https://sso2.login.system.steve.ed.pcfdemo.com/oauth/token' -d 'grant_type=client_credentials' -H
"Authorization:
Basic MGE5MDFiNDctOWJkZC00NzI1LWE1ZDgtMzIzNjNkZGQ3ZjMxOjdkMGY3ZTUyLWRjYWEtNDY3NS1hOGQxLTNjZDE5OWRlZGMxMw==" -k
```

Congratulations, you have set up a service plan/tenant, an OAuth client on the UAA and obtained a token from the OAuth Client. If you have time, enable security on an app by binding it to the service instance and setting the GRANT_TYPE environment variable for the application to “client_credentials”. You would then make the application OAuth-aware. You can also use the Spring Boot sample apps provided here: https://github.com/pivotal-cf/identity-sample-apps

Last updated 2015-12-22 15:20:42 PST