

PingOne Advanced Identity Cloud guide

This guide is for customers using an agent-based integration model, with AM on-premise, or another on-premise access management solution. The guide provides an example of how to transition from on-premise access management to Advanced Identity Cloud without changing the architecture of the agent-based model.

Advanced Identity Cloud is described in the [PingOne Advanced Identity Cloud Docs](#).

Example installation for this guide

Unless otherwise stated, the examples in this guide assume the following installation:

- Java Agent installed on `https://agent.example.com:443/app`, in the `alpha` realm.
- An Advanced Identity Cloud tenant with the default configuration, as described in the [PingOne Advanced Identity Cloud documentation](#).

When using Advanced Identity Cloud, you need to know the value of the following properties:

- The URL of your Advanced Identity Cloud tenant. For example, `https://tenant.forgeblocks.com`.

The URL of the AM component of Advanced Identity Cloud is the root URL of your Advanced Identity Cloud tenant followed by `/am`. For example, `https://tenant.forgeblocks.com/am`.

- The realm where you work. The examples in this guide use `alpha`.

Prefix each realm in the hierarchy with the `realms` keyword. For example, `/realms/root/realms/alpha`.

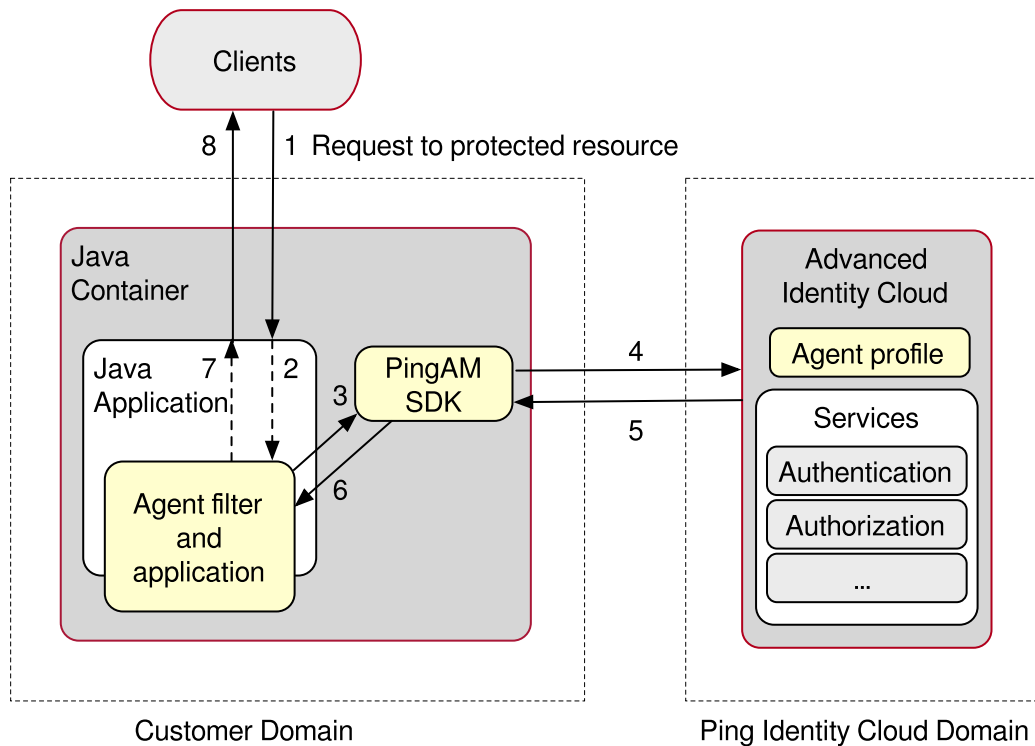
If you use a different configuration, substitute in the procedures accordingly.

About Java Agents and PingOne Advanced Identity Cloud

Advanced Identity Cloud simplifies the consumption of an identity platform. However, many organizations have business web applications and APIs deployed across multiple

clouds, or on-premise. This guide provides an example of how to use Java Agent with Advanced Identity Cloud, without changing the architecture of the agent-based model.

The following image illustrates the flow of an inbound request to a website, through a Java Agent, and the Java Agent's interaction with Advanced Identity Cloud to enforce resource-based policies.



For information, refer to the [Advanced Identity Cloud documentation](#).


Prepare for installation

Learn more about installing Java Agent in [Installation](#). Consider the following points for using the agent with Advanced Identity Cloud:


- Configure Advanced Identity Cloud and set up a policy before you install the agent. When you configure the agent in the Advanced Identity Cloud admin UI, you can select the policy.
- For environments with load balancers or reverse proxies, consider the communication between the agent and the Advanced Identity Cloud tenants, and between the agent and the client. Configure the environment **before** you install the agent.

Add a demo user in Advanced Identity Cloud


Add a user so you can test the examples in this guide.

1. In the Advanced Identity Cloud admin UI, select  **Identities > Manage > Alpha realm - Users**.
2. Add a new user with the following values:
 - **Username** : demo
 - **First name** : demo
 - **Last name** : user
 - **Email Address** : demo@example.com
 - **Password** : Ch4ng3!t

Create a policy set and policy in Advanced Identity Cloud

1. In the Advanced Identity Cloud admin UI, select  **Native Consoles > Access Management**. The AM admin UI is displayed.
2. In the AM admin UI, select **Authorization > Policy Sets > New Policy Set**, and add a policy set with the following values:
 - **Id** : PEP
 - **Resource Types** : URL
3. In the policy set, add a policy with the following values:
 - **Name** : PEP-policy
 - **Resource Type** : URL
 - **Resource pattern** : */**/*/*
 - **Resource value** : */**/*/*
4. On the **Actions** tab, add actions to allow HTTP GET and POST .
5. On the **Subjects** tab, remove any default subject conditions, add a subject condition for all Authenticated Users .

Create an agent profile in Advanced Identity Cloud

1. In the Advanced Identity Cloud admin UI, go to  **Gateways & Agents > New Gateway/Agent**, and add a Java Agent with the following values:
 - **Agent ID** : java-agent
 - **Password** : password
 - **Application URL** : https://agent.example.com:443/app
 - **Use Secret Store for password**: (Optional) Enable to use a secret store for the agent profile password.

Once enabled, the **Secret Label Identifier** field displays.

- **Secret Label Identifier:** Enter a value that represents the `identifier` part of the secret label for the agent. This value should clearly identify the agent (for example, `java-agent`). Advanced Identity Cloud uses the identifier to generate a secret label in the following format:
`am.application.agents.identifier.secret`.

Learn more in [Secret labels](#) and [Map ESV secrets to secret labels](#).

2. Click **Save Profile** and **Done**.
3. On the agent profile page, enable **Use Policy Authorization**, select a policy set to assign to the profile, and then click **Save**.

If a suitable policy set isn't available, select **Edit advanced settings** to edit or create one.

Secret Label Identifier changes

Advanced Identity Cloud maintains secret mappings when the **Secret Label Identifier** is changed as follows:

- If you update the **Secret Label Identifier**:
 - If no other agent shares that secret mapping, Advanced Identity Cloud updates any corresponding secret mapping for the previous identifier.
 - If another agent shares that secret mapping, Advanced Identity Cloud creates a new secret mapping for the updated identifier and copies its aliases from the previously shared secret mapping.
- If you delete the **Secret Label Identifier**, Advanced Identity Cloud deletes any corresponding secret mapping for the previous identifier, provided no other agent shares that secret mapping.

Enforce policies decisions from Advanced Identity Cloud

This example sets up Advanced Identity Cloud as a policy decision point for requests processed by Java Agent. For more information about Java Agent, refer to the [User guide](#).

1. Using the [Advanced Identity Cloud documentation](#), log in to Advanced Identity Cloud as an administrator.
2. Make sure that you are managing the `alpha` realm. If not, [switch realms](#).
3. [Create a policy set and policy](#).
4. [Create an agent profile](#).

When a policy set is assigned to the agent profile during creation, the agent uses that policy set. If a suitable policy set isn't available during creation, select **Edit advanced settings** to edit or create one and assign it to the agent profile.

5. Test the setup:

- a. Go to `https://agent.example.com:443/app`. The Advanced Identity Cloud login page is displayed.
- b. Log in to Advanced Identity Cloud as user `demo`, password `Ch4ng3!t`, to access the web page protected by the Java Agent.

Was this helpful?  

Copyright © 2010-2024 ForgeRock, all rights reserved.