ForgeRock Identity Cloud guide

ForgeRock® Identity Platform serves as the basis for our simple and comprehensive Identity and Access Management solution. We help our customers deepen their relationships with their customers, and improve the productivity and connectivity of their employees and partners. For more information about ForgeRock and about the platform, see https://www.forgerock.com.

About this guide

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

The examples in this document are based on an available version of Identity Cloud. As Identity Cloud evolves, the examples in this document will be updated to reflect the changes.

Example installation for this guide

Identity Cloud is described in the Identity Cloud Docs.

Find the value of the following properties:

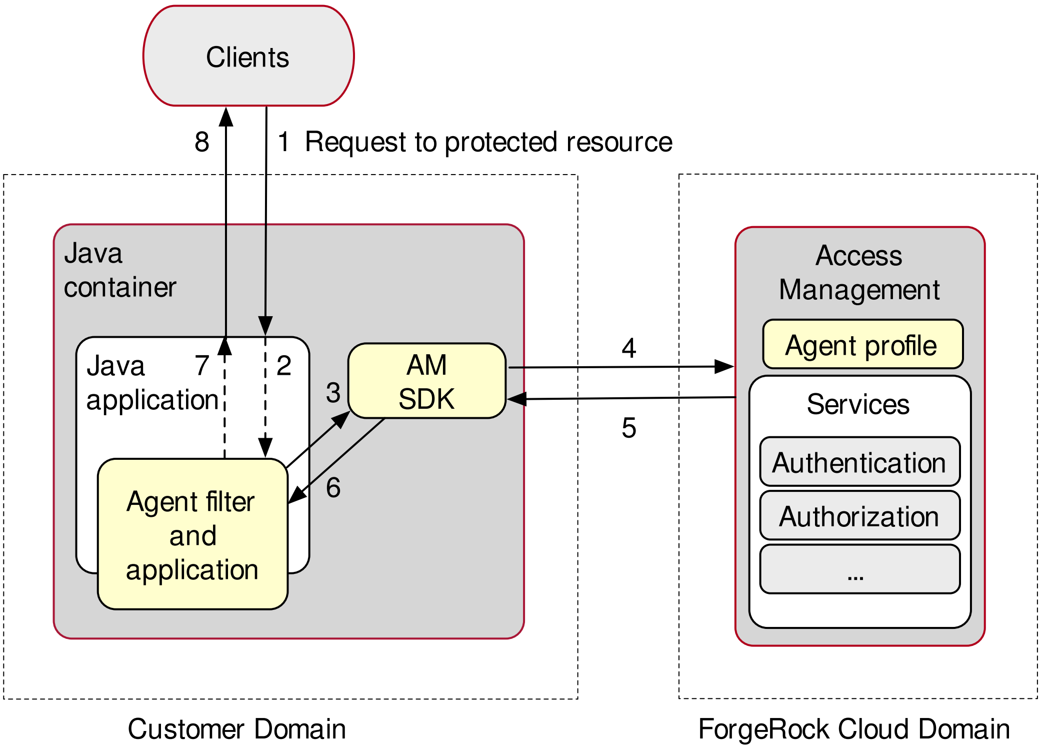
- The agent URL. This guide uses Java Agent installed on `http://agent.example.com:80/app`.
- The root URL of your Identity Cloud. This guide uses `https://tenant.forgeblocks.com:443`.
- The server URL of the Access Management component of the Identity Cloud. This guide uses `https://tenant.forgeblocks.com:443/am`.
- The realm where you work. This guide uses `alpha`.

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

About Java Agents and the ForgeRock Identity Cloud

ForgeRock Identity Cloud simplifies the consumption of ForgeRock as 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 Agents with the ForgeRock 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 ForgeRock Identity Cloud to enforce resource-based policies.



For information, see the Identity Cloud docs.

Enforce policies decisions from ForgeRock Identity Cloud

This example sets up ForgeRock Identity Cloud as a policy decision point for requests processed by Java Agents. For more information about Java Agents, see the User guide.

Before you start, use the Installation guide to install a Java Agent with the following values:

- AM server URL: `https://tenant.forgeblocks.com:443/am`
- Agent URL: `http://agent.example.com:80/app`
- Agent profile name: `java-agent`
- Agent profile realm: `/alpha`
- Agent profile password: `/secure-directory/pwd.txt`

1. Using the ForgeRock Identity Cloud Docs, log in to Identity Cloud as an administrator.
2. Make sure that you are managing the `alpha` realm. If not, switch realms.
3. Add a Java Agent profile in one of the following ways:
   - In the Identity Cloud admin UI:

     ▼ *Details*

     a. In the 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** : `http://agent.example.com:80/app`

     b. Click **Done**
   - In the AM admin UI:

     ▼ *Details*

     a. In the Identity Cloud admin UI, go to **Native Consoles** > **Access Management**. The AM admin UI is displayed.

     b. Select **Applications** > **Agents** > **Java**, and add an agent profile by using the following hints:

        Agent ID

        The ID of the agent profile. This ID resembles a username in AM and is used during the agent installation. For example, `MyAgent` .

        > TIP
        >
        > When AM is not available, the related error message contains the agent profile name. Consider this in your choice of agent profile name.

        Agent URL

        The URL where the agent resides. For more information, refer to Example installation for this guide.

        When the agent is in remote configuration mode, the Agent URL is used to populate the agent profile for services, such as notifications.

        Server URL

The full URL to an authorization server, such as Identity Cloud or AM. For more information, refer to Example installation for this guide.

If the authorization server is deployed in a site configuration (behind a load balancer), enter the site URL. When the agent is in remote configuration mode, the Server URL is used to populate the agent profile for login, logout, naming, and cross-domain SSO.

Password

The password the agent uses to authenticate to an authorization server, such as Identity Cloud or AM. Use this password when installing an agent.

> TIP
>
> Although the agent accepts any password length and content, you are strongly encouraged to generate secure passwords. This can be achieved in various ways, for example using a password manager or by using the command line tool agentadmin --key.

c. On the Java Agent page, select **AM Services** > **OAuth Login URL List**, and enter the redirect URL for unauthenticated requests. The examples in this document use `|?realm=/alpha`. For more information, refer to OAuth Login URL List.

d. Select **AM Services** > **Policy Evaluation Realm Map**, and enter the name of an AM realm to use for policy evaluations. The examples in this document use `/alpha`. For more information, refer to Policy Evaluation Realm Map.

e. Select **AM Services** > **Policy Set Map**, and enter the name of an AM policy set to use for policy evaluations. The examples in this document use `PEP`. For more information, refer to Policy Set Map.

f. Click **Save Changes**.

4. Add a policy set and policy:

   a. In the Identity Cloud admin UI, select 🛡 **Gateways & Agents** and refresh the page.

   b. Select the agent you just created.

   c. On the agent profile page, make sure that **Use Policy Authorization** is selected.

   d. Go to **Policy Set** > **Add**. The AM admin UI is displayed.

   e. Add a policy set with the following values:

        - **Id** : `PEP`
        - **Resource Types** : `URL`

    f. In the policy set, add a policy with the following values:

        - **Name** : `PEP-policy`
        - **Resource Type** : `URL`
        - **Resource pattern** : `*://*:*/*`
        - **Resource value** : `*://*:*/*`

    g. On the **Actions** tab, add actions to allow HTTP `GET` and `POST` .

    h. On the **Subjects** tab, remove any default subject conditions, add a subject condition for all `Authenticated Users` .

5. Assign the new policy set to the agent profile:

    a. Return to the agent profile page on the Identity Cloud admin UI, and refresh the page.

    b. In **Policy Set**, select `PEP` to assign the PEP policy set to the agent profile, and then click **Save**.

6. Test the setup:

    a. In the Identity Cloud admin UI, select **Identities** > **Manage** > **Alpha realm - Users**, and add a new user with the following values:

        - **Username** : `demo`
        - **First name** : `demo`
        - **Last name** : `user`
        - **Email Address** : `demo@example.com`
        - **Password** : `Ch4ng3!t`

    b. Log out of Identity Cloud, and clear any cookies.

    c. Go to `http://agent.example.com:80/app` . The Identity Cloud login page is displayed.

    d. Log in to Identity Cloud as user `demo` , password `Ch4ng3!t` , to access the web page protected by the Java Agent.