Installing a sample instance of Keycloak

This section describes how to install and start a Keycloak server in standalone mode, set up the initial admin user, and log in to the Keycloak Admin Console.

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/standalone/proc-installing-server-community.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/standalone/proc-installing-server-community.adoc)

Installing the Server

You can install the server on Linux or Windows. The server download ZIP file contains the scripts and binaries to run the Keycloak server.

Procedure

1. Download **keycloak-15.0.2.[zip|tar.gz]** from [Keycloak downloads](https://www.keycloak.org/downloads.html).
2. Place the file in a directory you choose.
3. Unpack the ZIP file using the appropriate unzip utility, such as unzip, tar, or Expand-Archive.

Linux/Unix

$ unzip keycloak-15.0.2.zip

or

$ tar -xvzf keycloak-15.0.2.tar.gz

Windows

> Expand-Archive -Path 'C:Downloads\keycloak-15.0.2.zip' -DestinationPath 'C:\Downloads'

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/standalone/proc-starting-server.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/standalone/proc-starting-server.adoc)

Starting the Keycloak server

You start the server on the system where you installed it.

Prerequisites

* You saw no errors during the Keycloak server installation.

Procedure

1. Go to the bin directory of the server distribution.
2. Run the standalone boot script.

Linux/Unix

$ cd bin

$ ./standalone.sh

Windows

> ...\bin\standalone.bat

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/standalone/proc-creating-admin.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/standalone/proc-creating-admin.adoc)

Creating the admin account

Before you can use Keycloak, you need to create an admin account which you use to log in to the Keycloak admin console.

Prerequisites

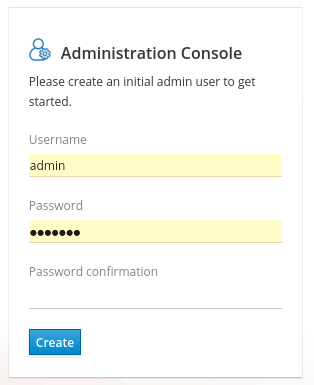
* You saw no errors when you started the Keycloak server.

Procedure

1. Open <http://localhost:8080/auth> in your web browser.

The welcome page opens, confirming that the server is running.

Welcome page



1. Enter a username and password to create an initial admin user.

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/standalone/proc-logging-in-admin-console.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/standalone/proc-logging-in-admin-console.adoc)

Logging into the admin console

After you create the initial admin account, you can log in to the admin console. In this console, you add users and register applications to be secured by Keycloak.

Prerequisites

* You have an admin account for the admin console.

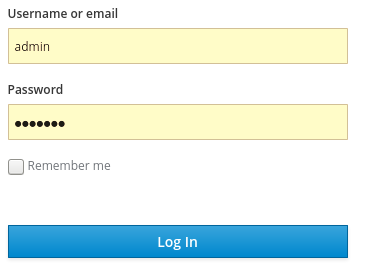
Procedure

1. Click the **Administration Console** link on the **Welcome** page or go directly to <http://localhost:8080/auth/admin/> (the console URL).

|  |  |
| --- | --- |
|  | The Administration Console is generally referred to as the admin console for short in Keycloak documentation. |

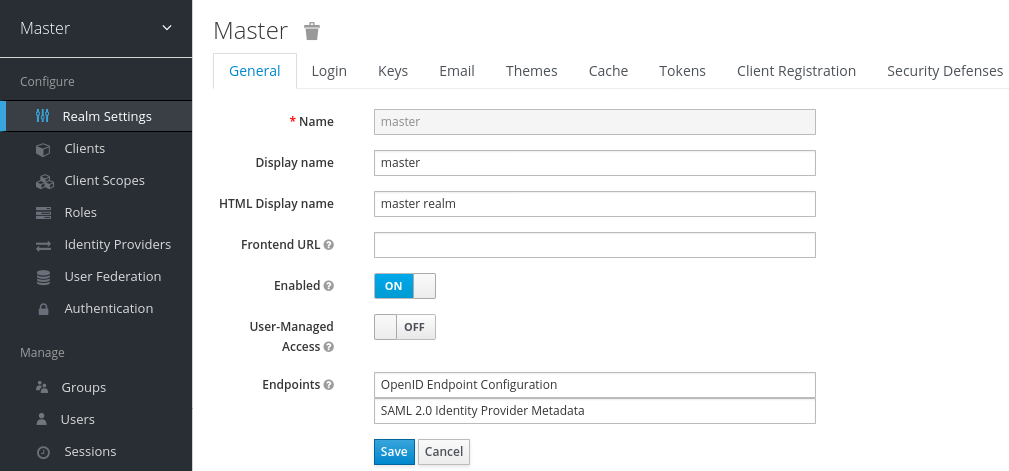
1. Enter the username and password you created on the **Welcome** page to open the **admin console**.

Admin console login screen



The initial screen for the admin console appears.

Admin console



Next steps

Now that you can log into the admin console, you can begin creating realms where administrators can create users and give them access to applications. For more details, see [Creating a realm and a user](https://www.keycloak.org/docs/latest/getting_started/index.html#creating-first-realm_).

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/assembly-creating-first-realm.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/assembly-creating-first-realm.adoc)

Creating a realm and a user

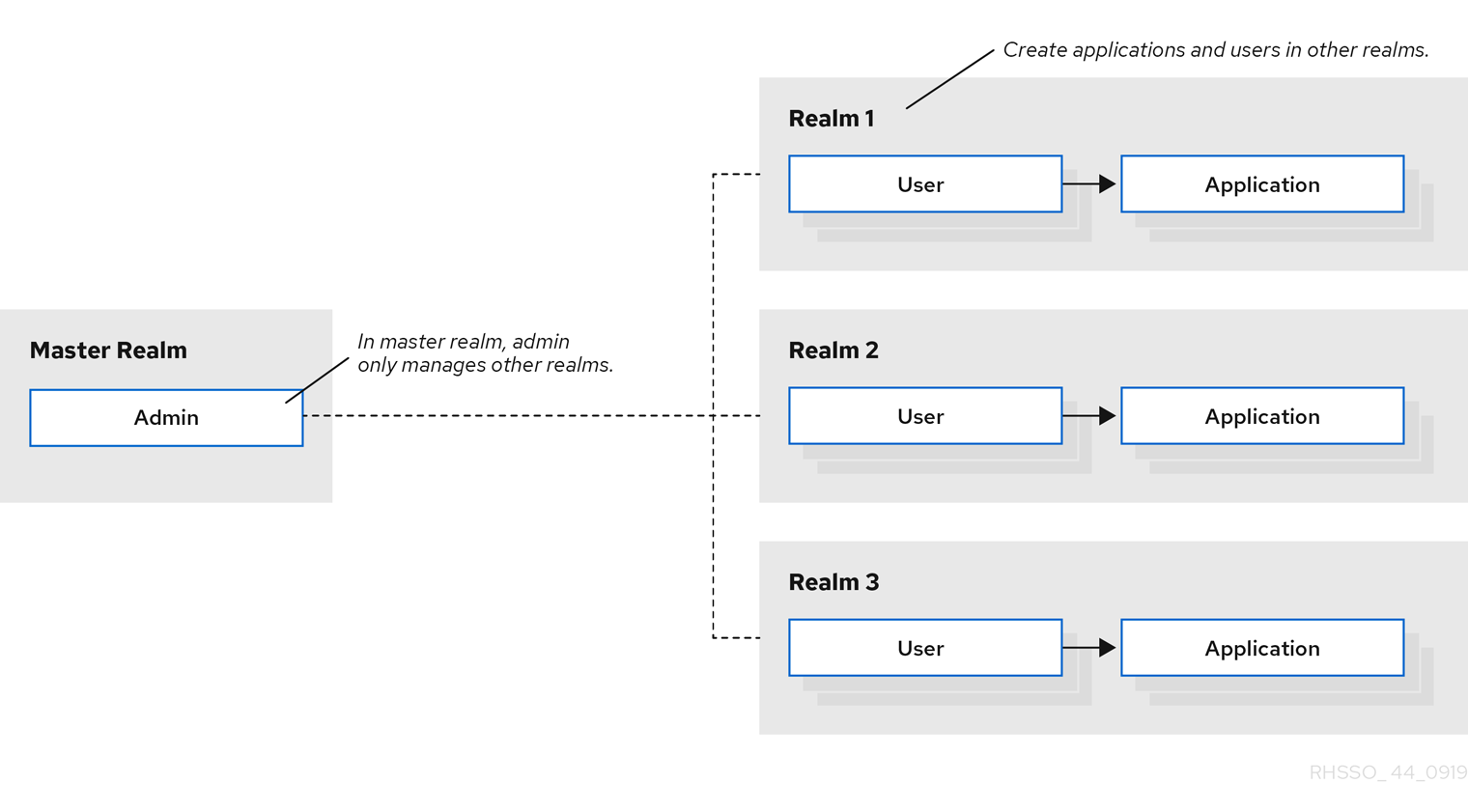
The first use of the Keycloak admin console is to create a realm and create a user in that realm. You use that user to log in to your new realm and visit the built-in account console, to which all users have access.

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/first-realm/con-realms-apps.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/first-realm/con-realms-apps.adoc)

Realms and users

When you log in to the admin console, you work in a realm, which is a space where you manage objects. Two types of realms exist:

* Master realm - This realm was created for you when you first started Keycloak. It contains the admin account you created at the first login. You use this realm only to create other realms.
* Other realms - These realms are created by the admin in the master realm. In these realms, administrators create users and applications. The applications are owned by the users.



[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/first-realm/proc-create-realm.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/first-realm/proc-create-realm.adoc)

Creating a realm

As the admin in the master realm, you create the realms where administrators create users and applications.

Prerequisites

* Keycloak is installed.
* You have the initial admin account for the admin console.

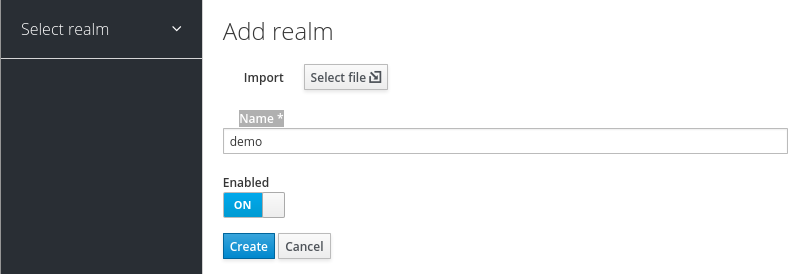
Procedure

1. Go to <http://localhost:8080/auth/admin/> and log in to the Keycloak admin console using the admin account.
2. From the **Master** menu, click **Add Realm**. When you are logged in to the master realm, this menu lists all other realms.
3. Type demo in the **Name** field.

A new realm

A screenshot of a computer

Description automatically generated

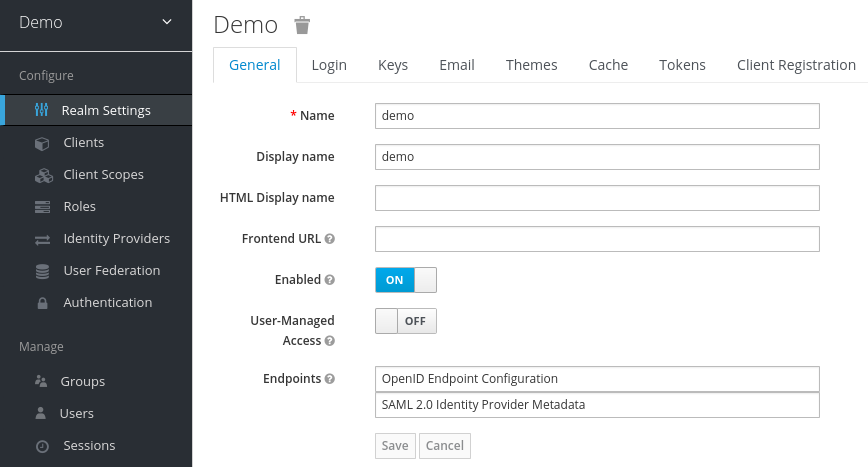


|  |  |
| --- | --- |
|  | The realm name is case-sensitive, so make note of the case that you use. |

1. Click **Create**.

The main admin console page opens with realm set to demo.

Demo realm



1. Switch between managing the master realm and the realm you just created by clicking entries in the **Select realm** drop-down list.

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/first-realm/proc-create-user.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/first-realm/proc-create-user.adoc)

Creating a user

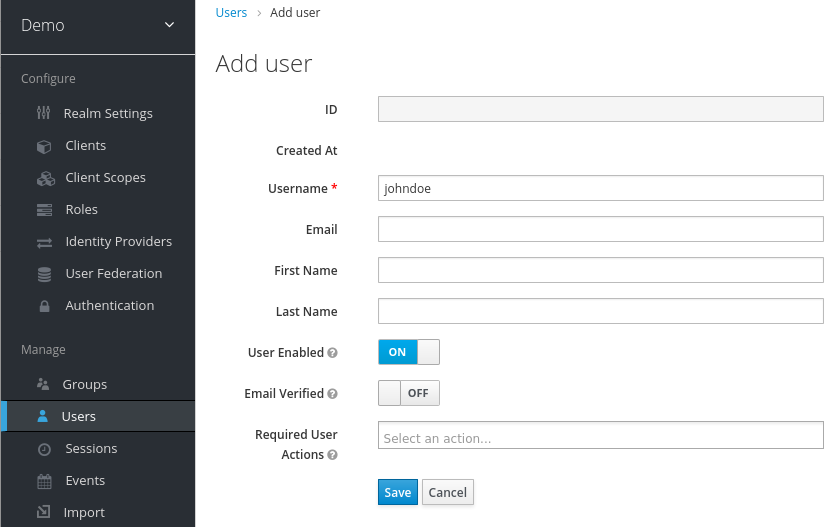
In the demo realm, you create a new user and a temporary password for that new user.

Procedure

1. From the menu, click **Users** to open the user list page.
2. On the right side of the empty user list, click **Add User** to open the Add user page.
3. Enter a name in the Username field.

This is the only required field.

Add user page

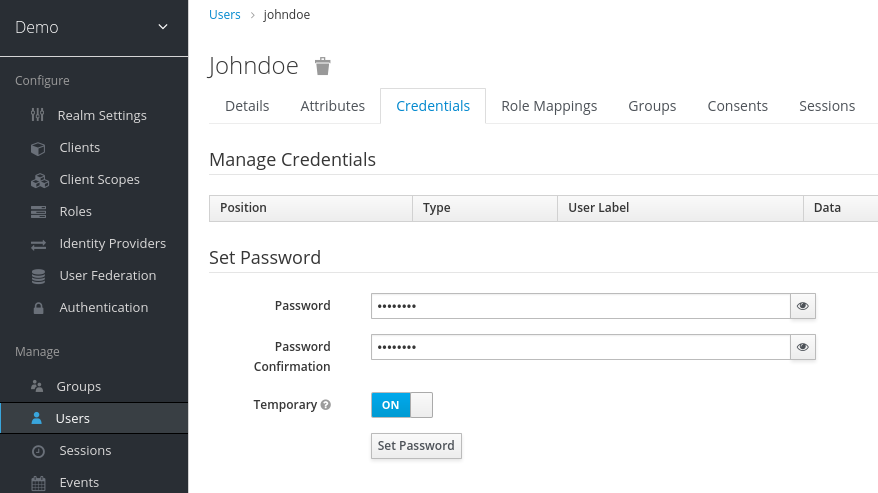


1. Flip the **Email Verified** switch to **On** and click **Save**.

The management page for the new user opens.

1. Click the **Credentials** tab to set a temporary password for the new user.
2. Type a new password and confirm it.
3. Click **Set Password** to set the user password to the new one you specified.

Manage Credentials page



|  |  |
| --- | --- |
|  | This password is temporary and the user will be required to change it at the first login. If you prefer to create a password that is persistent, flip the **Temporary** switch to **Off** and click **Set Password**. |

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/first-realm/proc-view-account.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/first-realm/proc-view-account.adoc)

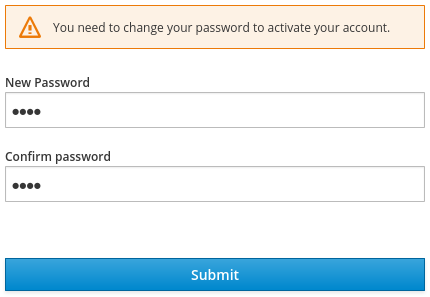
Logging into the Account Console

Every user in a realm has access to the account console. You use this console to update your profile information and change your credentials. You can now test logging in with that user in the realm that you created.

Procedure

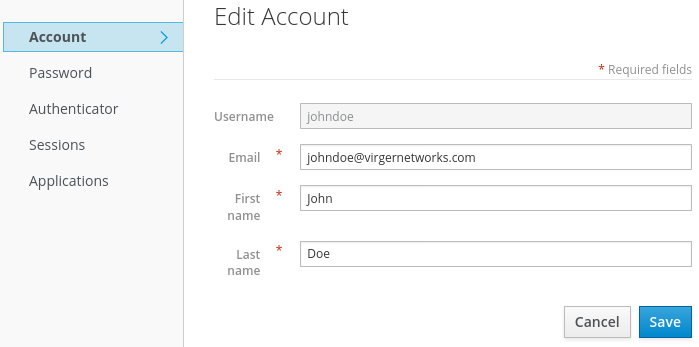
1. Log out of the admin console by opening the user menu and selecting **Sign Out**.
2. Go to <http://localhost:8080/auth/realms/demo/account> and log in to your demo realm as the user that you just created.
3. When you are asked to supply a new password, enter a password that you can remember.

Update password



The account console opens for this user.

Account console



1. Complete the required fields with any values to test using this page.

Next steps

You are now ready for the final procedure, which is to secure a sample application that runs on WildFly. See [Securing a sample application](https://www.keycloak.org/docs/latest/getting_started/index.html#securing-sample-app_).

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/assembly-securing-sample-app.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/assembly-securing-sample-app.adoc)

Securing a sample application

Now that you have an admin account, a realm, and a user, you can use Keycloak to secure a sample WildFly servlet application. You install a WildFly client adapter, register the application in the admin console, modify the WildFly instance to work with Keycloak, and use Keycloak with some sample code to secure the application.

Prerequisites

* You need to adjust the port used by Keycloak to avoid port conflicts with WildFly.

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/sample-app/proc-adjusting-ports.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/sample-app/proc-adjusting-ports.adoc)

Adjusting the port used by Keycloak

The instructions in this guide apply to running WildFly on the same machine as the Keycloak server. In this situation, even though WildFly is bundled with Keycloak, you cannot use WildFly as an application container. You must run a separate WildFly instance for your servlet application.

To avoid port conflicts, you need different ports to run Keycloak and WildFly.

Prerequisites

* You have an admin account for the admin console.
* You created a demo realm.
* You created a user in the demo realm.

Procedure

1. Download WildFly from [WildFly.org](https://www.wildfly.org/).
2. Unzip the downloaded WildFly.

$ unzip <filename>.zip

1. Change to the Keycloak root directory.
2. Start the Keycloak server by supplying a value for the jboss.socket.binding.port-offset system property. This value is added to the base value of every port opened by the Keycloak server. In this example, **100** is the value.

Linux/Unix

$ cd bin

$ ./standalone.sh -Djboss.socket.binding.port-offset=100

Windows

> ...\bin\standalone.bat -Djboss.socket.binding.port-offset=100

Windows Powershell

> ...\bin\standalone.bat -D"jboss.socket.binding.port-offset=100"

1. Confirm that the Keycloak server is running. Go to <http://localhost:8180/auth/admin/> .

If the admin console opens, you are ready to install a client adapter that enables WildFly to work with Keycloak.

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/sample-app/proc-installing-client-adapter.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/sample-app/proc-installing-client-adapter.adoc)

Installing the WildFly client adapter

When WildFly and Keycloak are installed on the same machine, WildFly requires some modification. To make this modification, you install a Keycloak client adapter.

Prerequisites

* WildFly is installed.
* You have a backup of the ../standalone/configuration/standalone.xml file if you have customized this file.

Procedure

1. Download the **WildFly OpenID Connect Client Adapter** distribution from [keycloak.org](https://www.keycloak.org/downloads.html).
2. Change to the root directory of WildFly.
3. Unzip the downloaded client adapter in this directory. For example:

$ unzip <filename>.zip

1. Change to the bin directory.

$ cd bin

1. Run the appropriate script for your platform.

|  |  |
| --- | --- |
|  | If you receive a file not found, make sure that you used unzip in the previous step. This method of extraction installs the files in the right place. |

1. WildFly 10 on Linux/Unix
2. $ ./jboss-cli.sh --file=adapter-install-offline.cli
3. WildFly 10 on Windows
4. > jboss-cli.bat --file=adapter-install-offline.cli
5. Wildfly 11 on Linux/Unix
6. $ ./jboss-cli.sh --file=adapter-elytron-install-offline.cli
7. Wildfly 11 on Windows
8. > jboss-cli.bat --file=adapter-elytron-install-offline.cli

|  |  |
| --- | --- |
|  | This script makes the necessary edits to the …​/standalone/configuration/standalone.xml file. |

1. Start the application server.

Linux/Unix

$ ./standalone.sh

Windows

> ...\standalone.bat

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/sample-app/proc-registering-app.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/sample-app/proc-registering-app.adoc)

Registering the WildFly application

You can now define and register the client in the Keycloak admin console.

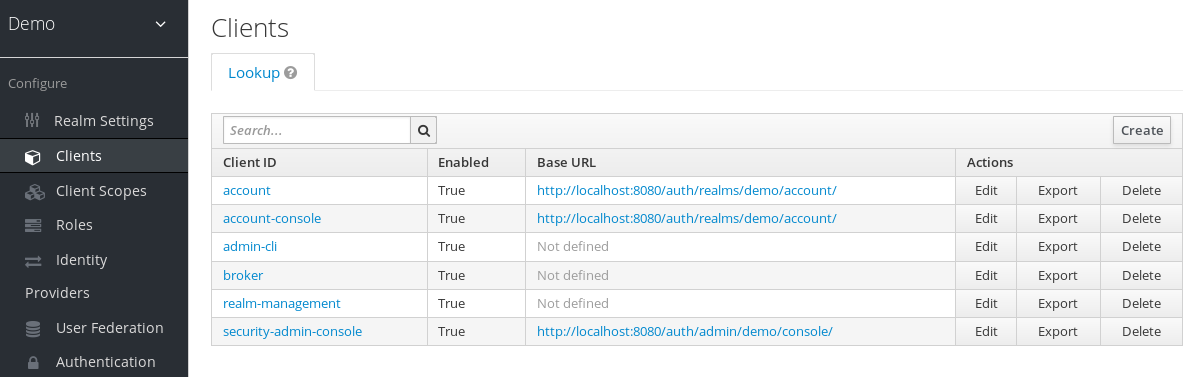
Prerequisites

* You installed a client adapter to work with WildFly.

Procedure

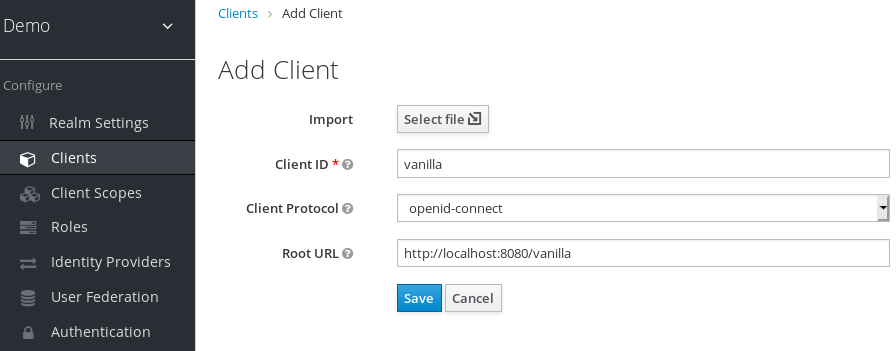
1. Log in to the admin console with your admin account: <http://localhost:8180/auth/admin/>
2. In the top left drop-down list, select the Demo realm.
3. Click Clients in the left side menu to open the Clients page.

Clients



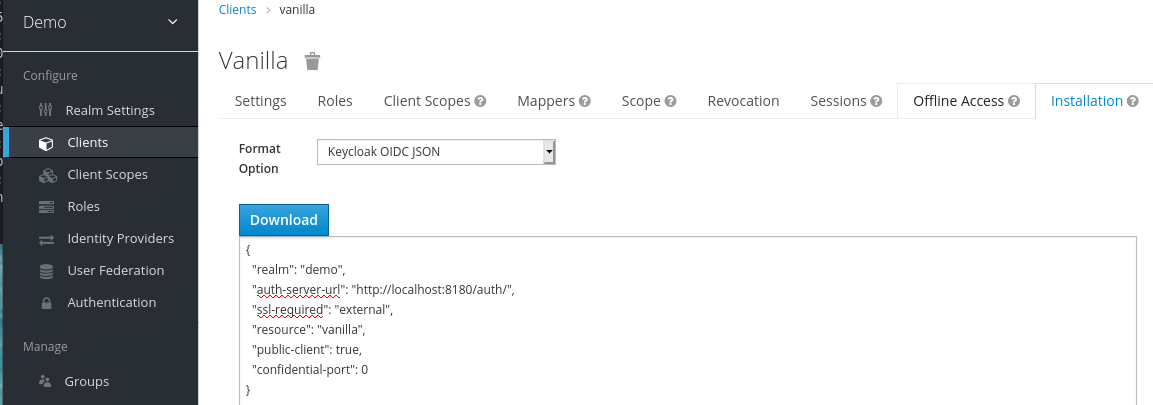
1. On the right side, click **Create**.
2. On the Add Client dialog, create a client called **vanilla** by completing the fields as shown below:

Add Client



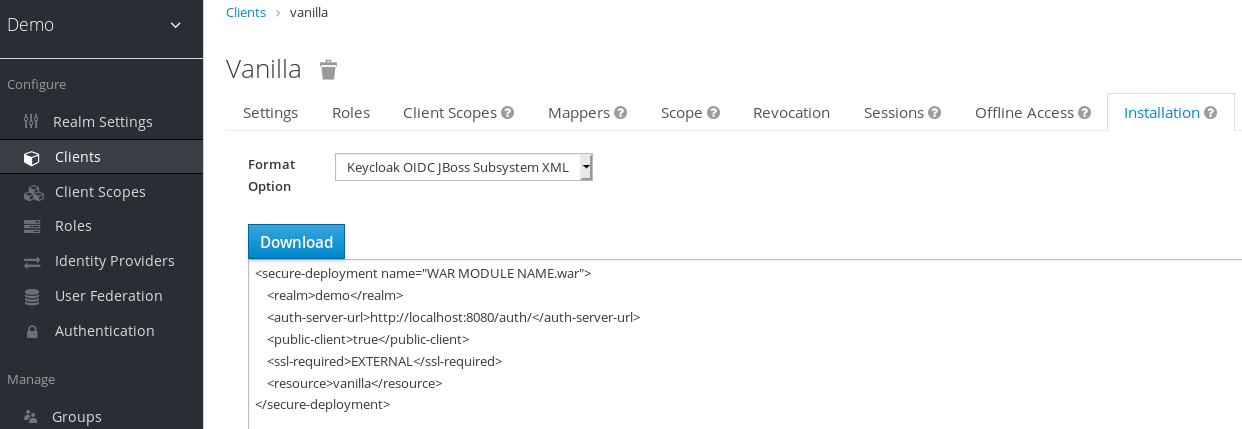
1. Click **Save**.
2. On the **Vanilla** client page that appears, click the **Installation** tab.
3. Select **Keycloak OIDC JSON** to generate a file that you need in a later procedure.

Keycloak.json file



1. Click **Download** to save **Keycloak.json** in a location that you can find later.
2. Select **Keycloak OIDC JBoss Subsystem XML** to generate an XML template.

Template XML



1. Click **Download** to save a copy for use in the next procedure, which involves WildFly configuration.

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/sample-app/proc-modifying-app.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/sample-app/proc-modifying-app.adoc)

Modifying the WildFly instance

The WildFly servlet application requires additional configuration before it is secured by Keycloak.

Prerequisites

* You created a client named **vanilla** in the **demo** realm.
* You saved a template XML file for this client.

Procedure

1. Go to the standalone/configuration directory in your WildFly root directory.
2. Open the standalone.xml file and search for the following text:

<subsystem xmlns="urn:jboss:domain:keycloak:1.1"/>

1. Change the XML entry from self-closing to using a pair of opening and closing tags as shown here:
2. <subsystem xmlns="urn:jboss:domain:keycloak:1.1">

</subsystem>

1. Paste the contents of the XML template within the <subsystem> element, as shown in this example:
2. <subsystem xmlns="urn:jboss:domain:keycloak:1.1">
3. <secure-deployment name="WAR MODULE NAME.war">
4. <realm>demo</realm>
5. <auth-server-url>http://localhost:8180/auth</auth-server-url>
6. <public-client>true</public-client>
7. <ssl-required>EXTERNAL</ssl-required>
8. <resource>vanilla</resource>
9. </secure-deployment>

</subsystem>

1. Change WAR MODULE NAME.war to vanilla.war:
2. <subsystem xmlns="urn:jboss:domain:keycloak:1.1">
3. <secure-deployment name="vanilla.war">
4. ...

</subsystem>

1. Reboot the application server.

[Edit this section](https://github.com/keycloak/keycloak-documentation/blob/master/getting_started/topics/sample-app/proc-installing-sample-code.adoc)[Report an issue](https://issues.redhat.com/secure/CreateIssueDetails!init.jspa?pid=12313920&components=12323375&issuetype=1&priority=3&description=File:%20getting_started/topics/sample-app/proc-installing-sample-code.adoc)

Installing sample code to secure the application

The final procedure is to make this application secure by installing some sample code from the <https://github.com/keycloak/keycloak-quickstarts> repository. The quickstarts work with the most recent Keycloak release.

The sample code is the **app-profile-jee-vanilla** quickstart. It demonstrates how to change a JavaEE application that is secured with basic authentication without changing the WAR. The Keycloak client adapter subsystem changes the authentication method and injects the configuration.

Prerequisites

You have the following installed on your machine and available in your PATH.

* Java JDK 8
* Apache Maven 3.1.1 or higher
* Git

You have a **keycloak.json** file.

Procedure

1. Make sure your WildFly application server is started.
2. Download the code and change directories using the following commands.
3. $ git clone https://github.com/keycloak/keycloak-quickstarts

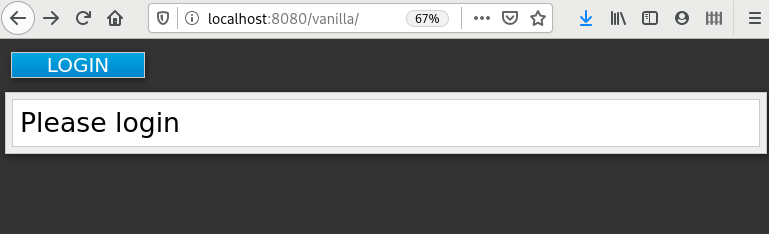
$ cd keycloak-quickstarts/app-profile-jee-vanilla/config

1. Copy the keycloak.json file to the current directory.
2. Move one level up to the app-profile-jee-vanilla directory.
3. Install the code using the following command.

$ mvn clean wildfly:deploy

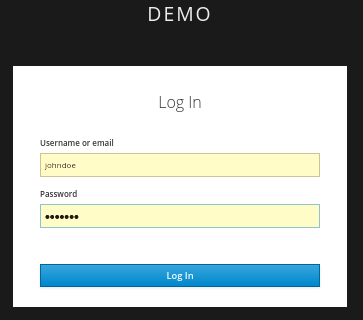
1. Confirm that the application installation succeeded. Go to <http://localhost:8080/vanilla> where a login page is displayed.

Login page confirming success



1. Log in using the account that you created in the demo realm.

Login page to demo realm



A message appears indicating you have completed a successful use of Keycloak to protect a sample WildFly application. Congratulations!

Complete success

