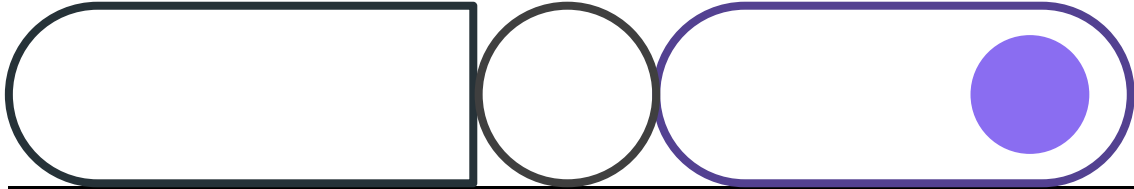


Cientes de reinos de Keycloak

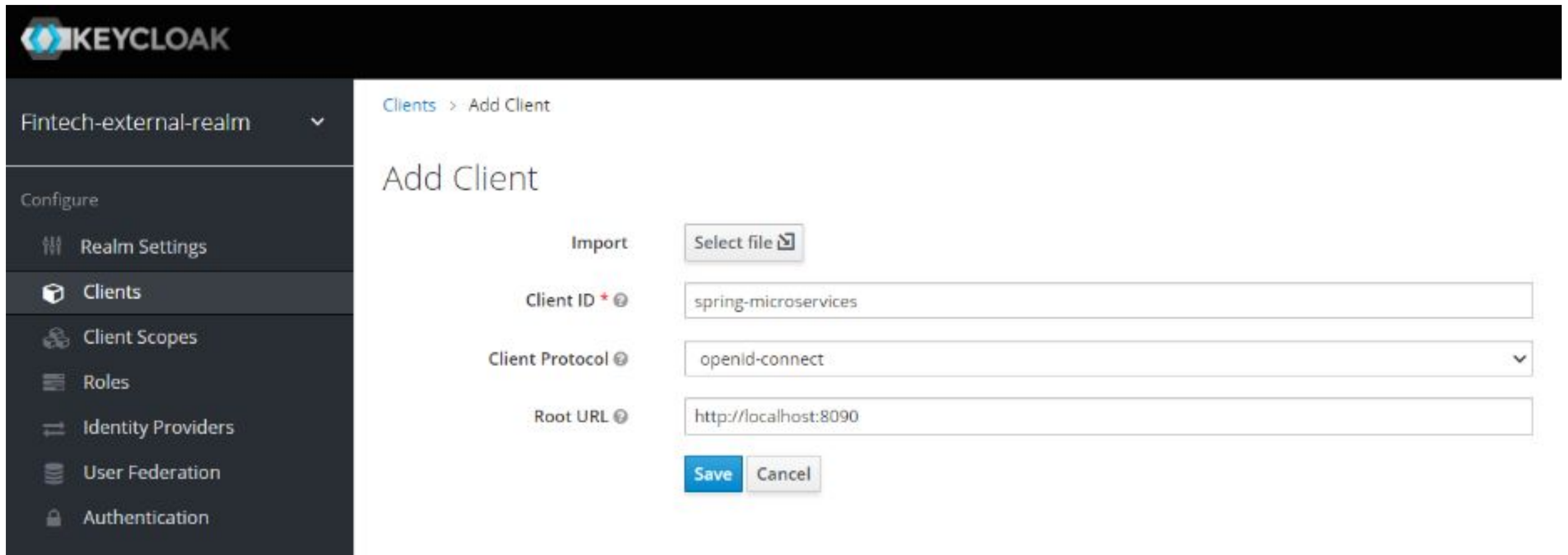


Son las entidades que pueden solicitar a Keycloak la autenticación de un usuario. Generalmente, son aplicaciones y/o servicios que necesitan autenticar vía single sign-on.

También pueden ser entidades que requieren información del usuario autenticado previamente o un token relacionado con ellos para poder invocar a otros servicios de la solución que también usen KeyCloak.



Para crear un cliente dentro de la opción de “**Clients**” seleccionamos el ícono de “**Create**” y dentro de la opción “**Root URL**” especificamos la URL root de nuestra aplicación o servicio.



The screenshot shows the Keycloak administration interface. On the left is a dark sidebar with the 'KEYCLOAK' logo at the top. Below the logo, the selected realm 'Fintech-external-realm' is shown with a dropdown arrow. Under the 'Configure' section, several menu items are listed: 'Realm Settings', 'Clients' (which is highlighted), 'Client Scopes', 'Roles', 'Identity Providers', 'User Federation', and 'Authentication'. The main content area on the right is titled 'Clients > Add Client'. Below this title, there is an 'Add Client' heading. To the left of the form fields is an 'Import' button and a 'Select file' button with a file icon. The form fields include: 'Client ID' with a red asterisk and a help icon, containing the text 'spring-microservices'; 'Client Protocol' with a help icon, set to a dropdown menu showing 'openid-connect'; and 'Root URL' with a help icon, containing the text 'http://localhost:8090'. At the bottom of the form are two buttons: a blue 'Save' button and a grey 'Cancel' button.

Luego debemos hacer que nuestro cliente sea de acceso confidencial para que cada API o servicio que lo requiera deba tener una clave y una contraseña como **secret** para poder acceder a estos servicios de Keycloak.

Con esta opción habilitada tendremos que indicar que la característica **“Authorization Enabled”** esté habilitada.

Fintech-external-realm

Configure

Realm Settings

Clients

Client Scopes

Roles

Identity Providers

User Federation

Authentication

Manage

Groups

Users

Sessions

Events

Import

Export

Clients > spring-microservices

Spring-microservices

SettingsKeysRolesClient ScopesMappersScopeRevocationSessionsOffline AccessInstallation

Client ID

spring-microservices

Name

Description

Enabled

ON

Always Display in Console

OFF

Consent Required

OFF

Login Theme

Client Protocol

openid-connect

Access Type

confidential

Standard Flow Enabled

ON

Implicit Flow Enabled

OFF

Direct Access Grants Enabled

ON

Service Accounts Enabled

ON

OAuth 2.0 Device Authorization Grant Enabled

ON

OIDC CIBA Grant Enabled

OFF

Authorization Enabled

ON

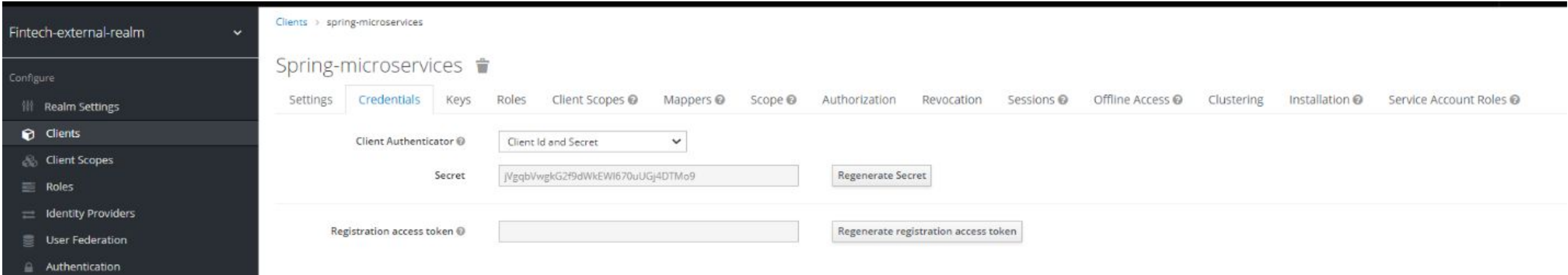
Front Channel Logout

OFF

Root URL

http://localhost:8090

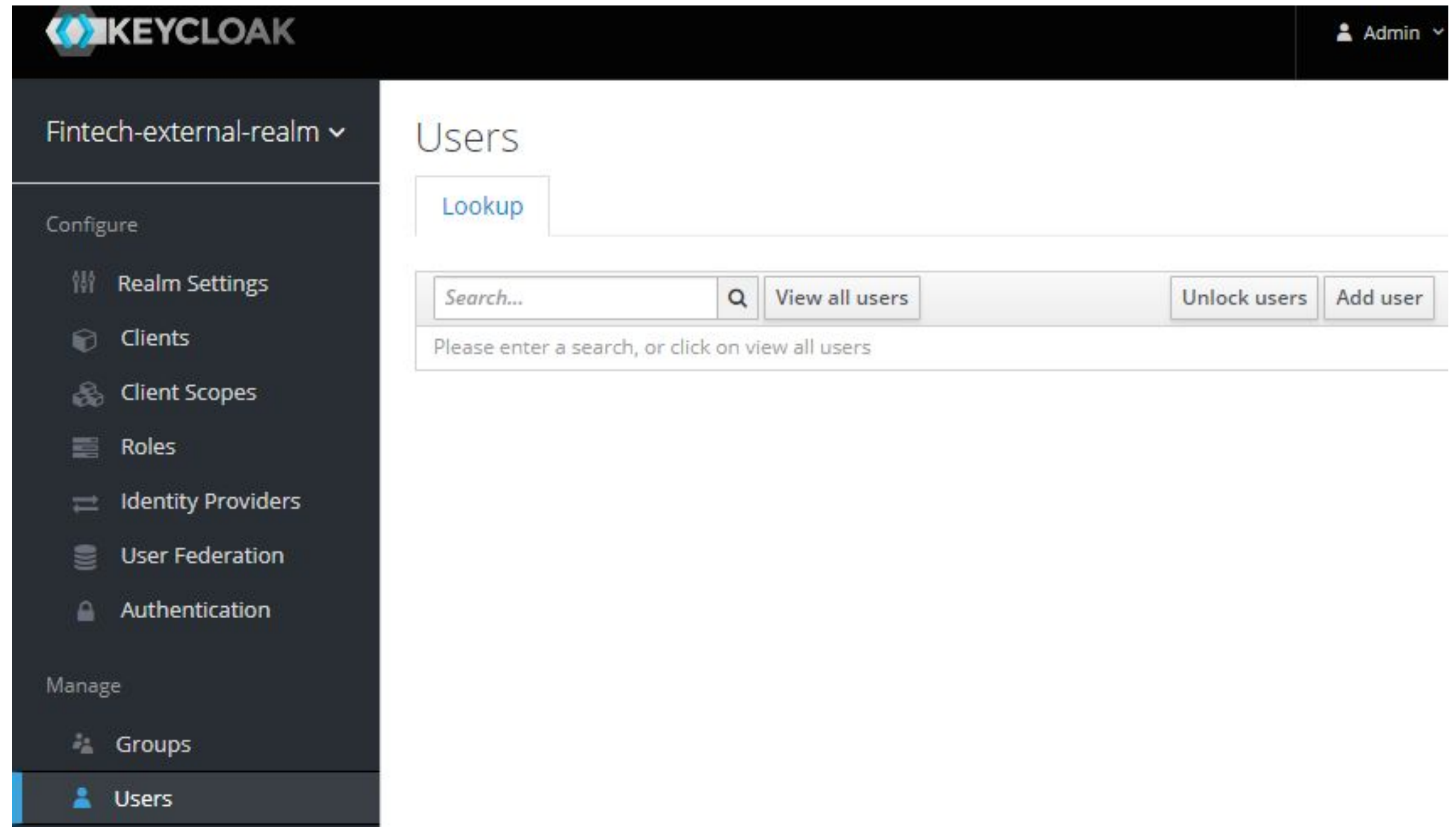
Luego de seleccionar la opción “**Save**”, en la solapa de “**Credentials**”, podremos visualizar el secret a utilizar para conectarnos como clientes de Keycloak desde nuestro microservicio que desea autenticar a un usuario. En este caso, sería: “**jVgqbVwgkG2f9dWkEWI670uUGj4DTMo9**”.



01

Crear un usuario

Para crear un usuario, debemos ir la opción de “**Manage > Users**” y hacer clic en el botón “**Add user**”.



Completamos los datos de usuario y lo dejamos, en esta instancia, sin grupo.

KEYCLOAK

Fintech-external-realm ▾

Configure

Realm Settings

Clients

Client Scopes

Roles

Identity Providers

User Federation

Authentication

Manage

Groups

Users

Sessions

Events

Import

Export

Users > Add user

Add user

ID

Created At

Username *

fintech-user-1

Email

fintech-user-1@dh-fintech.com.ar

First Name

fintech

Last Name

user

User Enabled ?

ON

Email Verified ?

OFF

Select existing group...

No group selected

Groups ?

Required User Actions ?

Save Cancel

Luego de guardarlo, vamos a ir a la opción de “**Credentials**” para asignarle una contraseña.

The screenshot shows the Keycloak administration console. On the left is a dark sidebar with a 'Configure' section containing 'Realm Settings', 'Clients', 'Client Scopes', 'Roles', 'Identity Providers', 'User Federation', and 'Authentication'. Below this is a 'Manage' section with 'Groups', 'Users' (highlighted), 'Sessions', 'Events', 'Import', and 'Export'. The main content area is titled 'Users > fintech-user-1' and shows tabs for 'Details', 'Attributes', 'Credentials' (selected), 'Role Mappings', 'Groups', 'Consents', and 'Sessions'. Under the 'Credentials' tab, there's a 'Manage Credentials' section with a table header: 'Position', 'Type', 'User Label', and 'Data'. Below this is a 'Set Password' section with fields for 'Password' and 'Password Confirmation' (both masked with dots), a 'Temporary' toggle switch set to 'ON', and a 'Set Password' button. At the bottom is a 'Credential Reset' section with a 'Reset Actions' dropdown set to 'Select an action...', an 'Expires In' field set to '12' with a unit dropdown set to 'Hours', and a 'Reset Actions Email' button set to 'Send email'.

Si queremos revisar nuestra configuración en formato JSON, podemos ingresar en la siguiente URL:
<http://localhost:9091/realms/{nombre-realm}/.well-known/openid-configuration>

¡Muchas gracias!