



G CONSELLERIA
O ADMINISTRACIONS
I PÚBLIQUES I
B MODERNITZACIÓ
/ DIRECCIÓ GENERAL
MODERNITZACIÓ I
ADMINISTRACIÓ DIGITAL

Estàndards

Desenvolupament d'aplicacions del GOIB

Entorn de desenvolupament

Desembre de 2019

Índex

Historial de versions.....	3
1. Introducció.....	4
2. OpenJDK 11.....	4
2.1. Instal·lació.....	4
3. JBoss EAP 7.2.....	4
3.1. Instal·lació.....	4
3.2. Configuració de datasources.....	5
3.3. Canvis importants respecte a la versió EAP 5.2.....	7
4. Keycloak 6.0.1.....	8
4.1. Instal·lació en local.....	8
4.2. Exemple de configuració.....	9
5. Connexió JBoss amb Keycloak.....	13
5.1. Exemple de configuració del connector.....	14

Historial de versions

Data	Versió	Descripció	Autor
05/06/19	1.0	Esborrany	Simó Calafat
05/07/19	1.1	Incorporació canvis Fundació Bit	Simó Calafat
09/08/19	1.2	Incorporació configuració Datasource	Simó Calafat
11/12/19	1.3	Revisió de textos, logos, unitats orgàniques i format del document	Asier Cuevas

1. Introducció

La finalitat d'aquest document és descriure el procés de configuració de l'entorn tecnològic per fer servir els estàndards de desenvolupament del *Govern de les Illes Balears (GOIB)*.

Les novetats més rellevants respecte a versions anteriors són:

- OpenJDK 11 com a plataforma de desenvolupament (enlloc de Java SE 7).
- JBoss EAP 7.2 com a servidor d'aplicacions (enlloc de JBoss EAP 5.2).
- Keycloak 6.0.1 com a sistema d'administració d'identitats i accés (enlloc de Seycon).

2. OpenJDK 11

OpenJDK 11 és la versió lliure de la plataforma de desenvolupament Java SE Development Kit 11.

2.1. Instal·lació

1. Accedir a l'adreça <https://jdk.java.net/java-se-ri/11> i escollir entre versió Linux/x64 o Windows/x64 (a aquest manual farem servir la versió Windows).
2. Descarregar el fitxer **openjdk-11+28_windows-x64_bin.zip**.
3. Extreure el fitxer al directori **C:\Program Files\Java**.
4. Establir la variable d'entorn **JAVA_HOME** amb el valor **C:\Program Files\Java\jdk-11** (això és necessari ja que tots els scripts de JBoss fan referència a la variable **JAVA_HOME**).
5. Afegir el valor **%JAVA_HOME%\bin** a la variable d'entorn **PATH**.

3. JBoss EAP 7.2

Red Hat JBoss Enterprise Application Platform 7.2 (JBoss EAP 7.2) és una implementació certificada de les especificacions completes i del perfil web de Java Enterprise Edition 7 (Java EE 7).

3.1. Instal·lació

1. Accedir a l'adreça <https://developers.redhat.com/products/eap/download/>
2. Descarregar el fitxer **jboss-eap-7.2.0-installer.jar** (és necessari registrar-se a la pàgina de RedHat amb un compte gratuït).
3. Executar l'assistent d'instal·lació.
4. Especificar el directori d'instal·lació del JBoss (per exemple: **C:\Desarrollo\jboss-eap-7.2**).
5. Donar d'alta l'usuari administrador del JBoss (per exemple: **admin**). Alternativament, aquest usuari es pot crear amb l'script **JBOSS_HOME\bin\add-user**.
6. Establir la variable d'entorn **JBOSS_HOME** amb el valor del directori d'instal·lació del JBoss (per exemple: **C:\Desarrollo\jboss-eap-7.2**).

3.2. Configuració de datasources

A continuació es descriu el procés de configuració de datasources per sistemes gestors de base de dades Oracle i PostgreSQL segons els estàndards de base de dades de la CAIB.

Oracle

1. Crear el directori **JBOSS_HOME\modules\system\layers\base\com\oracle\main**.
2. Descarregar el fitxer **ojdbc8.jar** al directori anterior des de <https://www.oracle.com/technetwork/database/features/jdbc/jdbc-ucp-122-3110062.html> (s'han d'acceptar els termes i condicions i tenir un compte en el web d'Oracle).
3. Crear el fitxer **module.xml** al directori anterior amb el següent contingut:

```
<module xmlns="urn:jboss:module:1.0" name="com.oracle">
  <resources>
    <resource-root path="ojdbc8.jar"/>
  </resources>
  <dependencies>
    <module name="javax.api"/>
    <module name="javax.transaction.api"/>
  </dependencies>
</module>
```

PostgreSQL:

1. Crear el directori **JBOSS_HOME\modules\system\layers\base\org\postgresql\main**.
2. Descarregar el fitxer **postgresql-42.2.5.jar** al directori anterior des de <https://jdbc.postgresql.org/download.html> (es recomana la descàrrega de la versió 42.2.5 que es troba en la taula de *Other versions* en la columna JDBC 4.2).
3. Crear el fitxer **module.xml** al directori anterior amb el següent contingut:

```
<module xmlns="urn:jboss:module:1.0" name="org.postgresql">
  <resources>
    <resource-root path="postgresql-42.2.5.jar"/>
  </resources>
  <dependencies>
    <module name="javax.api"/>
    <module name="javax.transaction.api"/>
  </dependencies>
</module>
```

Oracle y PostgreSQL:

1. Afegir la següent configuració de «drivers» al fitxer **JBOSS_HOME\standalone\configuration\standalone.xml**.

```
<datasources>
    ...
    <drivers>
        <driver name="h2" module="com.h2database.h2">
            <xa-datasource-class>org.h2.jdbcx.JdbcDataSource</xa-datasource-
class>
        </driver>
        <!-- CAIB drivers -->
        <driver name="oracle" module="com.oracle">
            <xa-datasource-class> oracle.jdbc.xa.client.OracleXADataSource
            </xa-datasource-class>
        </driver>
        <driver name="postgresql" module="org.postgresql">
            <xa-datasource-class>org.postgresql.xa.PGXADatasource
            </xa-datasource-class>
        </driver>
    </drivers>
</datasources>
```

2. Reiniciar el JBoss (si es trobàs en marxa) i afegir els datasources que facin falta. A aquest punt tenim dues opcions:

2.1. Afegint-los directament dins l'etiqueta <datasources> del fitxer **JBoss_HOME\standalone\configuration\standalone.xml**.

Per un datasource de tipus Oracle el contingut seria el següent:

```
<datasource jndi-name="java:jboss/datasources/codiAppDS" pool-
name="codiAppDS" enabled="true" use-java-context="true">
    <connection-url>jdbc:oracle:thin://host:1523/nombd</connection-url>
    <driver>oracle</driver>
    <security>
        <user-name>userapp</user-name>
        <password>pass</password>
    </security>
</datasource>
```

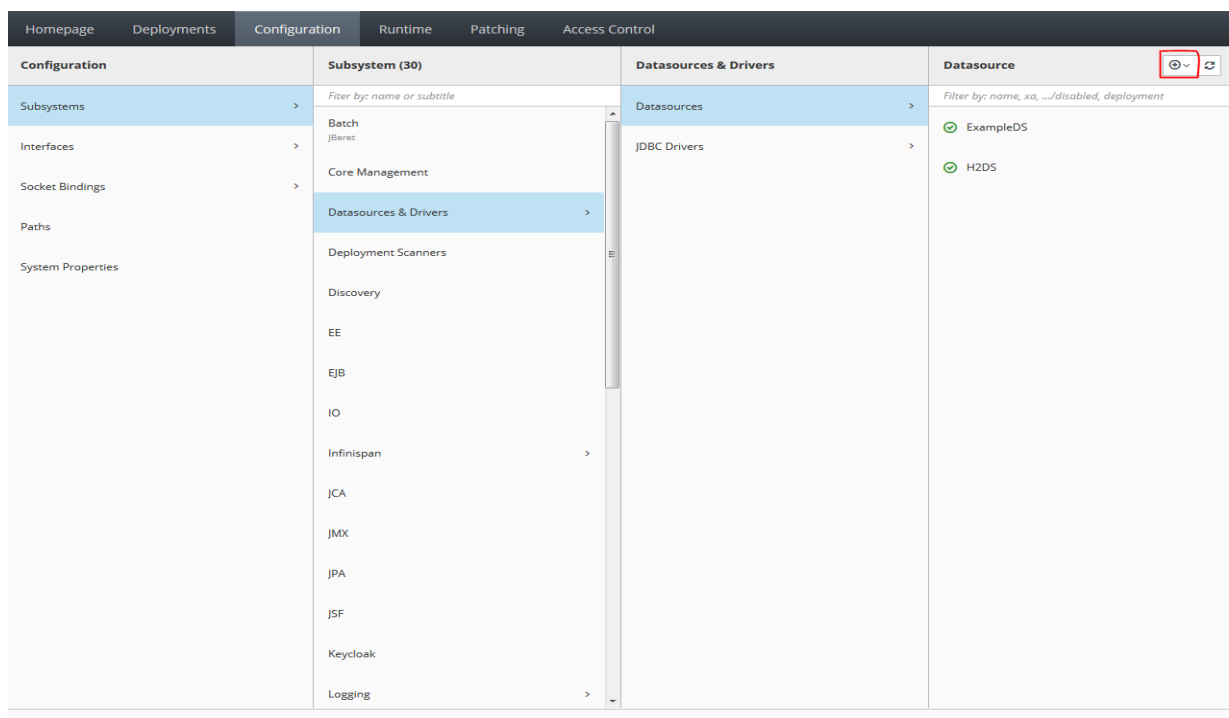
Per un datasource de tipus PostgreSQL el contingut seria el següent:

```
<datasource jndi-name="java:jboss/datasources/codiAppDS" pool-
name="codiAppDS" enabled="true" use-java-context="true">
    <connection-url>jdbc:postgresql://host:5432/nombd</connection-url>
    <driver>postgresql</driver>
    <security>
        <user-name>userapp</user-name>
        <password>pass</password>
    </security>
</datasource>
```

2.2. Fer servir la consola d'administració del JBoss:

1. Accedir a l'adreça de l'**Administration Console** (per defecte: <http://localhost:9990/console/index.html>).
2. Accedir a la pipella «**Configuration**».

3. Accedir a l'apartat «Subsystems/Datasource&Drivers /Datasource».
4. Prémer el botó «+» ressaltat a la imatge següent i seguir l'assistent.



3.3. Canvis importants respecte a la versió EAP 5.2

1. Per iniciar el JBoss s'ha d'executar l'script `JBOSS_HOME\bin\standalone.bat`
2. Per desplegar aplicacions s'ha de copiar el fitxer EAR dins del directori `JBOSS_HOME\standalone\deployments`.
3. El fitxer de configuració principal es troba a `JBOSS_HOME\standalone\configuration\standalone.xml`.
4. Els datasources ja no es configuren a fitxers XML independents sinó que es configuren directament al fitxer `JBOSS_HOME\standalone\configuration\standalone.xml`.

Nota: Si durant l'inici del JBoss aparegués un error al WeldStartService de «Contexto de només lectura», s'ha de afegir el paràmetre `require-bean-descriptor="true"` al subsistema Weld (`<subsystem xmlns="urn:jboss:domain:weld:4.0" require-bean-descriptor="true"/>`) del fitxer `standalone.xml`.

4. Keycloak 6.0.1

Keycloak és un producte de programari de codi obert que permet l'inici de sessió únic (IdP) amb Identity Management i Access Management. A la CAIB farem servir, per un costat, el Keycloak com a servidor esperant peticions d'autenticació, i per altre, el JBoss 7.2 EAP amb un adaptador per poder connectar-lo amb Keycloak.

Nota: A continuació es descriu el procés per instal·lar un servidor Keycloak en local. Aquesta passa no és necessària si es configura el JBoss amb un Keycloak present a l'entorn de desenvolupament de la DGMAD.

4.1. Instal·lació en local

1. Accedir a l'adreça <https://www.keycloak.org/downloads.html>
2. Descarregar el **Standalone Server Distribution versió 6.0.1**.
3. Extreure el fitxer **keycloak-6.0.1.zip** al directori d'instal·lació (per exemple: **C:\Desarrollo\keycloak-6.0.1**).
4. Establir la variable d'entorn **KEYCLOAK_HOME** amb el valor del directori d'instal·lació (per exemple: **C:\Desarrollo\keycloak-6.0.1**).
5. Keycloak és un JBoss modificat. Perquè no hi hagi conflictes de ports entre el JBoss EAP 7.2 i el JBoss del Keycloak, a un dels dos servidors s'ha de substituir el valor de la propietat **port-offset**. Això es degut a que JBoss EAP 7.2 i Keycloak fan servir els mateixos jocs de ports:
 - 8080/8443 per accés HTTP/HTTPS
 - 9990/9993 per configuració HTTP/HTTPS
 - 8009 per AJP

Amb aquest canvi tots els valors dels ports del servidor sumarien 100 al seu valor original:

- 8180/8543 per accés HTTP/HTTPS
- 10090/10093 per configuració HTTP/HTTPS
- 8109 per AJP

El canvi dels ports es pot fer de dues maneres (a aquest manual farem el canvi al Keycloak):

- a) Modificant el paràmetre **port-offset** de la propietat **socket-binding-group** al fitxer **KEYCLOAK_HOME\standalone\configuration\standalone.xml**

```
<socket-binding-group name="standard-sockets" default-interface="public" port-offset="{jboss.socket.binding.port-offset:0}">
```

```
<socket-binding-group name="standard-sockets" default-interface="public" port-offset="{jboss.socket.binding.port-offset:100}">
```

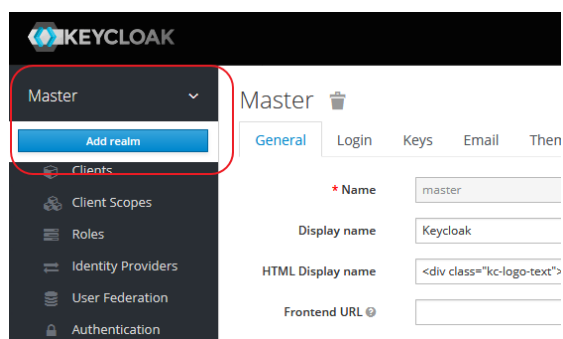
- b) Iniciant el servidor amb la comanda d'execució **KEYCLOAK_HOME\bin\standalone.bat -Djboss.socket.binding.port-offset=100**.
6. Reemplaçar el nom de la variable **JBOSS_HOME** per **KEYCLOAK_HOME** a l'script **KEYCLOAK_HOME\bin\standalone.bat** (a Linux **standalone.sh**).
 7. Iniciar el Keycloak executant l'script **KEYCLOAK_HOME\bin\standalone.bat**.

- Accedir a la consola d'administració del Keycloak (per defecte amb els ports desplaçats: <http://localhost:8180/auth>).
- Si no tenim cap usuari administrador, cal afegir-lo per primera vegada mitjançant la pròpia consola d'administració o a través de l'script `KEYCLOAK_HOME\bin\add-user-keycloak`.

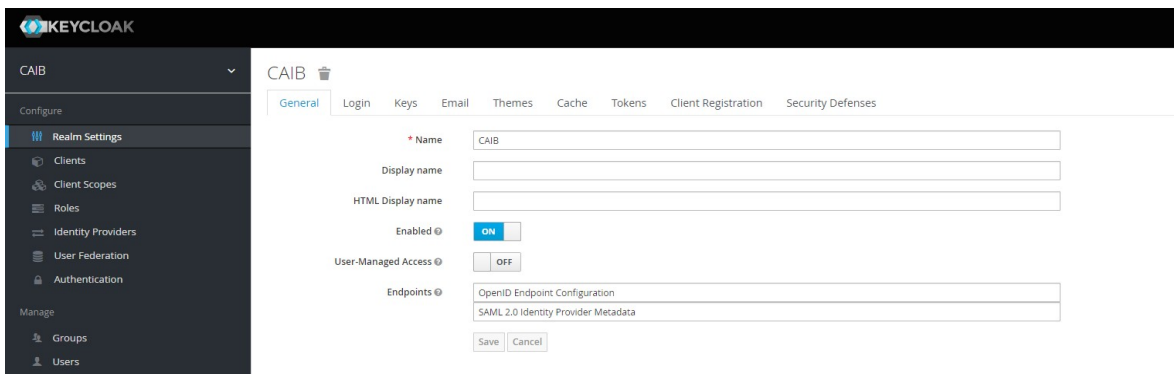
4.2. Exemple de configuració

A continuació es mostra un exemple de configuració de Keycloak per controlar l'accés a l'aplicació de projecte base. A l'exemple crearem un domini d'actuació (realm) i dos clients (un per al backoffice i un per al frontend).

- Accedim a la consola d'administració <http://localhost:8180/auth>.
- Pitjam sobre el desplegable del menú i seleccionam «Add realm»



- Li posam de nom **CAIB** amb la resta de valors per defecte.



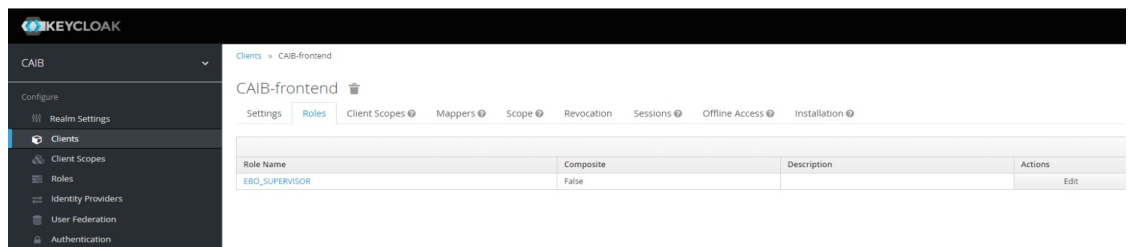
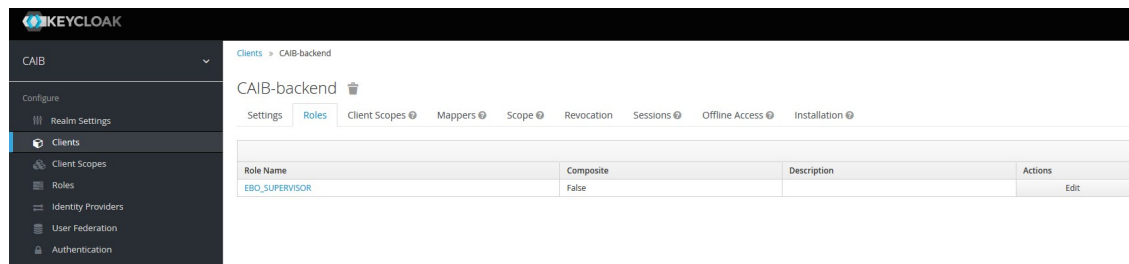
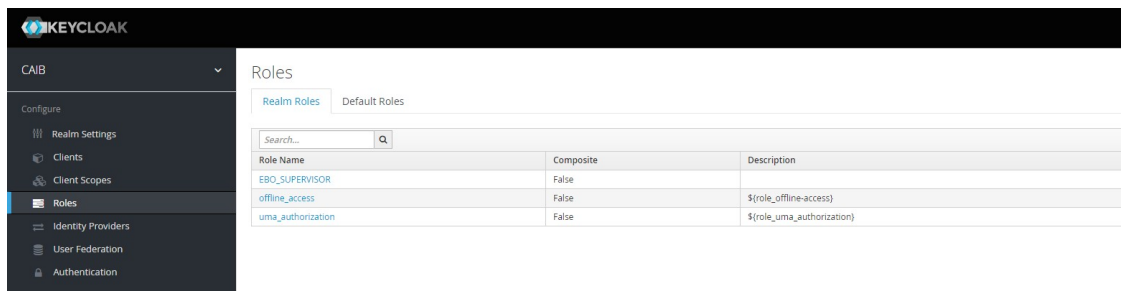
- Afegim els clients **CAIB-backend** i **CAIB-frontend** amb els paràmetres ROOT URL amb el valor `/goibwstest` i `/goibusuari` respectivament i Valid Redirect URLs amb el valor `*` als dos clients.

The screenshot shows the Keycloak Admin Console interface. On the left is a sidebar with a 'CAIB' realm dropdown and a menu with options like 'Clients', 'Client Scopes', 'Roles', 'Identity Providers', 'User Federation', 'Authentication', 'Manage', 'Groups', 'Users', 'Sessions', 'Events', 'Import', and 'Export'. The main area is titled 'CAIB-backend' and has tabs for 'Settings', 'Roles', 'Client Scopes', 'Mappers', 'Scope', 'Revocation', 'Sessions', 'Offline Access', and 'Installation'. The 'Settings' tab is active, displaying various configuration fields for the client. Fields include 'Client ID' (CAIB-backend), 'Name', 'Description', 'Enabled' (ON), 'Consent Required' (OFF), 'Login Theme', 'Client Protocol' (openid-connect), 'Access Type' (public), 'Standard Flow Enabled' (ON), 'Implicit Flow Enabled' (OFF), 'Direct Access Grants Enabled' (ON), 'Authorization Enabled' (OFF), 'Root URL' (/goibwstest), 'Valid Redirect URIs' (with a '+' button), 'Base URL', 'Admin URL', and 'Web Origins' (with '-' and '+' buttons).

The screenshot shows the Keycloak Admin Console interface for the 'CAIB-frontend' client. The layout is identical to the previous screenshot, with the 'Settings' tab active. The configuration fields are: 'Client ID' (CAIB-frontend), 'Name', 'Description', 'Enabled' (ON), 'Consent Required' (OFF), 'Login Theme', 'Client Protocol' (openid-connect), 'Access Type' (public), 'Standard Flow Enabled' (ON), 'Implicit Flow Enabled' (OFF), 'Direct Access Grants Enabled' (ON), 'Authorization Enabled' (OFF), 'Root URL' (/goibusuari), 'Valid Redirect URIs' (with a '+' button), 'Base URL', 'Admin URL', and 'Web Origins' (with '-' and '+' buttons).

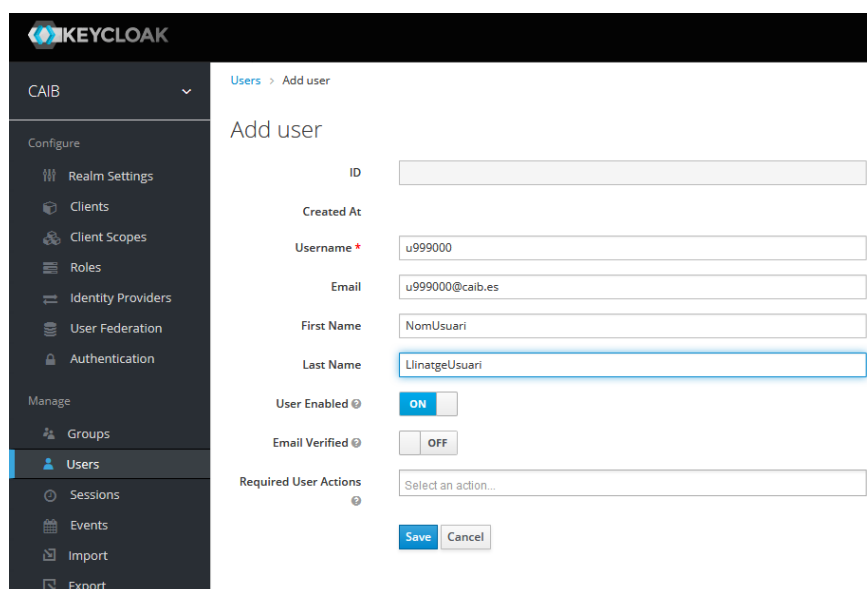
5. Afegim el rol **EBO_SUPERVISOR** dins el realm i dins de cada client. D'aquesta manera, podem configurar rols a nivell de realm (perquè els usuaris tinguin accés a tots els mòduls) o a nivell de client (perquè els usuaris tinguin accés només a un mòdul en particular).

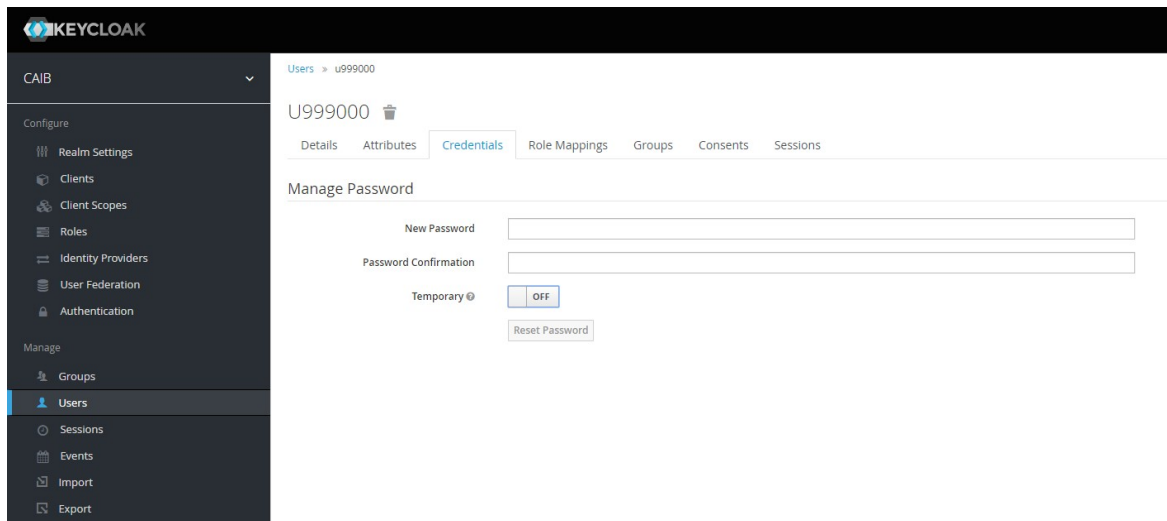
Nota: Al capítol 5 veurem com configurar el connector de JBoss amb Keycloak per establir el nivell d'autenticació fent servir el paràmetre «**use-resource-role-mappings**».



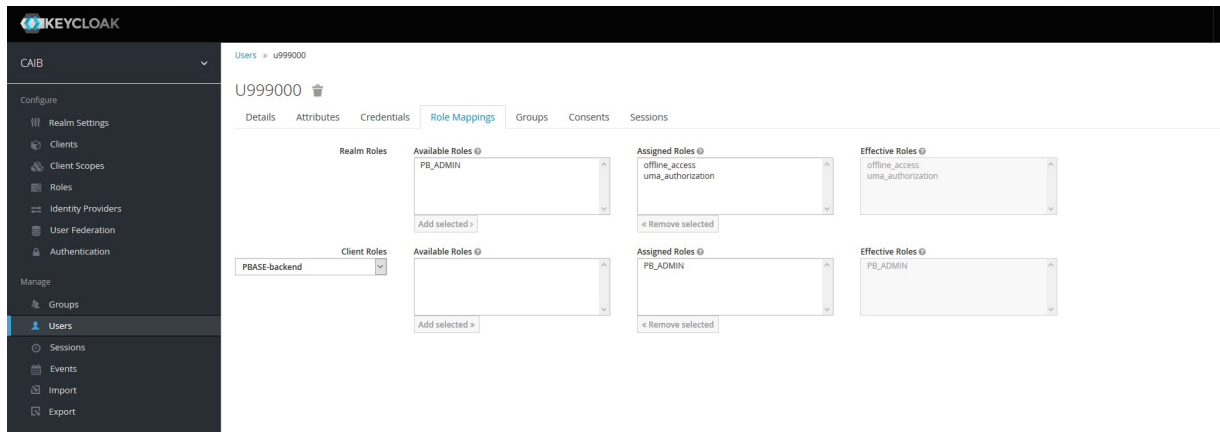
6. Afegim l'usuari **u999000** i li assignam una contrasenya dins l'apartat «**credentials**».

Nota: Si no li assignam contrasenya no es podrà autenticar.





7. Assignam a l'usuari creat el rol **EBO_SUPERVISOR** per al client **CAIB-backend**.



5. Connexió JBoss amb Keycloak

Per connectar JBoss EAP 7.2 amb un servidor Keycloak (ja sigui en local o fent servir un servidor present a l'entorn de desenvolupament de la DGMAD) s'ha d'instal·lar un adaptador.

1. Accedir a l'adreça <https://www.keycloak.org/downloads.html>
2. Descarregar el **Client Adapter** de Keycloak (OPENID CONNECT) per a JBoss 7 EAP.
3. Extreure el fitxer **keycloak-wildfly-adapter-dist-6.0.1.zip** al **JBOSS_HOME**. Al directori **JBOSS_HOME\bin** s'afegiran els següents executables:
 - adapter-install-offline.cli
 - adapter-install.cli
 - adapter-elytron-install-offline.cli
 - adapter-elytron-install.cli

Important: Actualment, la versió amb ELYTRON té UN BUG i dona problemes amb els EJBs. Per tant, es desaconsella fer-lo servir. La diferència entre les versions «normal» i les «offline» és que el seu èxit depèn de si el JBoss està en marxa o no, respectivament.

4. Amb el JBoss aturat, executar la comanda **jboss-cli.bat -file=adapter-install-offline.cli**.

```
C:\DesarrolloSimo\jboss-eap-7.2\bin>jboss-cli.bat --file=adapter-install-offline.cli
OpenJDK 64-Bit Server VM warning: Ignoring option PermSize; support was removed in 8.0
OpenJDK 64-Bit Server VM warning: Ignoring option MaxPermSize; support was removed in 8.0
{"outcome" => "success"}
{"outcome" => "success"}
{"outcome" => "success"}
{"outcome" => "success"}
Presione una tecla para continuar . . .
```

5. Per últim, s'ha d'afegir la següent configuració dins el «subsystem» que fa referència al keycloak al fitxer **JBOSS_HOME\standalone\configuration\standalone.xml**:

```
<subsystem xmlns="urn:jboss:domain:keycloak:1.1">
  <realm name="NOM_REALM">
    <auth-server-url>URL_KEYCLOAK</auth-server-url>
    <ssl-required>NONE/EXTERNAL/ALL</ssl-required>
  </realm>
  <secure-deployment name="NOM_WAR.war">
    <realm>NOM_REALM</realm>
    <resource>NOM_CLIENT</resource>
    <use-resource-role-mappings>TRUE/FALSE</use-resource-role-mappings>
    <public-client>true</public-client>
    <verify-token-audience>true</verify-token-audience>
  </secure-deployment>
  <secure-deployment name="NOM_WAR_2.war">
    <realm>NOM_REALM</realm>
    <resource>NOM_CLIENT_2</resource>
    <use-resource-role-mappings>TRUE/FALSE</use-resource-role-mappings>
    <public-client>true</public-client>
    <verify-token-audience>true</verify-token-audience>
  </secure-deployment>
</subsystem>
```

Els valors a configurar són els següents:

- **realm name:** nom del REALM (domini d'actuació del keycloak).
- **auth-server-url:** Url del servidor keycloak (si es té en local, <http://localhost:8181/auth>; si es vol fer servir el de desenvolupament de la DGMAD, <http://dev.caib.es/auth>).
- **ssl-required:** Els valors possibles són:
 - NONE: No es requereix HTTPS per cap adreça IP de client.
 - EXTERNAL: Les adreces IP privades i de localhost poden accedir sense HTTPS.
 - ALL: Es requereix HTTPS per totes les adreces IP.
- **secure-deployment:** configuració d'un model identificat pel nom de WAR. S'hi ha d'especificar el nom de realm sota el qual fa feina el mòdul.
- **resource:** Nom de CLIENT o mòdul a que es fa referència dins el Keycloak
- **use-resource-role-mappings:**
 - TRUE: avalua el rol a nivell de CLIENT.
 - FALSE: avalua el rol a nivell de REALM.

Nota: Si es vol fer servir el servidor de desenvolupament de la DGMAD aquests valors s'han de consensuar amb Serveis Web de la DGMAD.

5.1. Exemple de configuració del connector.

A continuació es mostra un exemple de configuració per connectar una aplicació denominada **goibusuari** amb el servidor Keycloak local descrit a l'apartat «4.2. Exemple de configuració»; es a dir, farem servir el realm CAIB i els clients CAIB-backend i CAIB-frontend.

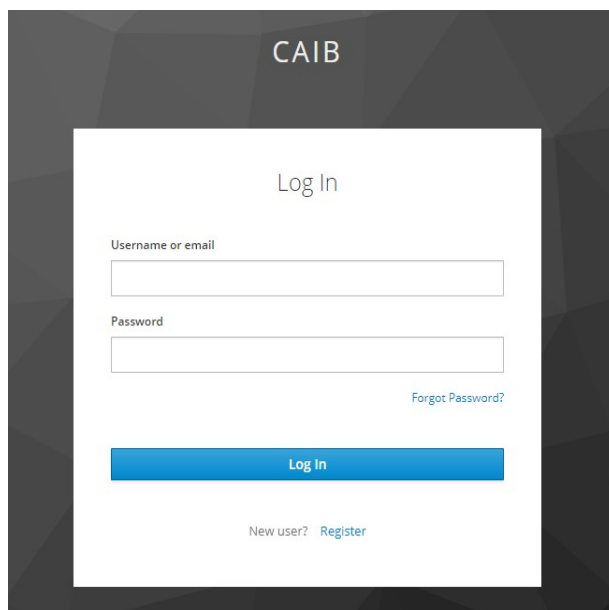
1. Suposant que volem accedir només a nivell de client, la configuració del fitxer **JBOSS_HOME\standalone\configuration\standalone.xml** seria la següent:

```
<subsystem xmlns="urn:jboss:domain:keycloak:1.1">
  <realm name="CAIB">
    <auth-server-url>http://localhost:8180/auth</auth-server-url>
    <ssl-required>EXTERNAL</ssl-required>
  </realm>
  <secure-deployment name="userinfo.war">
    <realm>CAIB</realm>
    <resource>CAIB-backend</resource>
    <use-resource-role-mappings>true</use-resource-role-mappings>
    <public-client>true</public-client>
    <verify-token-audience>true</verify-token-audience>
  </secure-deployment>
  <secure-deployment name="rest.war">
    <realm>CAIB</realm>
    <resource>CAIB-frontend</resource>
    <use-resource-role-mappings>true</use-resource-role-mappings>
    <public-client>true</public-client>
    <verify-token-audience>true</verify-token-audience>
  </secure-deployment>
</subsystem>
```

2. El projecte **goibusuari** ja té configurat el rol **EBO_SUPERVISOR** dins el paquet **userinfo.war**, en concret, dins del fitxer **src\main\webapp\WEB-INF\web.xml**:

```
<security-constraint>
  <web-resource-collection>
    <web-resource-name>UserInfo</web-resource-name>
    <url-pattern>/*</url-pattern>
    <http-method>POST</http-method>
    <http-method>GET</http-method>
  </web-resource-collection>
  <auth-constraint>
    <role-name>EBO_SUPERVISOR</role-name>
  </auth-constraint>
</security-constraint>
<login-config>
  <auth-method>KEYCLOAK</auth-method>
  <realm-name>Autenticacio</realm-name>
</login-config>
<security-role>
  <role-name>EBO_SUPERVISOR</role-name>
</security-role>
```

3. Desplegar el fitxer **goibusuari.ear** dins del directori **JBoss_HOME\standalone\deployments** del JBoss EAP 7.2.
4. Accedir a l'adreça <http://localhost:8080/goibusuari/>
5. Apareixerà una finestra on s'ha d'inserir les credencials per accedir a l'aplicació (a aquest cas, l'usuari **u999000** creat a la secció 4.2).



CAIB

Log In

Username or email

Password

[Forgot Password?](#)

Log In

New user? [Register](#)

6. Després d'inserir les credencials correctament s'obté el resultat esperat.



Dades de l'usuari autenticat:

Atribut	Valor
id	u999000
Nom	NomUsuari
Llinatges	LlinatgeUsuari
Correu	u999000@caib.es
EJB aleatori	83