

PROCEDIMENTO DE INSTALAÇÃO KEYCLOAK

SUMÁRIO

1. Objetivo	4
2. Arquitetura.....	4
2.1. Cenário 1 – Modelo Servidor Único	5
2.2. Cenário 2 – Modelo Segregado.....	6
2.3. Cenário 3 – Modelo Alta Disponibilidade	7
3. Pré-Requisitos	7
4. Procedimento de Instalação.....	8
4.1. Banco de Dados	8
4.2. Servidor do Keycloak	8
4.3. Arquivo standalone.conf.....	8
4.4. Arquivo standalone.xml.....	9
4.4.1. Configuração da Rotação de Log	9
4.4.2. Configuração do Datasource	10
4.5. Serviço de Inicialização	12
4.6. Criação do Usuário Administrador	13
5. Administração Keycloak	13
5.1. Alteração de senha do usuário administrador.....	13
5.2. Criação da Federação	14
5.3. Criação dos Clients.....	15
5.4. Criação de Usuários	17
6. Configuração de Alta Disponibilidade.....	18
6.1. Arquivo standalone.conf.....	18
6.2. Arquivo standalone-ha.xml	19
6.2.1. Configuração do Infinispan e JGroups.....	19
6.3. Banco de Dados	19
6.4. Configurações de Rede.....	20
6.5. Iniciando o cluster	20

HISTÓRICO DE REVISÃO

Data	Versão	Autor	Descrição
07/04/2020	1.0	Time Arquitetura	Criação do documento
21/06/2023	2.0	Ivan Pereira	Alteração de layout Atualização dos procedimentos de instalação Adição da configuração do modo de alta disponibilidade
06/12/2023	2.1	Ivan Pereira	Adição das configurações de usuários
14/03/2024	2.2	Ivan Pereira	Revisão do documento

1. Objetivo

Este documento tem como objetivo descrever o processo de configuração do servidor de autenticação Keycloak, permitindo técnicos e administradores de sistemas executarem os procedimentos de implantação dos sistemas DIMENSA. É requerido que os executores estejam habituados com rotinas de instalação e configuração de Sistemas Operacionais, Banco de Dados e Servidores de Aplicação, bem como as tarefas de implantação de aplicações.

Será necessário um DBA ou um profissional técnico com conhecimento e habilidade para executar as tarefas relacionadas ao Banco de Dados.

2. Arquitetura

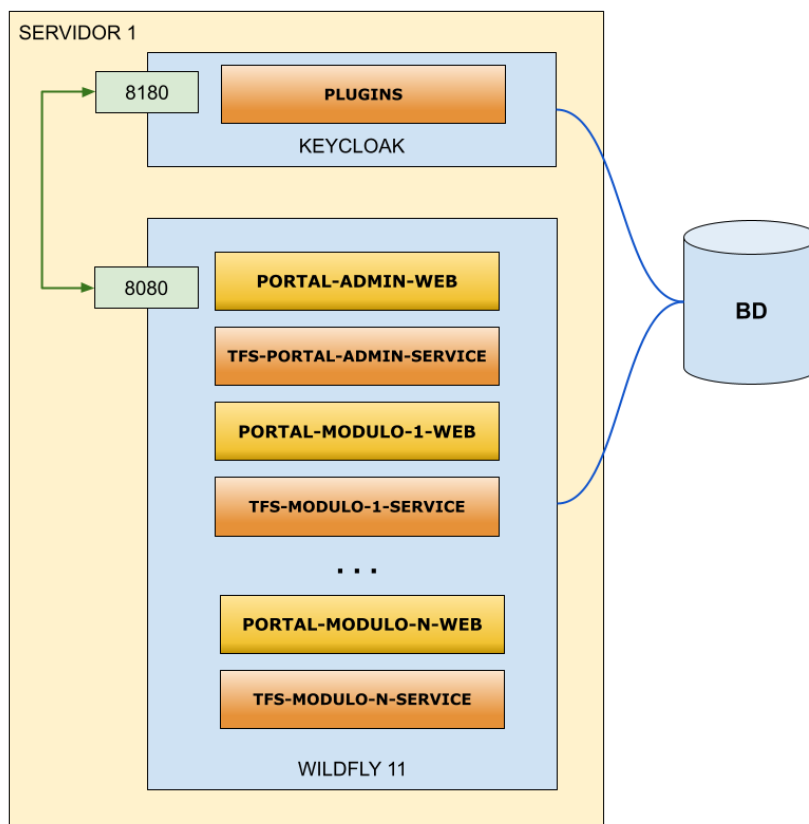
O Keycloak é bastante versátil em relação a execução, por isto sugerimos três cenários para implantação dos servidores de aplicação e autenticação no ambiente. Entender e definir esta arquitetura inicial influencia nas configurações a serem realizadas nos passos posteriores deste documento.

É importante ressaltar que as configurações definidas nessa arquitetura são mínimas e necessárias para o funcionamento do ambiente em implantação.

2.1. Cenário 1 – Modelo Servidor Único

Neste modelo de arquitetura, o servidor de aplicação e autenticação são executados no mesmo servidor. Recomendado para ambientes onde são executados poucos módulos, com baixa utilização e poucos usuários simultâneos. Os recursos do servidor deverão ser dimensionados de acordo com o número de módulos implantados.

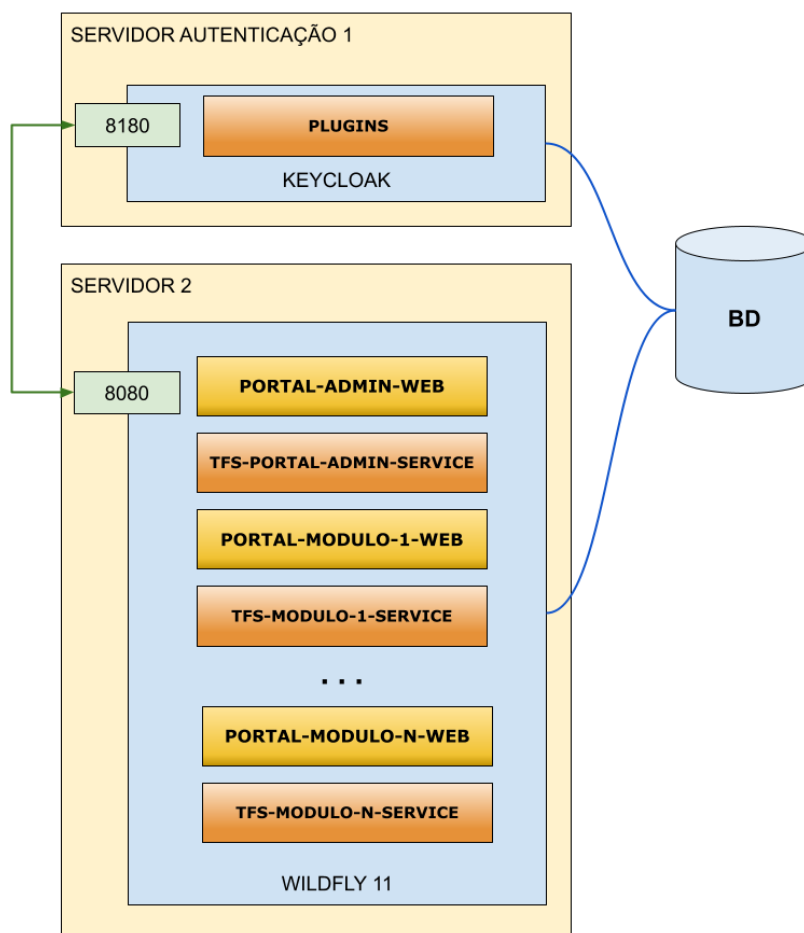
O diagrama do modelo de Servidor único está ilustrado abaixo:



2.2. Cenário 2 – Modelo Segregado

Neste modelo de arquitetura, os servidores de aplicação ficam separados do servidor de autenticação, podendo ter 1 ou mais servidores de aplicação utilizando o mesmo servidor de autenticação. Recomendado para ambientes onde o número de módulos e usuários são um pouco maiores, mas que não necessitem de distribuição de carga nem alta disponibilidade.

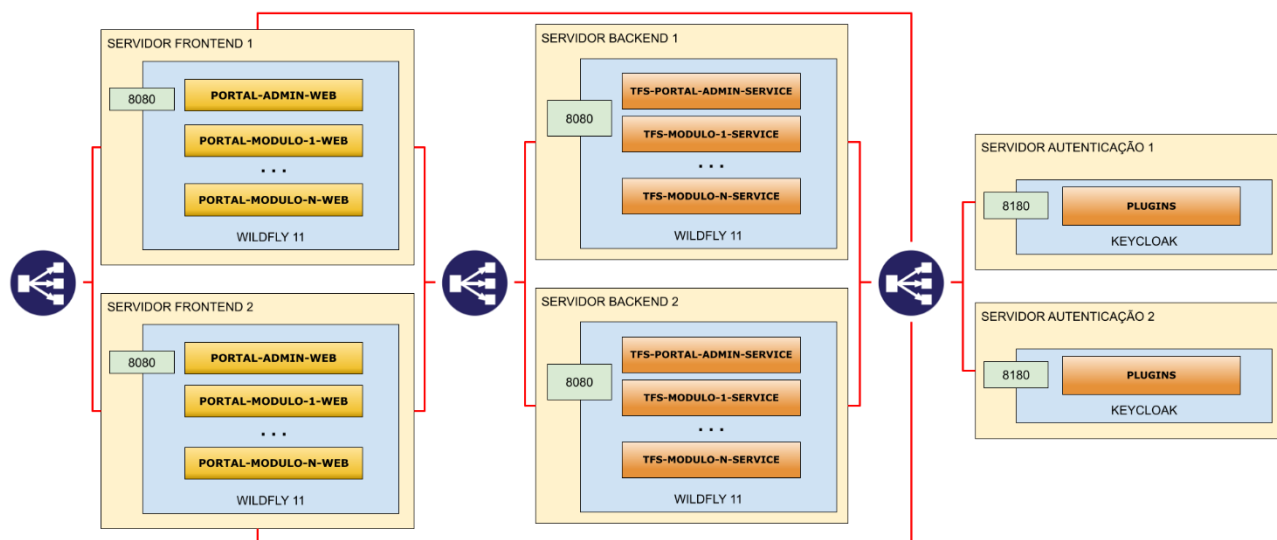
Este modelo segregado está ilustrado no diagrama abaixo:



2.3. Cenário 3 – Modelo Alta Disponibilidade

Neste modelo de arquitetura, os servidores de aplicação e os servidores de autenticação operam no modo cluster tendo suas conexões distribuídas por um balanceador de carga, podendo utilizar dois ou mais servidores de autenticação. Esta arquitetura é recomendada para ambientes onde exige-se um número maior de acessos aos serviços de autenticação.

Uma sugestão do modelo de alta disponibilidade está ilustrada no diagrama abaixo:



3. Pré-Requisitos

Sistema Operacional

Este produto é compatível com a maioria das versões do Windows e distribuições do Linux, contudo pode haver pequenas diferenças de comportamento nos sistemas que ainda não foram homologados pela Dimensa. Por isso, recomendamos que sejam utilizados os seguintes sistemas operacionais:

- RedHat Enterprise Linux 8 ou superior
- Oracle Enterprise Linux 8 ou superior
- CentOS 7 ou superior
- OpenSuse Leap 15 ou superior
- Ubuntu Server 20.04 LTS ou superior
- Debian 12 ou superior

A instalação do sistema operacional, bem como sua administração são de responsabilidade do cliente e os passos descritos neste manual somente contemplam os ajustes necessários para a execução das aplicações que serão instaladas.

Java 8

É necessário realizar previamente a instalação da JDK 1.8.0_202 (gratuita) ou, caso possua licença da Oracle, uma versão superior. A utilização da OpenJDK 1.8.0_362 ou superior também é uma possibilidade.

Usuário de Sistema

Para executar o Keycloak recomenda-se a criação de um usuário exclusivo ou ainda utilizar um usuário de serviços já existente. Por questões de segurança, o Keycloak não deverá ser executado com o usuário root ou outro usuário com poderes equivalentes. Para exemplificar, vamos utilizar o usuário **keycloak**.

4. Procedimento de Instalação

4.1. Banco de Dados

O sistema de autenticação utiliza o database ou schema SEGURANCA, podendo variar o nome de acordo com o ambiente (ex. TOTVS_SEGURANCA, DIMENSA_SEGURANCA, etc). O Keycloak efetua a criação das suas tabelas automaticamente, porém existem outras tabelas utilizadas por outros módulos. Os scripts destas outras tabelas são disponibilizados durante a etapa de implantação.

4.2. Servidor do Keycloak

O Keycloak é um servidor de autenticação completo já com seu próprio servidor de aplicação Wildfly e será fornecido o arquivo compactado contendo as configurações padrões. Este arquivo deverá ser instalado no diretório **/opt/app/keycloak-3.4.0.Final** e o mesmo deverá pertencer ao usuário **keycloak** ou outro usuário de serviço estipulado na seção de pré-requisitos. É possível também ajustar a ACL do diretório para o usuário, desde que o mesmo tenha permissão de leitura, escrita e execução. Deve-se verificar se todos os arquivos com extensão .sh no diretório **/opt/app/keycloak-3.4.0.Final/bin** estão com permissão de execução.

4.3. Arquivo standalone.conf

O arquivo **standalone.conf** está localizado no diretório **/opt/app/keycloak-3.4.0.Final/bin** e é onde se define as configurações do **JAVA_HOME**, **JAVA_OPTS** e **SERVER_OPTS**. Caso a JDK ou a OpenJDK esteja no **PATH**, a variável **JAVA_HOME** poderá ser comentada ou retirada do arquivo, caso contrário é necessário informar o caminho da mesma. Os parâmetros de memória na variável **JAVA_OPTS** poderá ser ajustado de acordo com o tamanho do ambiente e a carga no servidor de autenticação.

Abaixo valores sugeridos para cada uma dessas variáveis:

```
JAVA_HOME="/opt/jdk1.8.0_202"
```

```
JAVA_OPTS="-Xms1G -Xmx1G"
```

```
JAVA_OPTS="$JAVA_OPTS -Djava.net.preferIPv4Stack=true -Duser.language=pt -Duser.country=BR"
```


4.4. Arquivo standalone.xml

Neste arquivo são definidas as configurações de rotação de logs, configuração de datasource, portas, entre outros e o mesmo está localizado no diretório `/opt/app/keycloak-3.4.0.Final/standalone/configuration`.

4.4.1. Configuração da Rotação de Log

Dentro da tag `<subsystem xmlns="urn:jboss:domain:logging:3.0">` é possível efetuar a configuração de rotação de logs, permitindo parametrizar o tipo, padrão, tamanho máximo, entre outros. Normalmente configura-se a rotação por tamanho de arquivo, conforme exemplo abaixo, onde o tamanho do arquivo está para 50Mb e serão mantidos os últimos 5 arquivos.

```
<size-rotating-file-handler name="FILE">
    <formatter>
        <pattern-formatter pattern="%d{HH:mm:ss,SSS} %-5p [%c] (%t) %s%E%n"/>
    </formatter>
    <file relative-to="jboss.server.log.dir" path="server.log"/>
    <rotate-size value="50m"/>
    <max-backup-index value="5"/>
    <append value="true"/>
</size-rotating-file-handler>
```

Para utilização do tipo de rotação de logs de periodicidade diária é necessário trocar toda a tag `size-rotating-file-handler` pelas tags abaixo. Neste exemplo, está configurado para criar um arquivo por dia e quando o mesmo atingir 200Mb, é gerado um novo arquivo com a mesma data anexando uma numeração no final do arquivo, limitando-se a 10 arquivos por dia no máximo.

```
<periodic-size-rotating-file-handler name="FILE" autoflush="true">
    <encoding value="UTF-8"/>
    <formatter>
        <named-formatter name="PATTERN"/>
    </formatter>
    <file relative-to="jboss.server.log.dir" path="server.log"/>
    <rotate-size value="200M"/>
    <max-backup-index value="10"/>
    <suffix value=".yyyy-MM-dd"/>
    <append value="true"/>
</periodic-size-rotating-file-handler>
```

4.4.2. Configuração do Datasource

Os procedimentos abaixo apresentam as informações de configuração do Datasource, local onde é informado qual servidor de banco de dados será usado, bem como usuário e senha de conexão.

As informações abaixo correspondem as configurações aplicadas no servidor do Keycloak de acordo com o tipo de banco de dados utilizado.

Oracle

Para configurar o datasource para o banco de dados Oracle, é necessário substituir as seguintes informações no exemplo abaixo.

HOST – Endereço IP ou hostname do servidor Oracle

PORTA – Porta do servidor, normalmente 1521

SERVICE-NAME – Service Name do banco de dados

USUARIO_BD – Deve-se utilizar o owner SEGURANCA ou equivalente

SENHA_BD – Senha do owner.

```
<datasource jndi-name="java:jboss/datasources/KeycloakDS" pool-name="KeycloakDS"
enabled="true" use-java-context="true">
  <connection-url>jdbc:oracle:thin:@//HOST:PORTA/SERVICE-NAME</connection-url>
  <driver>oracle</driver>
  <pool>
    <min-pool-size>10</min-pool-size>
    <max-pool-size>60</max-pool-size>
    <prefill>>false</prefill>
    <flush-strategy>FailingConnectionOnly</flush-strategy>
  </pool>
  <security>
    <user-name>USUARIO_BD</user-name>
    <password>SENHA_BD</password>
  </security>
  <validation>
    <check-valid-connection-sql>SELECT 1 FROM REALM</check-valid-connection-sql>
    <validate-on-match>>true</validate-on-match>
    <background-validation>>true</background-validation>
    <background-validation-millis>1000</background-validation-millis>
    <use-fast-fail>>true</use-fast-fail>
  </validation>
  <timeout>
    <set-tx-query-timeout>>false</set-tx-query-timeout>
    <blocking-timeout-millis>0</blocking-timeout-millis>
    <idle-timeout-minutes>0</idle-timeout-minutes>
    <query-timeout>0</query-timeout>
    <use-try-lock>0</use-try-lock>
    <allocation-retry>0</allocation-retry>
    <allocation-retry-wait-millis>0</allocation-retry-wait-millis>
  </timeout>
</datasource>
```

MS SQL Server

Para configurar o datasource para o banco de dados MS SQL Server, é necessário substituir as seguintes informações no exemplo abaixo.

HOST – Endereço IP ou hostname do servidor

NOME_BD – Nome do Banco de Dados

NOME_INSTANCIA – Nome da instância

USUARIO_BD – Usuário com permissão para acessar o database SEGURANCA ou equivalente

SENHA_BD – Senha do usuário.

```
<datasource jndi-name="java:jboss/datasources/KeycloakDS" pool-name="KeycloakDS"
enabled="true" use-java-context="true">
  <connection-
url>jdbc:sqlserver://HOST;DatabaseName=NOME_BD;instanceName=NOME_INSTANCIA;sendStringParam
etersAsUnicode=false<connection-url>
  <driver>sqlserver</driver>
  <pool>
    <min-pool-size>10</min-pool-size>
    <max-pool-size>60</max-pool-size>
    <prefill>false</prefill>
    <flush-strategy>FailingConnectionOnly</flush-strategy>
  </pool>
  <security>
    <user-name>USUARIO_BD</user-name>
    <password>SENHA_BD</password>
  </security>
  <validation>
    <check-valid-connection-sql>SELECT 1 FROM REALM</check-valid-connection-sql>
    <validate-on-match>true</validate-on-match>
    <background-validation>true</background-validation>
    <background-validation-millis>1000</background-validation-millis>
    <use-fast-fail>true</use-fast-fail>
  </validation>
  <timeout>
    <set-tx-query-timeout>false</set-tx-query-timeout>
    <blocking-timeout-millis>0</blocking-timeout-millis>
    <idle-timeout-minutes>0</idle-timeout-minutes>
    <query-timeout>0</query-timeout>
    <use-try-lock>0</use-try-lock>
    <allocation-retry>0</allocation-retry>
    <allocation-retry-wait-millis>0</allocation-retry-wait-millis>
  </timeout>
</datasource>
```

4.5. Serviço de Inicialização

Para o Keycloak executar como serviço utilizamos o gerenciador de serviços **systemd** do Linux. Os passos abaixo são para criar e habilitar o serviço para iniciar o servidor de autenticação.

Inicialmente deve-se criar o arquivo **keycloak.service** no diretório **/etc/systemd/system** e para isto é necessário estar logado com o usuário **root** ou algum usuário com poder equivalente a super usuário. O conteúdo do arquivo deve ser o seguinte:

```
[Unit]
Description=DIMENSA - Keycloak 3.4.0-Final

[Service]
Type=simple
User=keycloak
RemainAfterExit=yes
ExecStart=/opt/app/keycloak-3.4.0.Final/bin/standalone.sh
Restart=on-failure
RestartSec=120s

[Install]
WantedBy=multi-user.target
```

Após salvar o arquivo, é necessário digitar o seguinte comando para carregar o novo serviço adicionado:

```
systemctl daemon-reload
```

Para habilitar a execução automática do serviço do Keycloak durante a inicialização do servidor, é necessário digitar o seguinte comando:

```
systemctl enable keycloak
```

Para iniciar o serviço do Keycloak, é necessário digitar o seguinte comando:

```
systemctl start keycloak
```

4.6. Criação do Usuário Administrador

Após o Keycloak estar instalado e executando, é necessário criar um usuário para administração e configuração dos sistemas. Os passos abaixo criam o usuário **root** com a senha temporária **admin123** e após este procedimento é necessário reiniciar o serviço do Keycloak para que o usuário seja adicionado efetivamente.

```
cd /opt/app/keycloak-3.4.0.Final/bin
./add-user-keycloak.sh -u root -p admin123
systemctl restart keycloak
```

5. Administração Keycloak

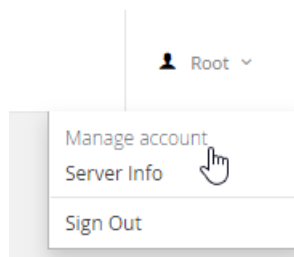
O console de administração do Keycloak serve para configurar os clients das aplicações, integração com LDAP / Active Directory e a Federação com os sistemas da Dimensa. Para acessá-lo é necessário digitar a seguinte URL no navegador e fornecer o usuário e senhas cadastrados no passo 4.6.

<http://nome-do-servidor:8180/auth/admin>

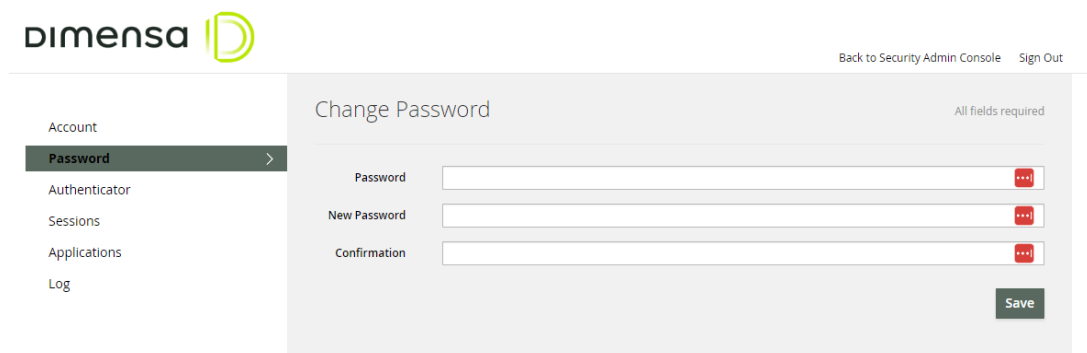
5.1. Alteração de senha do usuário administrador

Após o primeiro acesso, caso não queira utilizar a mesma senha cadastrada, é necessário seguir os seguintes passos para efetuar a troca:

1. Acessar o menu **Manage account** clicando em cima da seta ao lado do nome do usuário, no canto superior direito da tela.



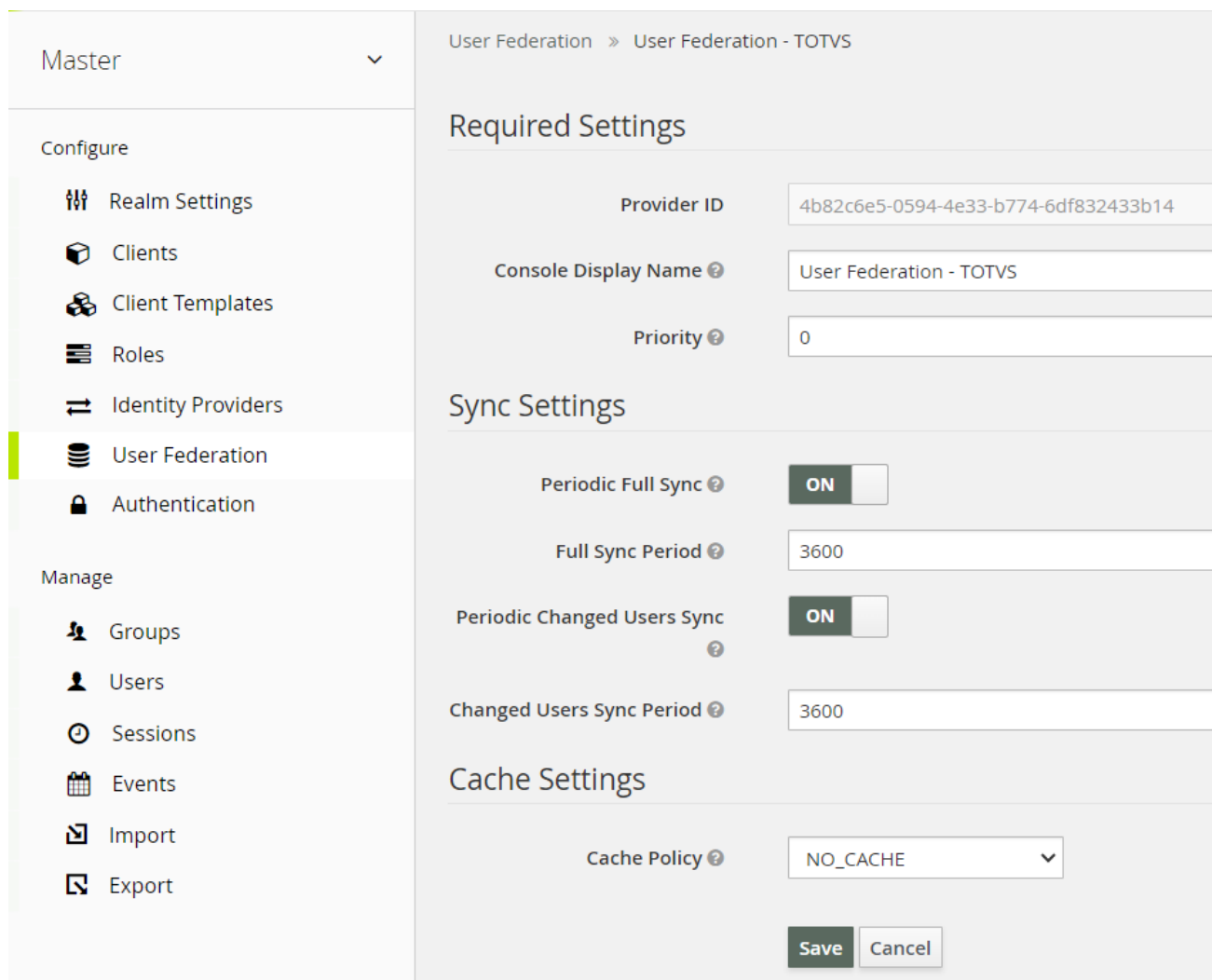
2. Clicar na opção **Password**, digitar a senha atual, a nova senha, confirma-la e clicar em **Save**.



5.2. Criação da Federação

Para configurar o plugin de federação é necessário seguir os passos abaixo. Este plugin é responsável por permitir a autenticação dos usuários já cadastrados na base de dados do módulo Segurança.

1. Clicar no menu **User Federation** do lado esquerdo da tela
2. No combo **Add provider**, escolher o item “**User Federation – TOTVS**”
3. Configurar a tela conforme a imagem abaixo e no final clicar em **Save**

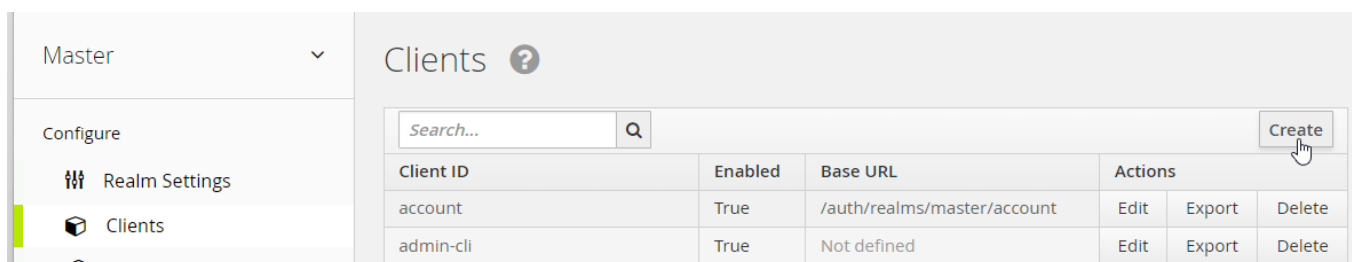


The screenshot shows the configuration interface for a User Federation provider in Keycloak. The left sidebar contains a navigation menu with sections 'Configure' and 'Manage'. The 'Configure' section includes 'Realm Settings', 'Clients', 'Client Templates', 'Roles', 'Identity Providers', 'User Federation' (highlighted), and 'Authentication'. The 'Manage' section includes 'Groups', 'Users', 'Sessions', 'Events', 'Import', and 'Export'. The main content area is titled 'User Federation > User Federation - TOTVS' and is divided into three sections: 'Required Settings', 'Sync Settings', and 'Cache Settings'. The 'Required Settings' section includes 'Provider ID' (4b82c6e5-0594-4e33-b774-6df832433b14), 'Console Display Name' (User Federation - TOTVS), and 'Priority' (0). The 'Sync Settings' section includes 'Periodic Full Sync' (ON), 'Full Sync Period' (3600), 'Periodic Changed Users Sync' (ON), and 'Changed Users Sync Period' (3600). The 'Cache Settings' section includes 'Cache Policy' (NO_CACHE). At the bottom right, there are 'Save' and 'Cancel' buttons.

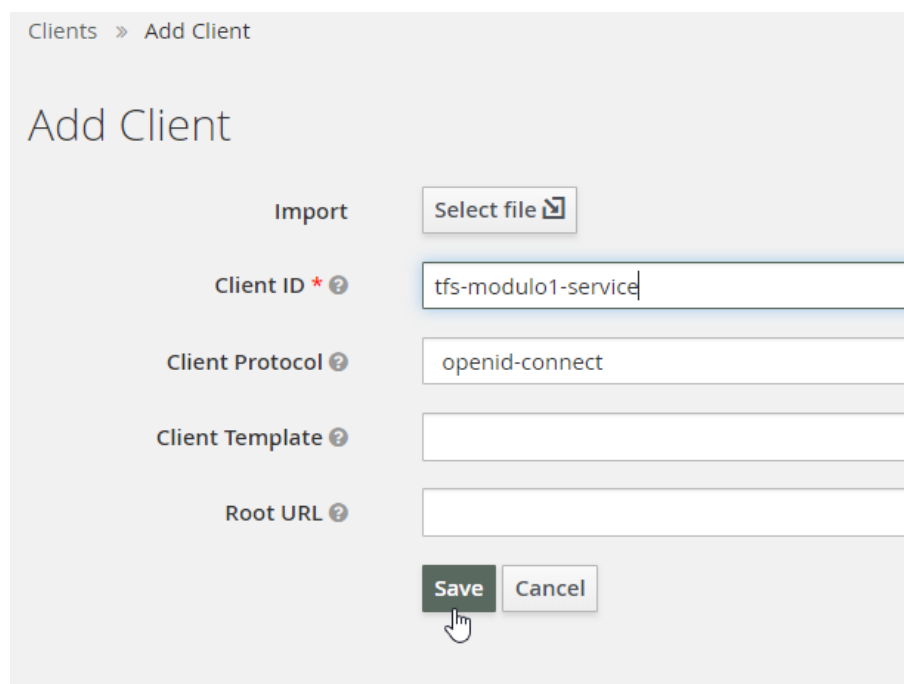
5.3. Criação dos Clients

Nesta etapa serão realizadas as configurações referentes a autenticação dos serviços e portais utilizados pela aplicação. O client será a entidade que a aplicação irá solicitar a autenticação e posteriormente, em caso positivo, obter o acesso ao sistema.

1. Clicar no item Clients, localizado no menu esquerdo da tela e em seguida clicar no botão **Create**.



2. Na tela de adição do client, é obrigatório preencher o campo **Client ID** com os dados fornecidos no documento de implantação de cada módulo. Em seguida, deve-se clicar em **Save** e aguardar a próxima tela.



3. Na tela de configuração deve-se preencher com o restante das informações fornecidas nos documentos de implantação dos módulos e no final clicar em **Save**.

Clients > tfs-modulo1-service

Tfs-modulo1-service

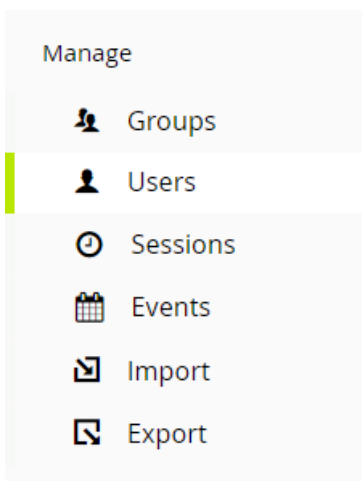
Settings Roles Mappers Scope Revocation Sessions Offline A

Client ID	<input type="text" value="tfs-modulo1-service"/>
Name	<input type="text"/>
Description	<input type="text"/>
Enabled	<input checked="" type="checkbox"/> ON
Consent Required	<input type="checkbox"/> OFF
Client Protocol	<input type="text" value="openid-connect"/>
Client Template	<input type="text"/>
Access Type	<input type="text" value="public"/>
Standard Flow Enabled	<input checked="" type="checkbox"/> ON
Implicit Flow Enabled	<input type="checkbox"/> OFF
Direct Access Grants Enabled	<input checked="" type="checkbox"/> ON
Authorization Enabled	<input type="checkbox"/> OFF

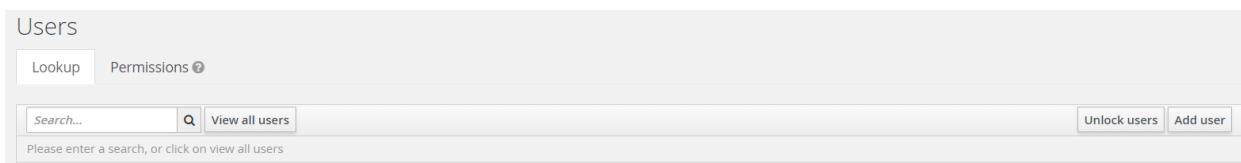
5.4. Criação de Usuários

Nesta etapa serão realizadas as configurações referentes aos usuários de serviço necessários para a utilização de alguns módulos e para isto, é necessário seguir os passos abaixo:

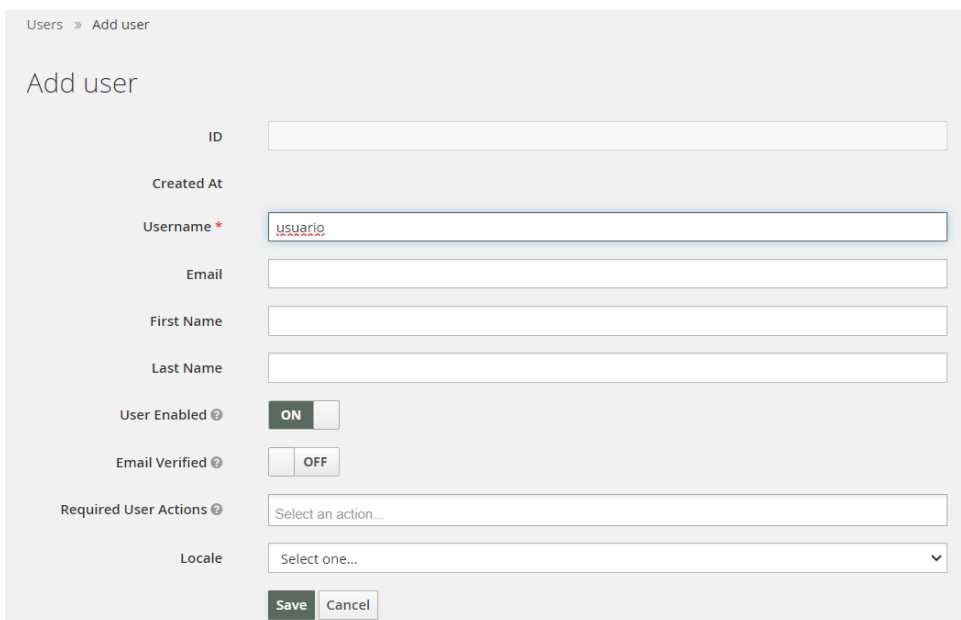
1. Acessar o menu Users do lado esquerdo da tela.



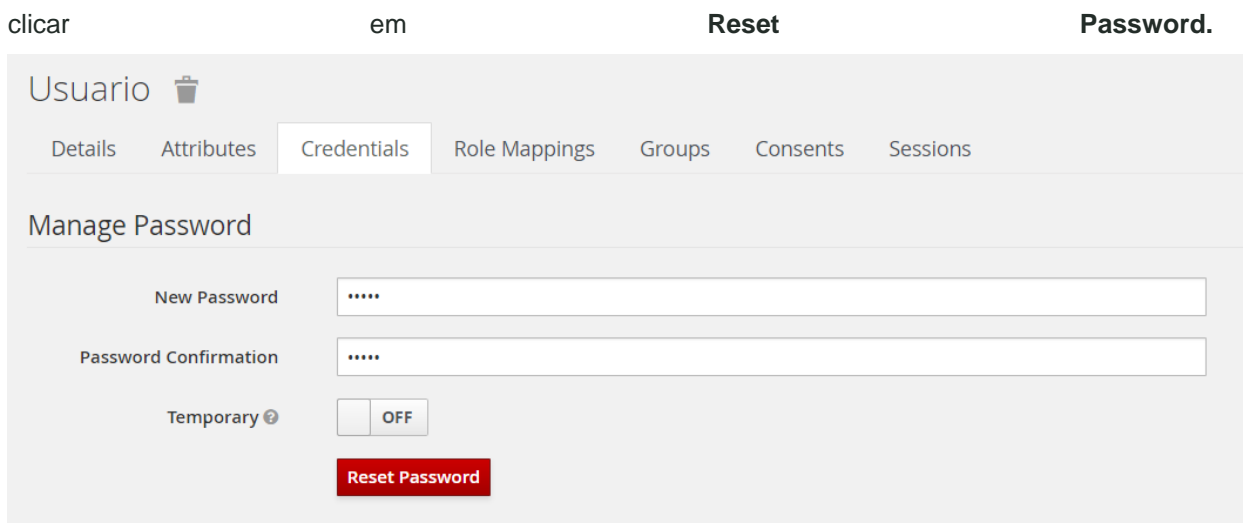
2. Clicar no botão **Add user** do lado direito da tela.



3. Digitar com o nome do usuário no campo **Username** e em seguida clicar em **Save**.



4. Clicar na aba **Credentials** e digitar e confirmar a senha do usuário nos campos **New Password** e **Password Confirmation**. A opção **Temporary** deverá estar selecionada como **OFF** e em seguida,



6. Configuração de Alta Disponibilidade

O Keycloak pode ser executado em dois ou mais servidores, porém é necessário configurá-lo para operar no modo cluster. É pré-requisito que o Keycloak esteja instalado em todos os servidores do cluster de autenticação seguindo os passos anteriores antes de executar os passos a seguir. Antes de iniciar os procedimentos, é necessário parar o serviço do Keycloak usando o comando abaixo:

```
systemctl stop keycloak
```

6.1. Arquivo standalone.conf

O arquivo **standalone.conf** está localizado no diretório **"/opt/app/keycloak-3.4.0.Final/bin"** e é onde se define as configurações do **JAVA_HOME**, **JAVA_OPTS** e **SERVER_OPTS**. Caso a JDK ou a OpenJDK esteja no **PATH**, a variável **JAVA_HOME** poderá ser comentada ou retirada do arquivo. Os parâmetros de memória na variável **JAVA_OPTS** poderá ser ajustado de acordo com o tamanho do ambiente e a carga no servidor de autenticação.

Abaixo valores sugeridos para cada uma dessas variáveis:

```
JAVA_HOME="/opt/jdk1.8.0_202"
```

```
JAVA_OPTS="-Xms1G -Xmx1G"
```

```
JAVA_OPTS="$JAVA_OPTS -Djava.net.preferIPv4Stack=true -Duser.language=pt -Duser.country=BR"
```

```
SERVER_OPTS="--server-config=standalone-ha.xml"
```

6.2. Arquivo standalone-ha.xml

Para habilitar o modo de alta disponibilidade é necessário ajustar alguns parâmetros no arquivo **standalone-ha.xml** tais como configurações de **datasource**, **clusters**, entre outros. Este arquivo localiza-se no diretório **/opt/app/keycloak-3.4.0.Final/standalone/configuration**. As configurações de rotação de logs e datasource são as mesmas abordadas nos tópicos 4.4.1 e 4.4.2.

6.2.1. Configuração do Infinispan e JGroups

Os comandos abaixo são necessários para parametrizar as configurações do Infinispan e do JGroups para o correto sincronismo de sessões e realização do registro de cada um dos nodos do cluster. Os comandos estão no arquivo **comandos_jbosscli-JDBC_PING.txt** compartilhado juntamente com este manual. Para realizar este procedimento, deverá seguir os seguintes passos:

```
export JG_HOST_TCP=<IP ou HOSTNAME do Servidor>
/opt/app/keycloak-3.4.0.Final/bin/jboss-cli.sh -file= comandos_jbosscli-JDBC_PING.txt
```

6.3. Banco de Dados

É necessário criar uma tabela chamada **JGROUPSPING** dentro do database / schema **SEGURANCA**, podendo variar o nome de acordo com o ambiente (ex. TOTVS_SEGURANCA, DIMENSA_SEGURANCA, etc). Esta tabela é fundamental para o Keycloak gerenciar quais servidores pertencem ao cluster. Segue abaixo o DDL de criação das tabelas para os bancos de dados Oracle e MS SQL Server.

Oracle

```
CREATE TABLE JGROUPSPING (
    "OWN_ADDR" VARCHAR2(200) NOT NULL ENABLE,
    "CLUSTER_NAME" VARCHAR2(200) NOT NULL ENABLE,
    "UPDATED" TIMESTAMP (0) DEFAULT SYSTIMESTAMP,
    "PING_DATA" BLOB DEFAULT NULL,
    PRIMARY KEY ("OWN_ADDR", "CLUSTER_NAME")
);
```

MS SQL Server

```
CREATE TABLE JGROUPSPING (
    [OWN_ADDR] VARCHAR(200) NOT NULL,
    [CLUSTER_NAME] VARCHAR(200) NOT NULL,
    [UPDATED] DATETIME2 (0) DEFAULT GETDATE(),
    [PING_DATA] VARBINARY(max) DEFAULT NULL,
    PRIMARY KEY ([OWN_ADDR], [CLUSTER_NAME])
)
GO
```

6.4. Configurações de Rede

Quando o Keycloak opera no modo cluster, a comunicação entre os nodos dá-se através da porta 7600, portanto a mesma deverá ser liberada no firewall, bem como a porta 8180 para acesso aos serviços e console de administração.

6.5. Iniciando o cluster

Antes de iniciar os serviços do Keycloak nos servidores do cluster, é necessário garantir que os horários dos servidores estão iguais. É altamente recomendável sincronizar o horário de todos os servidores do ambiente utilizando o serviço de NTP. Após esta verificação, basta digitar o seguinte comando em cada um dos servidores e aguardar o serviço ficar disponível.

```
systemctl start keycloak
```