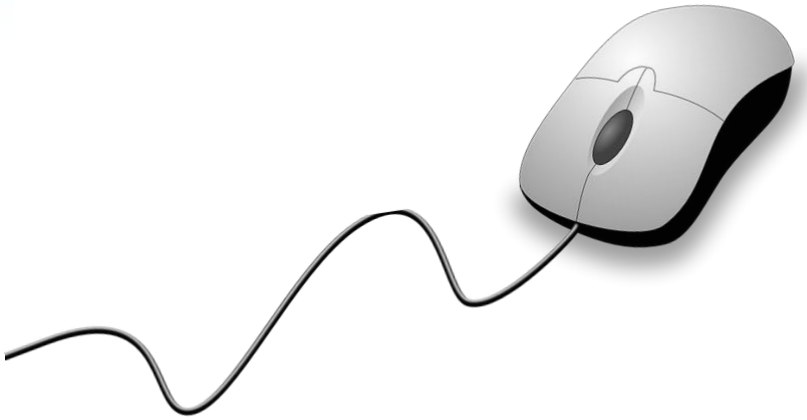
공개SW 솔루션 설치 & 활용 가이드

시스템SW > 보안



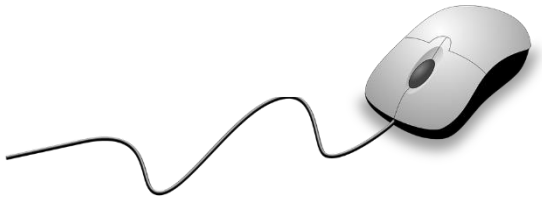
KEYCLOAK

제대로 배워보자

How to Use Open Source Software

Open Source Software Installation & Application Guide

nipa 공개SW역량프라자



CONTENTS

1. 개요

소개	• 최신 응용 프로그램 및 서비스를 대상으로 하는 ID 관리 및 액세스 관리로 Single Sign-On 허용하는 오픈소스 소프트웨어		
주요기능	• 사용자 등록 • 소셜 로그인 • 동일한 영역에 속한 모든 응용 프로그램의 SSO / Sign-Off • LDAP 통합		
대분류	• 시스템SW	**소분류**	• 보안
라이선스 형태	• Apache License 2.0	**사전설치 솔루션**	• Java JDK 8 이상 • Apache Maven 3.1.1 이상
운영체제	• Cross-platform	**버전**	• 4.5.0 (2018년 10월 기준)
특징	• 코딩을 거의 또는 전혀하지 않고 응용 프로그램과 서비스 보안 유지		
보안취약점	• 취약점 ID : CVE-2018-10912 • 심각도 : 4.9 MEDIUM(V3) • 취약점 설명 : 세션 대체에서 무한 루프로 인해 서비스 거부가 발생 • 대응방안 : 4.0.0 이상으로 업그레이드 • 참고 경로 : https://bugzilla.redhat.com/show_bug.cgi?id=CVE-2018-10912		
개발회사/커뮤니티	• Red Hat		
공식 홈페이지	• http://keycloak.org		

2. 기능요약

- Single-Sign On
- 소셜 로그인
- 관리 콘솔
- 계정 관리 콘솔
- 권한 부여 서비스
- 클라이언트 어댑터
- 표준 프로토콜
- User Federation

3. 실행환경

- 운영체제
  - Windows
  - Linux
  - Unix

- Java JDK 8
- Apache Maven 3.1.1 이상

4. 설치 및 실행

세부 목차

4. 설치 및 실행

4.1.1 배포 파일 설치

- https://www.keycloak.org/downloads.html
- 위 사이트에서 Keycloak Server 다운로드

- Linux / Unix

  - $ unzip keycloak-4.5.0.Final.zip or $ tar -xvzf keycloak-4.5.0.Final.tar.gz

- Windows

  - > unzip keycloak-4.5.0.Final.zip

4. 설치 및 실행

4.1.2 서버 부팅

- Keycloak 서버를 부팅하려면 keycloak-4.5.0.Final/bin 디렉터리로 이동하여 standalone 부팅 스크립트 실행

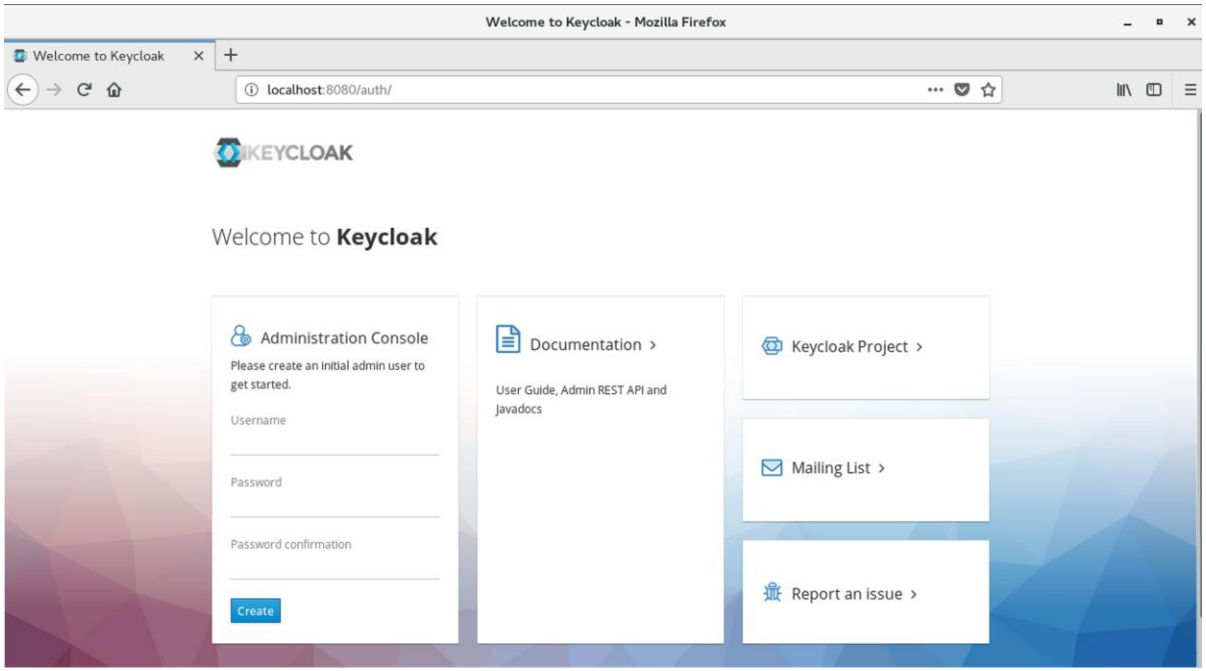
- Linux / Unix
  - $ cd bin
  - $ ./standalone.sh

- Windows
  - > ...₩bin₩standalone.bat

4. 설치 및 실행

4.1.3 관리자 계정 만들기

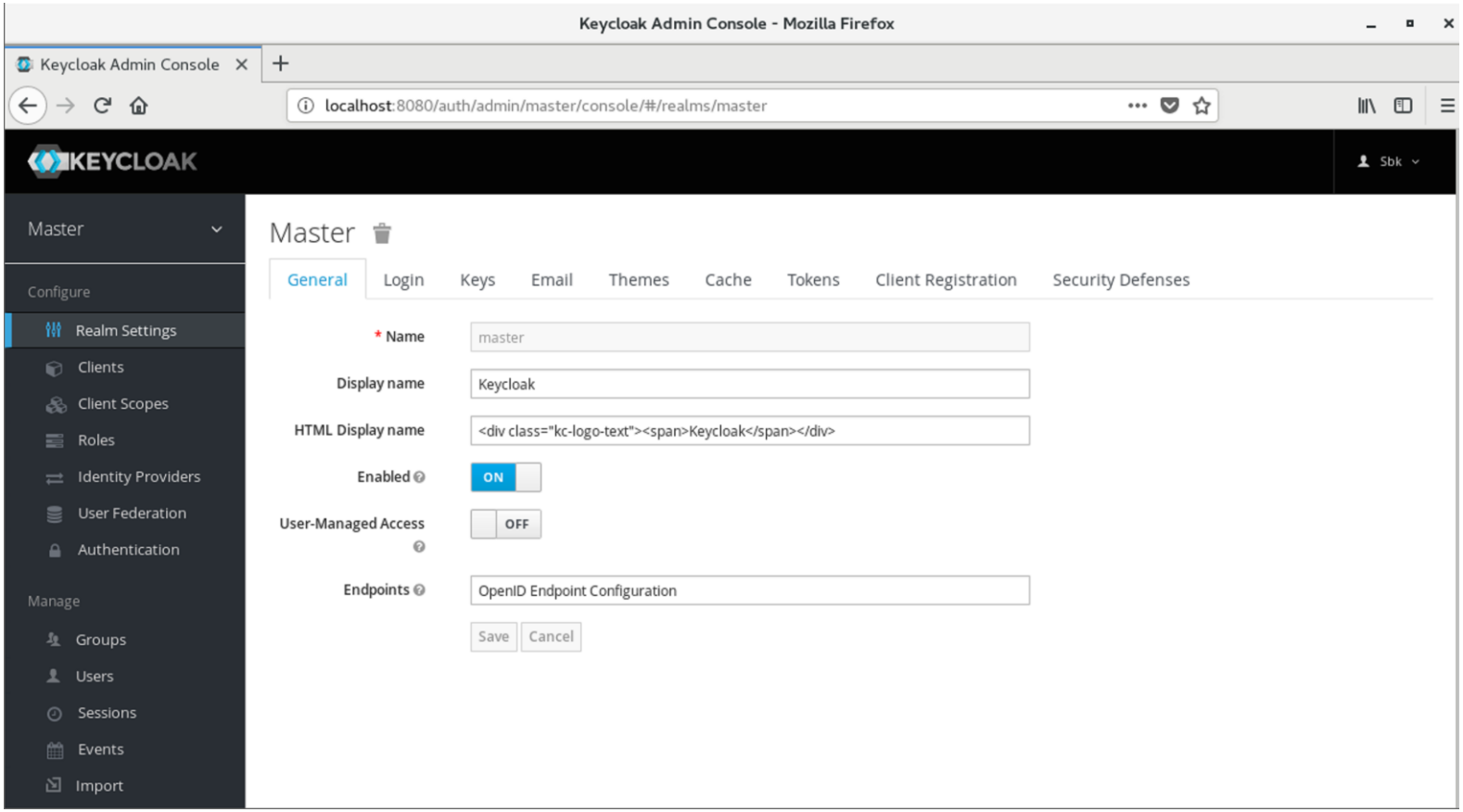
- 서버가 부팅 된 후 웹 브라우저에서 localhost:8080/auth 접속
- 시작페이지는 서버가 실행 중 나타냄
- 사용자 이름과 비밀번호를 입력하여 초기 관리 사용자 만듦
- 이 계정은 master 영역 및 사용자를 생성하고 Keycloak이 보호할 응용 프로그램 등록
- 또한 보호 할 응용 프로그램의 관리 콘솔에 로그인 가능



4. 설치 및 실행

4.1.4 관리 콘솔에 로그인

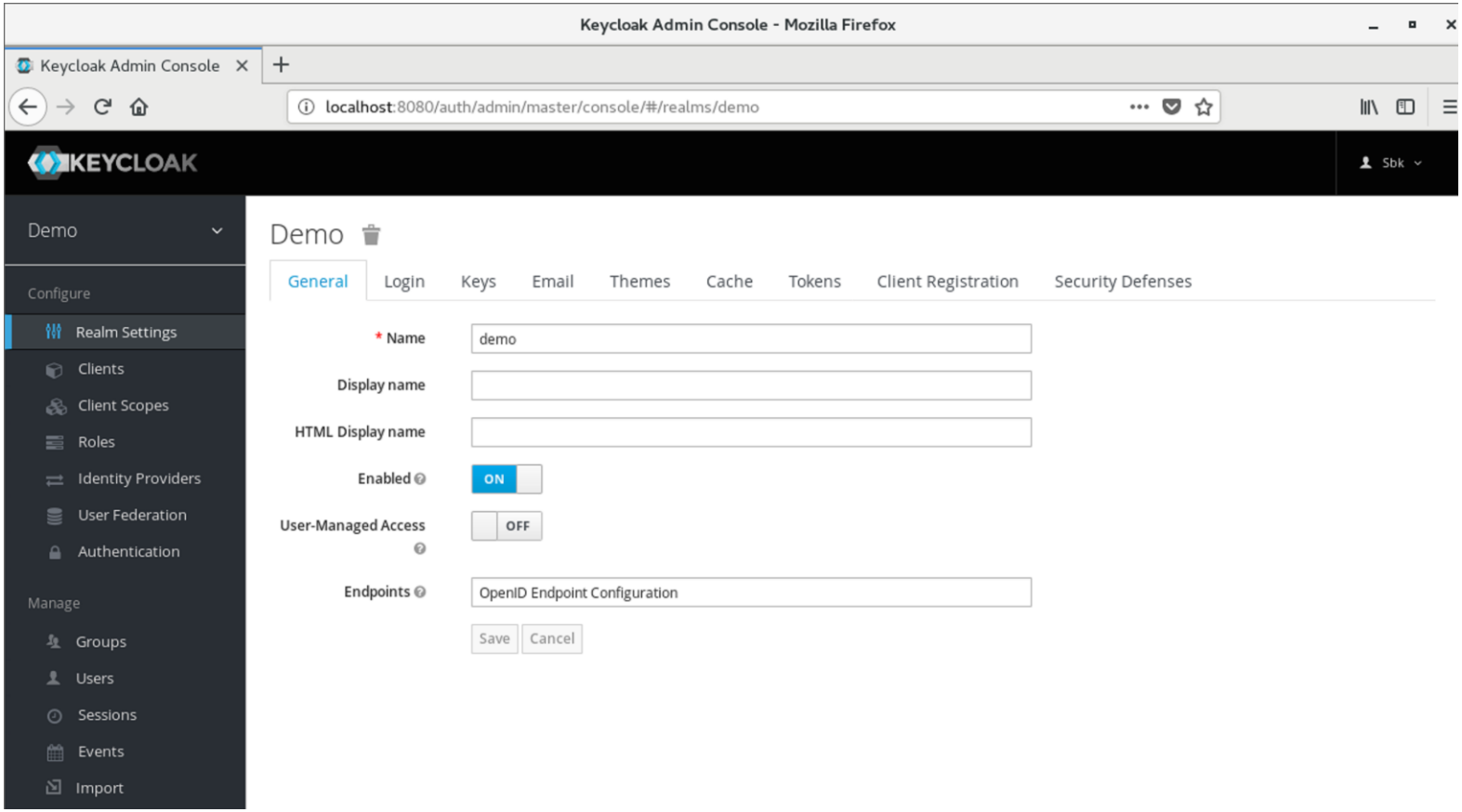
- 초기 관리자 계정을 생성 한 후 http://localhost:8080/auth/admin 으로 접속하여 로그인



4. 설치 및 실행

4.2.1 새 영역 만들기

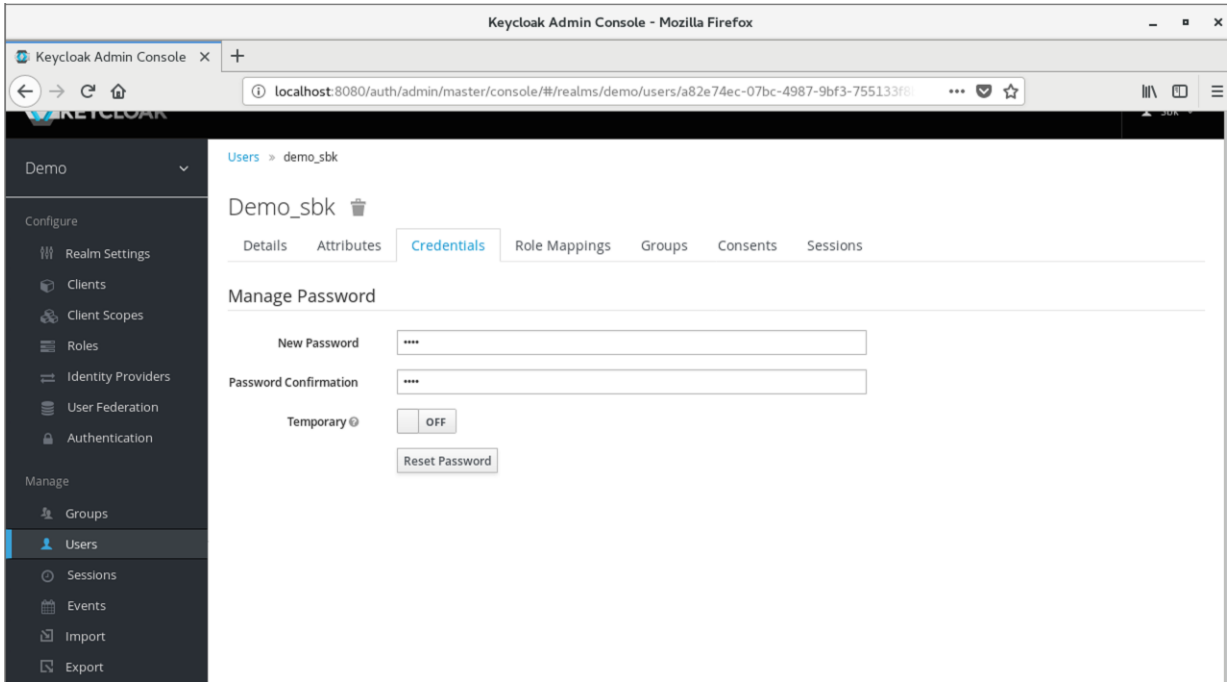
- 왼쪽 상단의 Master 드롭 다운 메뉴를 클릭 후 Add realm 클릭
- Name 에 demo 입력 후 생성



4. 설치 및 실행

4.2.2 새 사용자 만들기

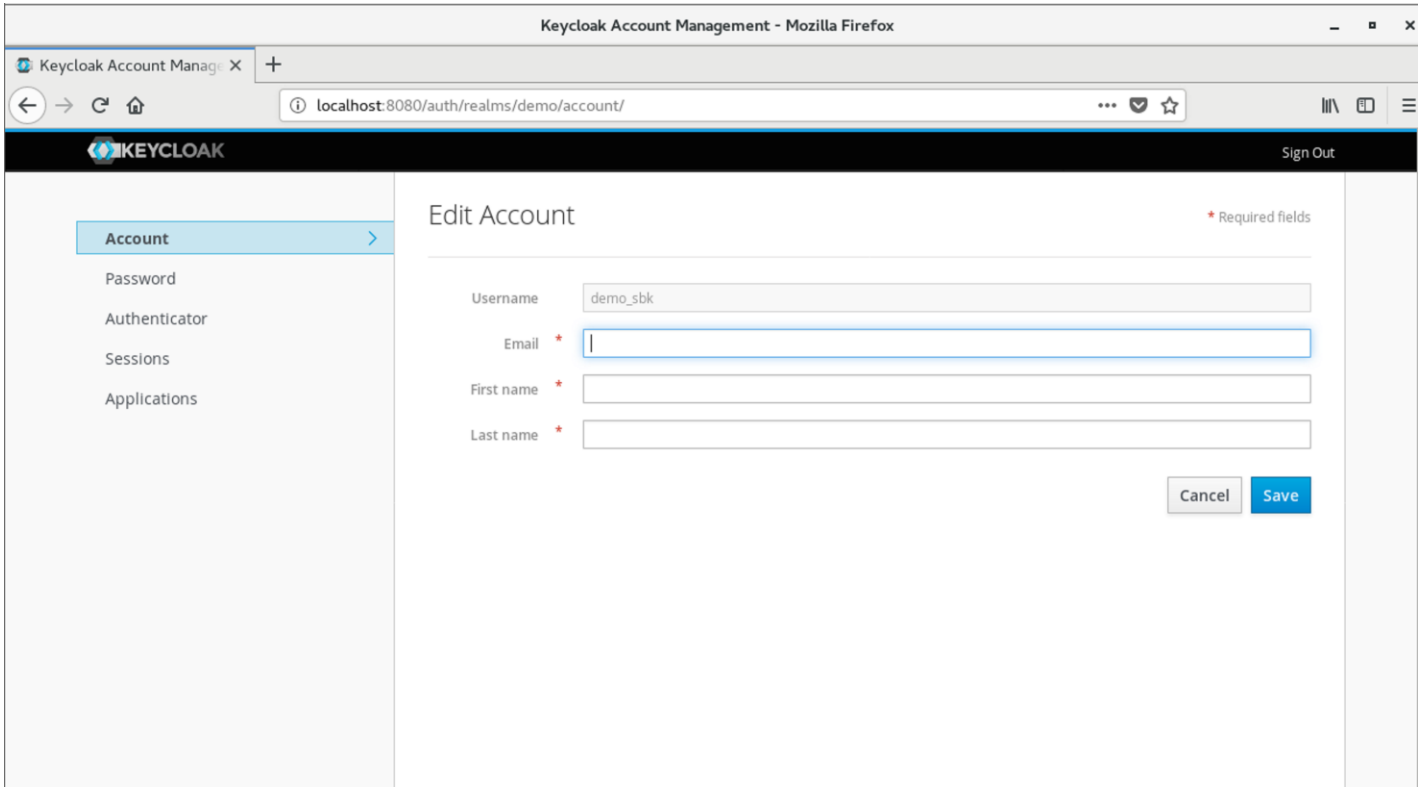
- 왼쪽에 Manage 메뉴에서 Users 클릭 후 Add user 클릭
- Username 필드에 이름을 입력한 후 저장
- Credentials 탭으로 이동 후 패스워드 입력
- Temporary를 on으로 하고 패스워드를 리셋하면 처음 로그인 한 후 암호 변경
- 영구적인 암호를 만드려면 off 로 전환하고 Reset Password 클릭



4. 설치 및 실행

4.2.3 사용자 계정 서비스

- http://localhost:8080/auth/realms/demo/account 로 접속하여 방금 만든 계정으로 로그인
- 사용자 계정 서비스 페이지 열리며, 이 페이지에서 프로필 정보를 업데이트하고 자격을 변경하거나 추가 가능



5. 기능소개

세부 목차

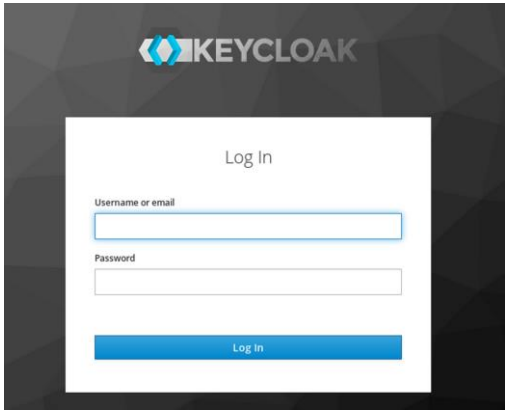
5. 기능소개

5.1 Single-Sign On

- 사용자는 개별 응용프로그램이 아닌 Keycloak으로 인증
- 애플리케이션이 로그인 양식, 사용자 인증 및 사용자 저장에 관여하지 않으며 Keycloak에 로그인한 후에는 사용자가 다른 애플리케이션에 액세스하기 위해 다시 로그인 불필요

  * 이것은 로그아웃에도 적용
- Keycloak은 Single Sign-out을 제공하므로 사용자는 Keycloak을 사용하는 모든 응용 프로그램에서 로그아웃하려면 한번만 로그아웃
- Kerberos bridge

  - 사용자가 Kerberos(LDAP 또는 Active Directory)를 사용하여 워크스테이션에 인증하면 워크스테이션에 로그온한 후 사용자 이름과 암호를 다시 제공할 필요 없이 Keycloak에 자동으로 인증 가능



5. 기능소개

5.2 소셜 로그인

- 소셜 네트워크에서 로그인을 사용하도록 설정하려면 관리 콘솔을 통해 쉽게 추가 가능

5. 기능소개

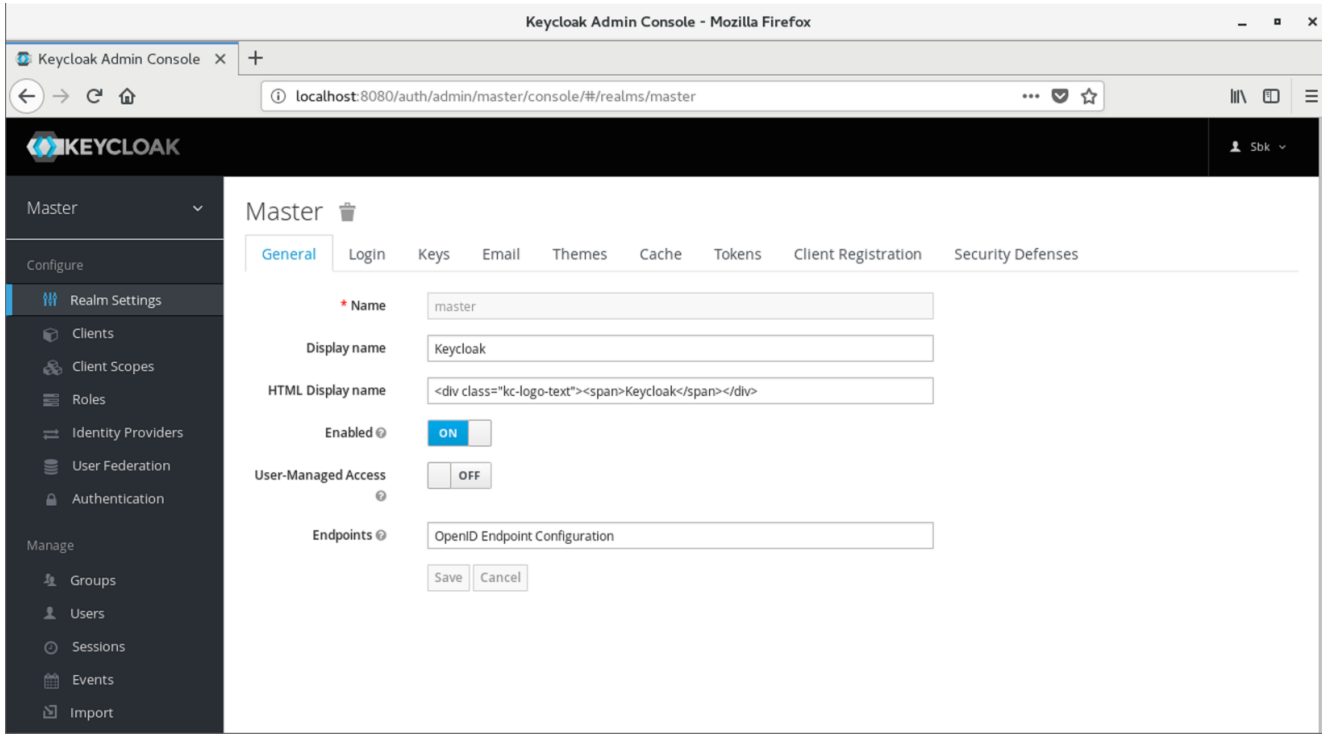
5.3 클라이언트 어댑터

- Keycloak Client Adapters를 사용하면 애플리케이션과 서비스를 매우 쉽게 보호 가능
- 표준 프로토콜을 기반으로 하므로, OpenID Connect Resource Library 또는 SAML2.0 라이브러리를 사용 가능
- 프록시를 사용하여 응용 프로그램을 보호할 수도 있으므로 응용 프로그램을 수정 불필요

5. 기능소개

5.4 관리 콘솔

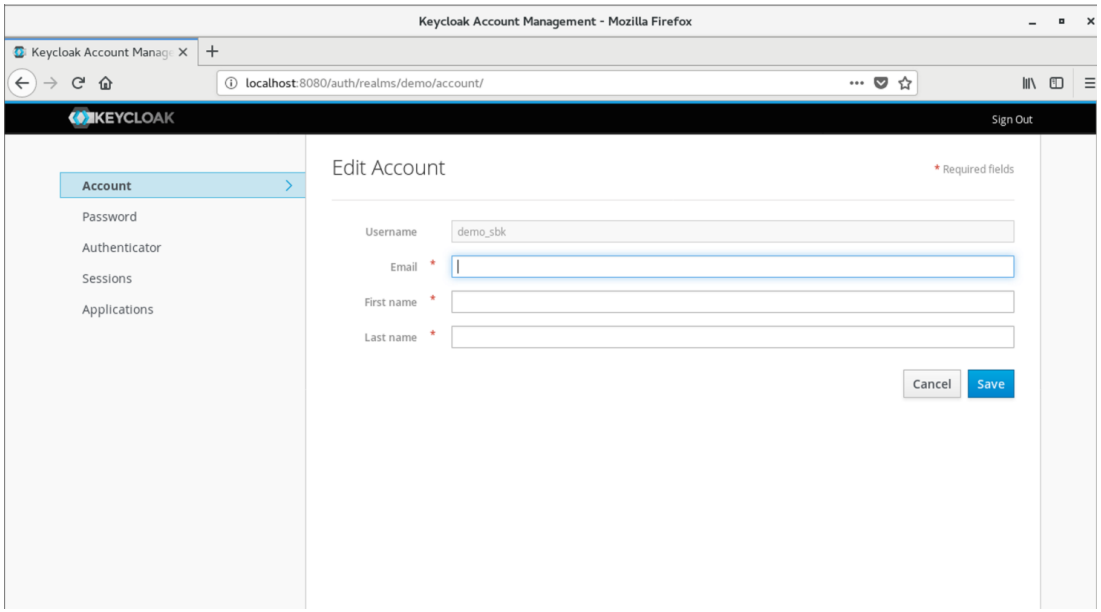
- 관리 콘솔을 통해 관리자는 Keycloak서버의 모든 측면을 중앙에서 관리 가능
- 다양한 기능을 활성화 및 비활성화 가능
- 애플리케이션 및 서비스를 생성 및 관리하고 세부적인 권한 부여 가능
- 권한 및 세션을 포함하여 사용자 관리 가능



5. 기능소개

5.5 계정 관리 콘솔

- 계정 관리 콘솔을 통해 사용자는 자신의 계정 관리 가능
- 프로필을 업데이트하고 암호를 변경하고 2단계 인증 설정 가능
- 세션을 관리하고 계정 기록 확인

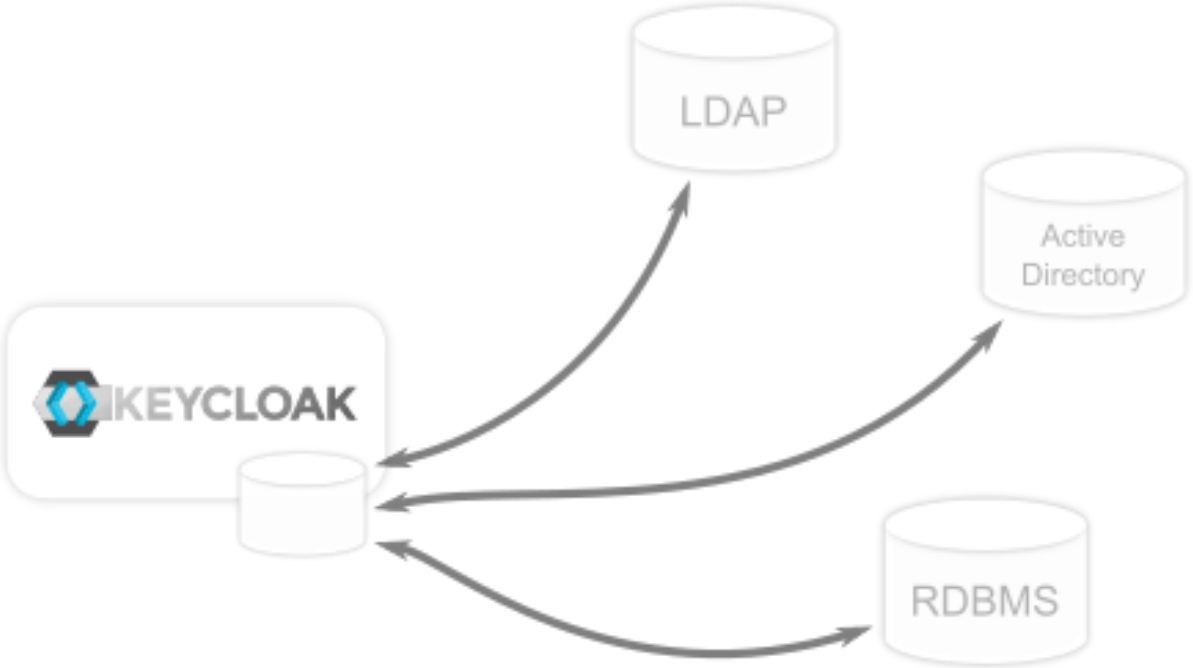


- Keycloak은 표준 프로토콜을 기반으로 하며 OpenID Connect, OAuth2.0및 SAML 지원

5. 기능소개

5.7 User Federation

- Keycloak에는 기존 LDAP또는 ActiveDirectory서버에 연결할 수 있는 지원 기능 내장
- 관계형 데이터베이스와 같은 다른 스토어에 사용자가 있는 경우 자체 provider를 구현 가능



- Keycloak은 세부적인 권한 부여 서비스도 제공
  - 이렇게 하면 Keycloak 관리 콘솔에서 모든 서비스에 대한 권한을 관리할 수 있으며 정확하게 정의할 수 있는 권한 제공

6. 활용예제

세부 목차

6. 활용예제

6.1 WildFly Servlet 애플리케이션 보안을 시작하기전에

- 활용예제에서는 WildFly 응용 프로그램 서버에서 java servlet 응용 프로그램을 보안하는 방법 설명
  - WildFly 응용 프로그램 서버 배포에 Keycloak 클라이언트 어댑터 설치
  - Keycloak 관리 콘솔에서 클라이언트 응용 프로그램 만들기 및 등록
  - Keycloak으로 보호할 응용 프로그램 구성

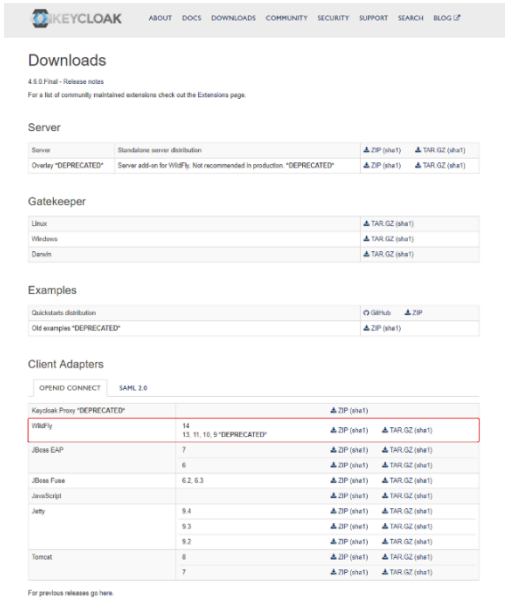
- Java Servlet 애플리케이션을 보호하려면 먼저 Keycloak 설치를 완료하고 초기 관리 사용자 생성
  - WildFly가 Keycloak과 함께 번들로 되어 있지만 응용프로그램 컨테이너로 사용 불가
  - Java servlet 애플리케이션을 실행하려면 Keycloak 서버와 동일한 시스템에서 별도의 WildFly 인스턴스를 실행 해야함
  - 포트 충돌을 방지하기 위해 WildFly와는 다른 포트를 사용하여 Keycloak 실행

- 포트를 조정하면서 Keycloak 서버를 시작하려면 아래와 같음
- Linux / Unix
  - $ cd bin
  - $ ./standalone.sh -Djboss.socket.binding.port-offset=100
- Windows
  - > ...₩bin₩standalone.bat -Djboss.socket.binding.port-offset=100

6. 활용예제

6.2 클라이언트 어댑터 설치

- https://www.keycloak.org/downloads.html 에서 WildFly adapter 다운로드
- Adapter를 keycloak의 하위디렉터리에 압축해제 후 bin 폴더로 이동하여 스크립트 실행



- Linux / Unix
  - $ cd keycloak-4.5.0.Final/bin
  - $ ./jboss-cli.sh --file=adapter-elytron-install-offline.cli
  - $ ./standalone.sh
- Windows
  - > cd keycloak-4.5.0.Final₩bin
  - > jboss-cli.bat --file=adapter-elytron-install-offline.cli
  - > ...keycloak-4.5.0.Final₩bin₩standalone.bat
- Keycloak 시작한 후 localhost:8080/auth/admin/으로 이동하여 관리 콘솔에 로그인

6. 활용예제

6.3 응용 프로그램 코드 다운로드, 빌드 및 배포

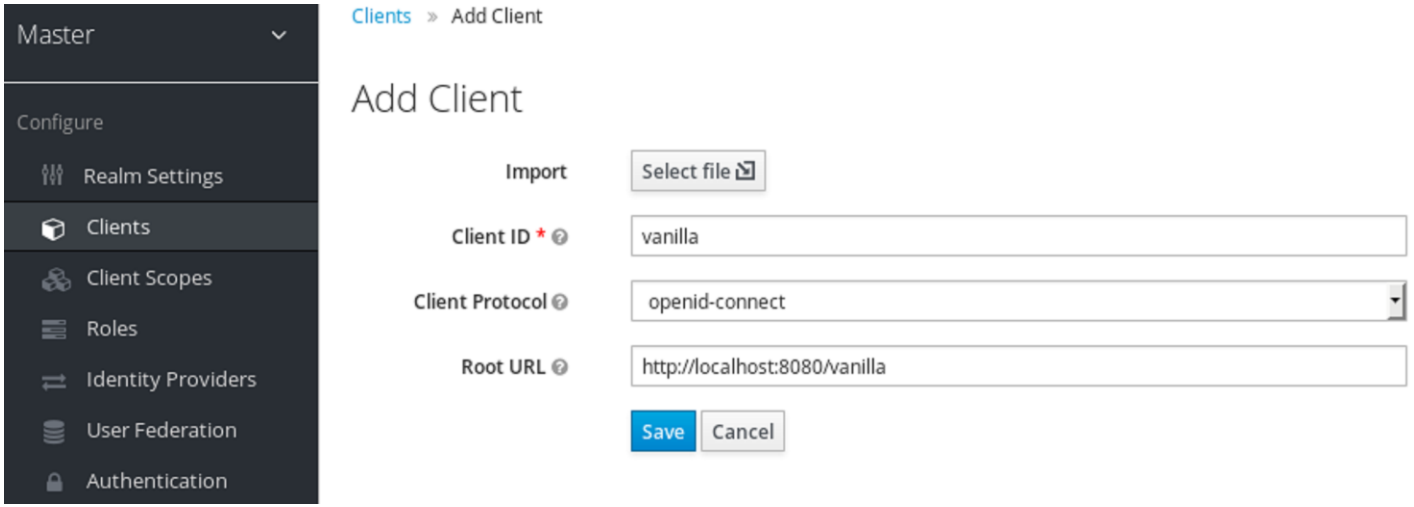
- $ git clone https://github.com/keycloak/keycloak-quickstarts
- $ cd keycloak-quickstarts/app-profile-jee-vanilla
- $ mvn clean wildfly:deploy
- http : // localhost : 8080 / vanilla 로 이동하면 로그인 페이지가 나타남

6. 활용예제

6.4 클라이언트 생성 및 등록(1/2)

- keycloak관리 콘솔에 관리자 계정으로 로그인
- 왼쪽 상단 드롭 다운 메뉴에서 Master영역을 선택
- Configure 메뉴에서 Clients 클릭
- 오른쪽에 Create 클릭
- 그림과 같이 작성 후 저장
- Client ID : vanilla

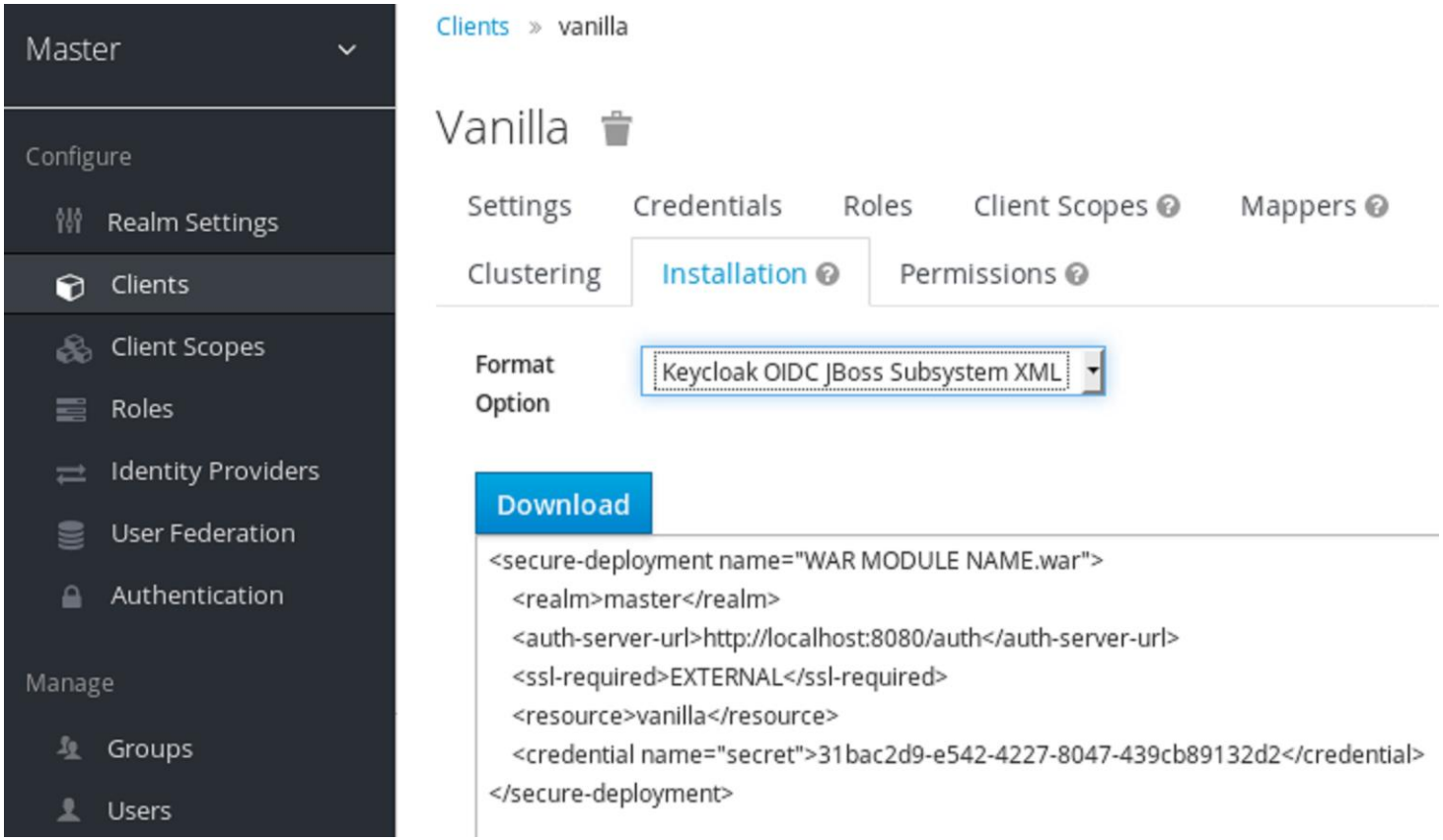
  Root URL : http://localhost:8080/vanilla



6. 활용예제

6.4 클라이언트 생성 및 등록(2/2)

- Installation 클릭 후 Format Option을 Keycloak OIDC JBoss Subsystem XML 으로 선택 후 내용을 복사



6. 활용예제

6.5 하위 시스템 구성(1/2)

- 응용 프로그램이 Keycloak에 의해 보호되도록 xml 파일을 수정

- keycloak-4.5.0.Final/standalone/configuration/standalone.xml 파일을 열고 다음 텍스트를 검색
  - <subsystem xmlns="urn:jboss:domain:keycloak:1.1"/>
- 태그를 아래와 같이 수정 후 복사한 내용을 붙여 넣음
  - <subsystem xmlns="urn:jboss:domain:keycloak:1.1">
  - </subsystem>

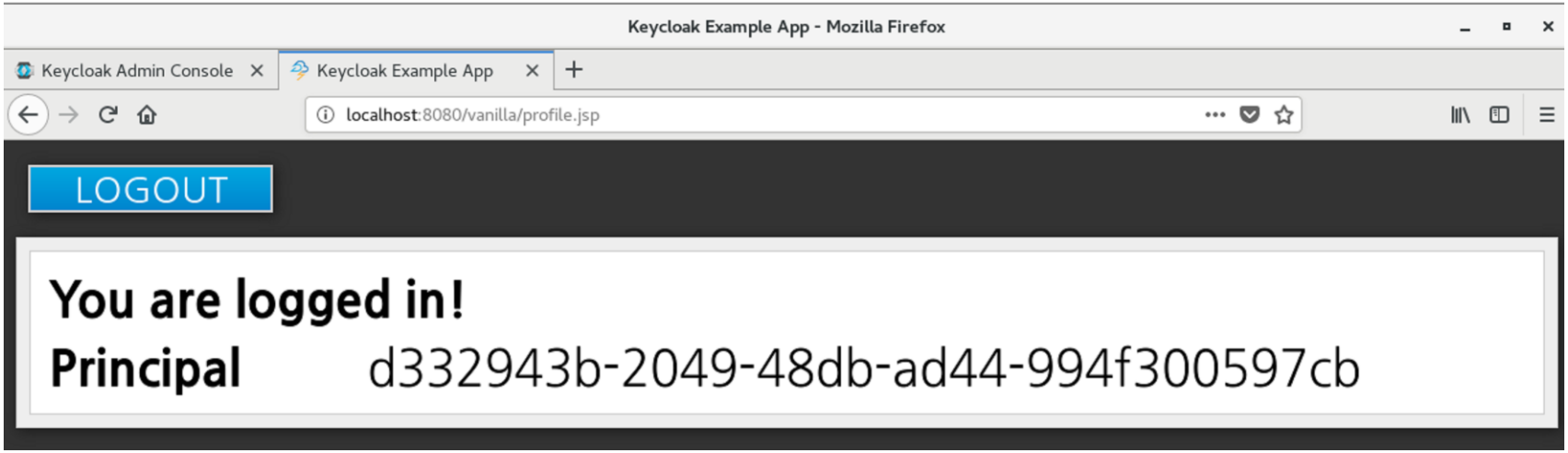
```
<subsystem xmlns="urn:jboss:domain:keycloak:1.1">
    <secure-deployment name="vanilla.war">
        <realm>master</realm>
        <auth-server-url>http://localhost:8080/auth</auth-server-url>
        <ssl-required>EXTERNAL</ssl-required>
        <resource>vanilla</resource>
        <credential name="secret">31bac2d9-e542-4227-8047-439cb89132d2
        </credential>
    </secure-deployment>
</subsystem>
```

- <secure-deployment name="WAR MODULE NAME.war">를 <secure-deployment name="vanilla.war">로 변경
- 응용 프로그램 서버 재기동
- http://localhost:8080/vanilla 로 이동하여 로그인 클릭
- Keycloak 로그인 페이지가 열리면 Master 계정을 사용하여 로그인

6. 활용예제

6.5 하위 시스템 구성(2/2)

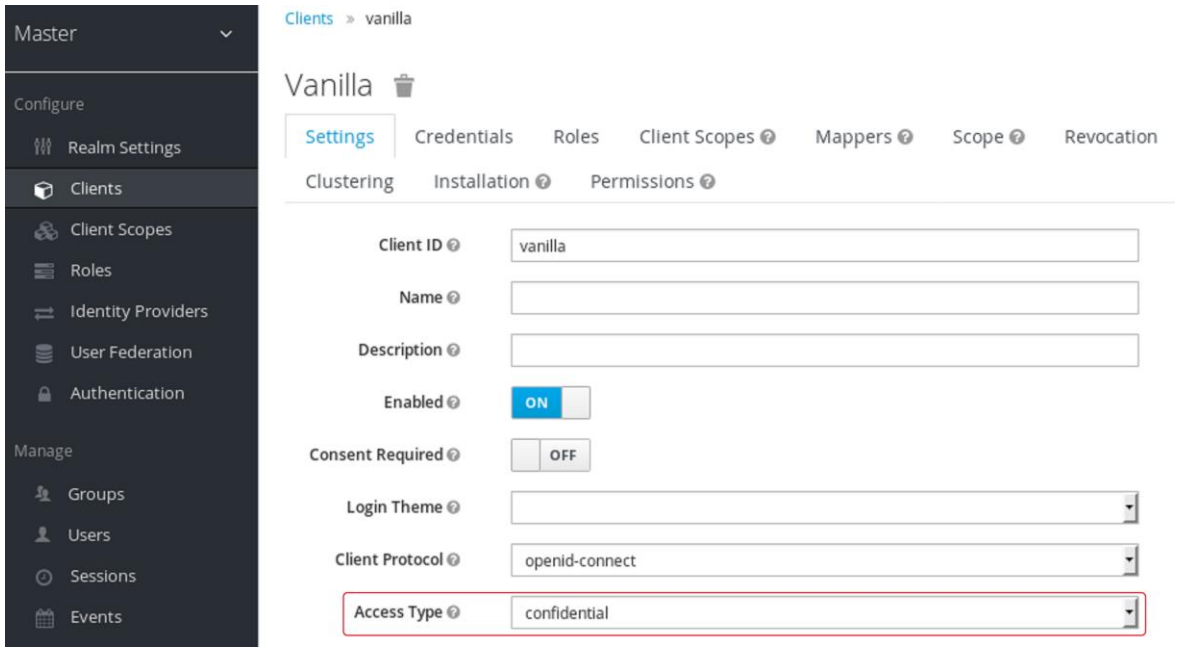
- 아래의 화면이 나온다면 성공



7. FAQ

Q **XML 에서 credential이 없어요?**

A Clients - Settings에서 Access Type을 보시면 public으로 설정 되어있습니다.
이것을 confidential로 바꿔 주시면 됩니다.



8. 용어정리

용어	설 명
Realm	인증, 인가가 작동하는 범위를 나타내는 단위 SSO(Single Sign-On)를 예로 들면 특정 클라이언트들이 SSO를 공유한다면 그 범위는 그 클라이언트들이 공통적으로 속한 Realm에 한정되며, 기본적으로 삭제가 불가능한 Master라는 Realm 제공
Client	Keycloack에게 인증, 인가 행위를 대행하도록 맡길 애플리케이션을 나타내는 단위 웹사이트 일수도 있고, REST API를 제공하는 서비스일수도 있으며, 하나의 Realm은 자신에게 종속된 n개의 Client를 생성하고 관리 가능
User	실제 각 Client에 로그인할 사용자를 나타내는 단위이며, 하나의 Realm은 자신에게 종속된 n개의 User를 생성하고 관리 가능함 기본적으로 User 개체는 Username, Email, First Name, Last Name 4개 항목을 가질 수 있는데 Custom User Attributes 기능을 통해 커스텀 항목을 자유롭게 추가 가능

Open Source Software Installation & Application Guide

nipa 공개SW역량프라자