AN NInh MẠNG  
Bài tập lớn

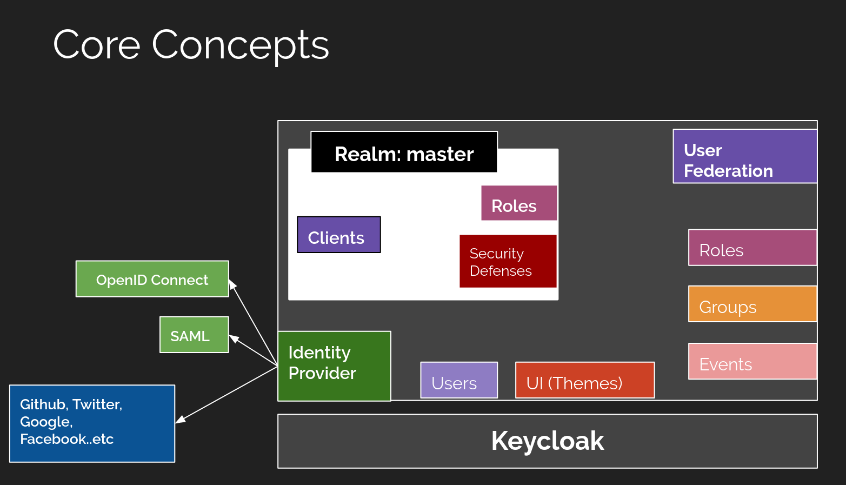
15/8/2021

# Overview

## Giới thiệu

|  |  |
| --- | --- |
|  | Để triển khai một hệ thống quản lý định danh và truy cập(IAM - Identity and Access Management), nhóm sẽ chon **Keycloak**, để báo cáo. |
|  | Keycloak cung cấp sự linh hoạt để xuất và nhập cấu hình một cách dễ dàng, sử dụng một chế độ xem duy nhất để quản lý mọi thứ. Cùng với nhau, những công nghệ này cho phép bạn tích hợp các ứng dụng front-end, di động và nguyên khối vào một kiến trúc microservice. |

**Khái niệm cốt lõi**



***Hình 1 minh hoạ***

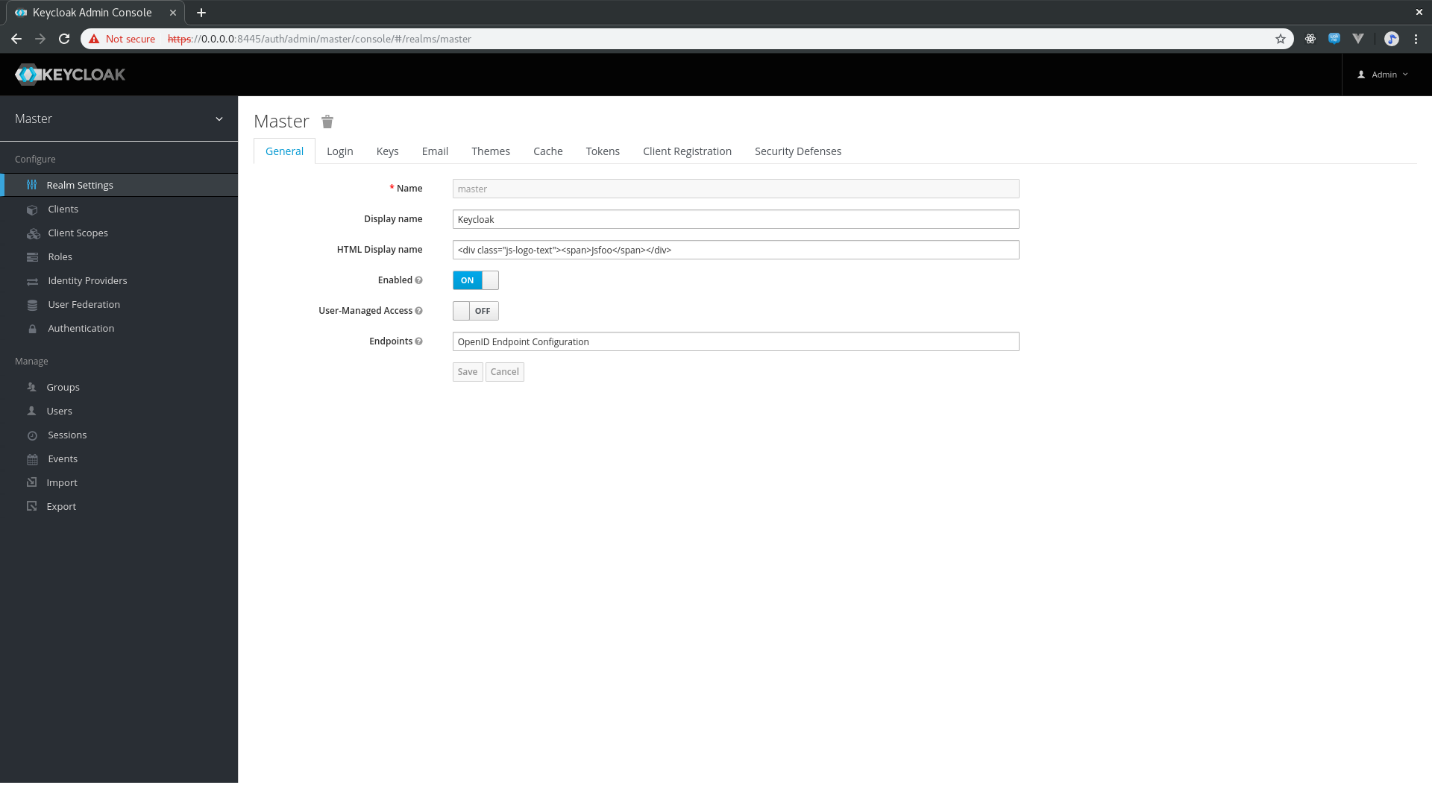
Keycloak realm giống như một không gian tên cho phép bạn quản lý tất cả siêu dữ liệu và cấu hình của mình. Chúng ta có thể có một hoặc nhiều realm tuỳ theo yêu cầu hệ thống.

Hình 1 minh hoạ, chúng ta có thể thấy các thông tin Keycloak cho phép bạn quản lý:

* Clients (per application)
* Configuration management
* Custom themes (UI)
* Events
* Federation
* LDAP or Active Directory integration
* User management (users and groups)

**Lưu ý:** Bạn có thể có một ứng dụng client chứa thông tin cấu hình cho một ứng dụng, chẳng hạn như URL, giao thức và URI direction.

Hình 2 minh hoạ, cho thấy cách Keycloak cung cấp cho bạn quyền truy cập vào tất cả thông tin này trong 1 màn hình đơn giản:



***Hình 2 minh hoạ***

(Nguồn: [Keycloak: Core concepts of open source identity and access management | Red Hat Developer](https://developers.redhat.com/blog/2019/12/11/keycloak-core-concepts-of-open-source-identity-and-access-management#launching_keycloak))

## Phân tích và thiết kế hệ thống (Keycloak)

|  |  |
| --- | --- |
|  | Keycloak cung cấp sự linh hoạt để xuất và nhập cấu hình một cách dễ dàng, sử dụng một chế độ xem duy nhất để quản lý mọi thứ. Cùng với nhau, những công nghệ này cho phép bạn tích hợp các ứng dụng front-end, di động và nguyên khối vào một kiến trúc microservice. Trong bài viết này, chúng tôi thảo luận về các khái niệm và tính năng cốt lõi của Keycloak và các cơ chế tích hợp ứng dụng của nó. Bạn sẽ tìm thấy các liên kết đến chi tiết triển khai ở gần cuối. |

Từ quan điểm thiết kế, dựa trên một tập hợp các mẫu ủy quyền được xác định rõ cung cấp các khả năng sau:

**Policy Administration Point (PAP)**

Cung cấp một bộ giao diện người dùng dựa trên Bảng điều khiển quản trị Keycloak để quản lý máy chủ tài nguyên, tài nguyên, phạm vi, quyền và chính sách. Một phần của việc này cũng được thực hiện từ xa thông qua việc sử dụng API bảo vệ.

**Policy Decision Point (PDP)**

Cung cấp điểm quyết định chính sách có thể phân phối đến nơi các yêu cầu ủy quyền được gửi và các chính sách được đánh giá tương ứng với các quyền được yêu cầu. Một phần của việc này cũng được thực hiện từ xa thông qua việc sử dụng các API Ủy quyền và Quyền lợi.

**Policy Enforcement Point (PEP)**

Cung cấp các triển khai cho các môi trường khác nhau để thực thi các quyết định ủy quyền ở phía máy chủ tài nguyên. Keycloak cung cấp một số Trình thực thi chính sách tích hợp sẵn.

**Policy Information Point (PIP)**

Dựa trên Máy chủ xác thực Keycloak, bạn có thể lấy các thuộc tính từ danh tính và môi trường thời gian chạy trong quá trình đánh giá các chính sách ủy quyền.

Tiến trình uỷ quyền (The Authorization Process)

Ba quy trình chính xác định các bước cần thiết để hiểu cách sử dụng Keycloak để kích hoạt phân quyền chi tiết cho các ứng dụng của bạn:

* Quản lý tài nguyên (Resource Management)
* Quản lý Quyền và Chính sách
* Thực thi chính sách

**Quản lý tài nguyên (Resource Management)**

**Quản lý tài nguyên (**Resource Management**)** bao gồm tất cả các bước cần thiết để xác định những gì đang được bảo vệ.

Resource Management Overview

Trước tiên, bạn cần chỉ định Keycloak bạn muốn bảo vệ cái gì, cái này thường đại diện cho một ứng dụng web hoặc một tập hợp một hoặc nhiều dịch vụ. Để biết thêm thông tin về máy chủ tài nguyên, hãy xem [Terminology](https://wjw465150.gitbooks.io/keycloak-documentation/content/authorization_services/topics/overview/terminology.html#_overview_terminology).

Máy chủ tài nguyên được quản lý bằng Bảng điều khiển quản trị Keycloak. Ở đó, bạn có thể kích hoạt bất kỳ ứng dụng khách đã đăng ký nào làm máy chủ tài nguyên và bắt đầu quản lý các tài nguyên và phạm vi mà bạn muốn bảo vệ.

Resource Server Overview

Tài nguyên có thể là một trang web, tài nguyên RESTFul, một tệp trong hệ thống tệp của bạn, EJB, v.v. Chúng có thể đại diện cho một nhóm tài nguyên (giống như một Lớp trong Java) hoặc chúng có thể đại diện cho một tài nguyên cụ thể và duy nhất.

Ví dụ: bạn có thể có tài nguyên Tài khoản ngân hàng đại diện cho tất cả các tài khoản ngân hàng và sử dụng tài nguyên đó để xác định các chính sách ủy quyền chung cho tất cả các tài khoản ngân hàng. Tuy nhiên, bạn có thể muốn xác định các chính sách cụ thể cho Tài khoản Alice (bản sao tài nguyên thuộc về khách hàng), nơi chỉ chủ sở hữu mới được phép truy cập một số thông tin hoặc thực hiện một thao tác.

Tài nguyên có thể được quản lý bằng Keycloak Administration Console hoặc [API bảo vệ](https://wjw465150.gitbooks.io/keycloak-documentation/content/authorization_services/topics/service/protection/protection-api.html#_service_protection_api). Trong trường hợp thứ hai, máy chủ tài nguyên có thể quản lý tài nguyên của họ từ xa.

Phạm vi thường đại diện cho các hành động có thể được thực hiện trên một tài nguyên, nhưng chúng không giới hạn ở đó. Bạn cũng có thể sử dụng phạm vi để đại diện cho một hoặc nhiều thuộc tính trong tài nguyên.

**Quản lý Quyền và Chính sách (Permission and Policy Management)**

Khi bạn đã xác định máy chủ tài nguyên của mình và tất cả các tài nguyên bạn muốn bảo vệ, bạn phải thiết lập các quyền và chính sách.

Quá trình này bao gồm tất cả các bước cần thiết để thực sự xác định các yêu cầu bảo mật và truy cập chi phối tài nguyên của bạn.

Permission and Policy Management Overview

Các chính sách xác định các điều kiện phải được thỏa mãn để truy cập hoặc thực hiện các hoạt động trên một thứ gì đó (tài nguyên hoặc phạm vi), nhưng chúng không ràng buộc với những gì chúng đang bảo vệ. Chúng là chung và có thể được sử dụng lại để xây dựng các quyền hoặc các chính sách phức tạp hơn.

Ví dụ: để chỉ cho phép truy cập vào một nhóm tài nguyên dành cho người dùng được cấp vai trò "Người dùng cao cấp", "bạn có thể sử dụng RBAC (Kiểm soát truy cập dựa trên vai trò).

Keycloak cung cấp một số loại chính sách tích hợp sẵn (và các nhà cung cấp chính sách tương ứng của chúng) bao gồm các cơ chế kiểm soát truy cập phổ biến nhất. Bạn thậm chí có thể tạo các chính sách dựa trên các quy tắc được viết bằng JavaScript hoặc JBoss Drools.

Khi bạn đã xác định các chính sách của mình, bạn có thể bắt đầu xác định các quyền của mình. Quyền đi đôi với tài nguyên mà chúng đang bảo vệ. Tại đây, bạn chỉ định những gì bạn muốn bảo vệ (tài nguyên hoặc phạm vi) và các chính sách phải được đáp ứng để cấp hoặc từ chối quyền.

**Thực thi chính sách (Policy Enforcement)**

**Thực thi chính sách (Policy Enforcement)** liên quan đến các bước cần thiết để thực thi các quyết định ủy quyền đối với máy chủ tài nguyên. Điều này đạt được bằng cách kích hoạt **Điểm thực thi chính sách (Policy Enforcement Point)** hoặc PEP tại máy chủ tài nguyên có khả năng giao tiếp với máy chủ ủy quyền, yêu cầu dữ liệu ủy quyền và kiểm soát quyền truy cập vào tài nguyên được bảo vệ dựa trên các quyết định và quyền do máy chủ trả về.

PEP Overview

Keycloak cung cấp một số triển khai Thực thi chính sách tích hợp sẵn mà bạn có thể sử dụng để bảo vệ các ứng dụng của mình tùy thuộc vào nền tảng chúng đang chạy.

Dịch vụ ủy quyền (Authorization Services)

Dịch vụ ủy quyền bao gồm các API RESTFul sau:

* API bảo vệ (Protection API)
* API ủy quyền (Authorization API)
* API quyền lợi (Entitlement API)

Mỗi dịch vụ này cung cấp một API cụ thể bao gồm các bước khác nhau liên quan đến quy trình ủy quyền.

**API bảo vệ (Protection API)**

API bảo vệ là một điểm cuối tuân thủ UMA cung cấp một tập hợp nhỏ các hoạt động cho máy chủ tài nguyên để giúp họ quản lý tài nguyên và phạm vi của mình. Chỉ các máy chủ tài nguyên mới được phép truy cập API này, API này cũng yêu cầu phạm vi uma\_protection.

Các hoạt động được cung cấp bởi API bảo vệ có thể được tổ chức thành hai nhóm chính:

* **Nhóm quản lý** **Resource (Resource Management)**
* **Tạo tài nguyên**
* **Xóa tài nguyên**
* **Tìm theo Id**
* **Tìm tất cả**
* **Tìm bằng bộ lọc (ví dụ: tìm kiếm theo tên, loại hoặc URI)**
* **Nhóm quản lý Permission (Permission Management)**

**API ủy quyền (Authorization API)**

API ủy quyền cũng là một điểm cuối tuân thủ UMA ([UMA-compliant](https://docs.kantarainitiative.org/uma/rec-uma-core.html)) cung cấp một hoạt động duy nhất trao đổi Access Token và  [Permission Ticket](https://wjw465150.gitbooks.io/keycloak-documentation/content/authorization_services/topics/overview/terminology.html#_overview_terminology_permission_ticket) với Mã thông báo của bên yêu cầu (RPT).

RPT chứa tất cả các quyền được cấp cho một máy khách và có thể được sử dụng để gọi một máy chủ tài nguyên để có quyền truy cập vào các tài nguyên được bảo vệ của nó.

Khi yêu cầu RPT, bạn cũng có thể cung cấp RPT đã được cấp trước đó. Trong trường hợp này, RPT kết quả sẽ bao gồm sự kết hợp của các quyền từ RPT trước đó và các quyền mới trong một phiếu cấp phép.

Authorization API Overview

Để biết thêm thông tin, hãy xem  [Authorization API](https://wjw465150.gitbooks.io/keycloak-documentation/content/authorization_services/topics/service/authorization/authorization-api.html#_service_authorization_api).

**API quyền lợi (Entitlement API)**

API Entitlement cung cấp một giao thức một chân để phát hành RPT. Không giống như API ủy quyền, API quyền lợi chỉ mong đợi một mã thông báo truy cập.

Từ API này, bạn có thể nhận được tất cả các quyền hoặc quyền cho người dùng (dựa trên các tài nguyên được quản lý bởi một máy chủ tài nguyên nhất định) hoặc chỉ các quyền cho một tập hợp một hoặc nhiều tài nguyên.

Entitlement API Overview

## Hiện thực và đánh giá hệ thống

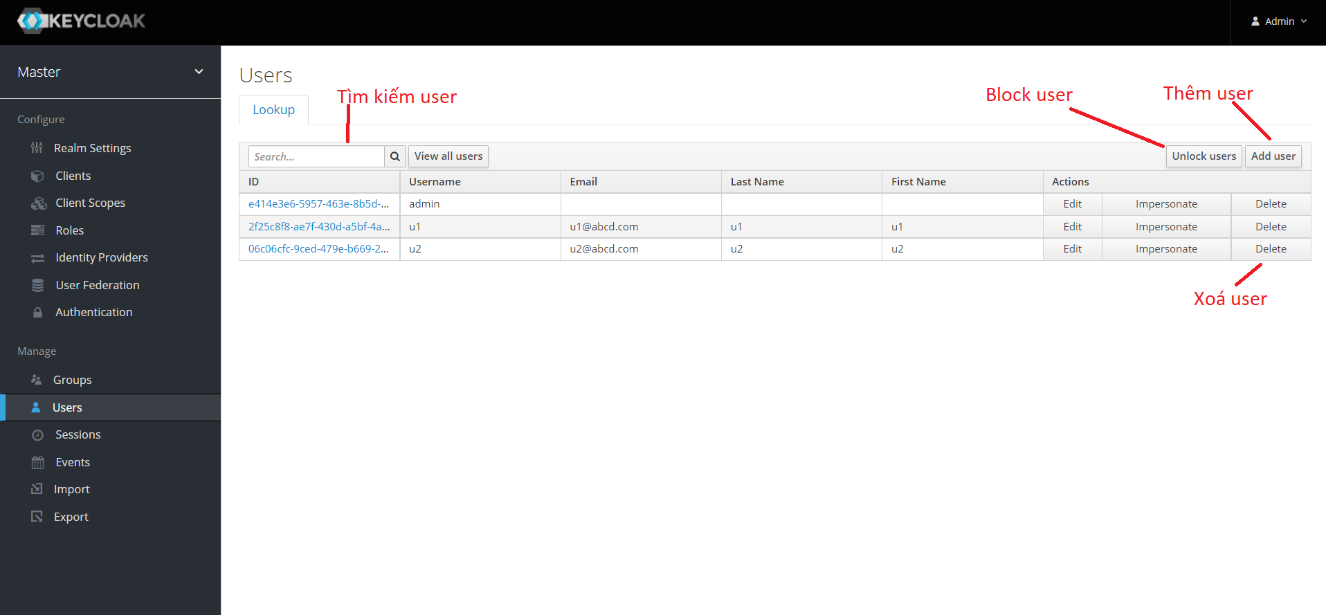
|  |  |
| --- | --- |
|  | Describe the high level requirements for the project. For example: |

1. **Các chức năng quản lý người dùng: tạo, cập nhật, xóa, tìm kiếm thông tin định danh của  
   người dùng với các dịch vụ thư mục như OpenLDAP.**

Keycloak có hệ thống quản lý người dùng của riêng nó, nó ko sử dụng dịch vụ thư mục như OpenLdap. Tuy nhiên nhiều công ty có cơ sở dữ liệu người dùng hiện có chứa thông tin về người dùng dựa trên dịch vụ thư mục (ví dụ OpenLDAP). Trong nhiều trường hợp, không thể chuyển từ các lưu trữ hiện có đó sang triển khai Keycloak thuần túy. Keycloak có chức năng liên kết các cơ sở dữ liệu người dùng bên ngoài (ví dụ OpenLDAP).

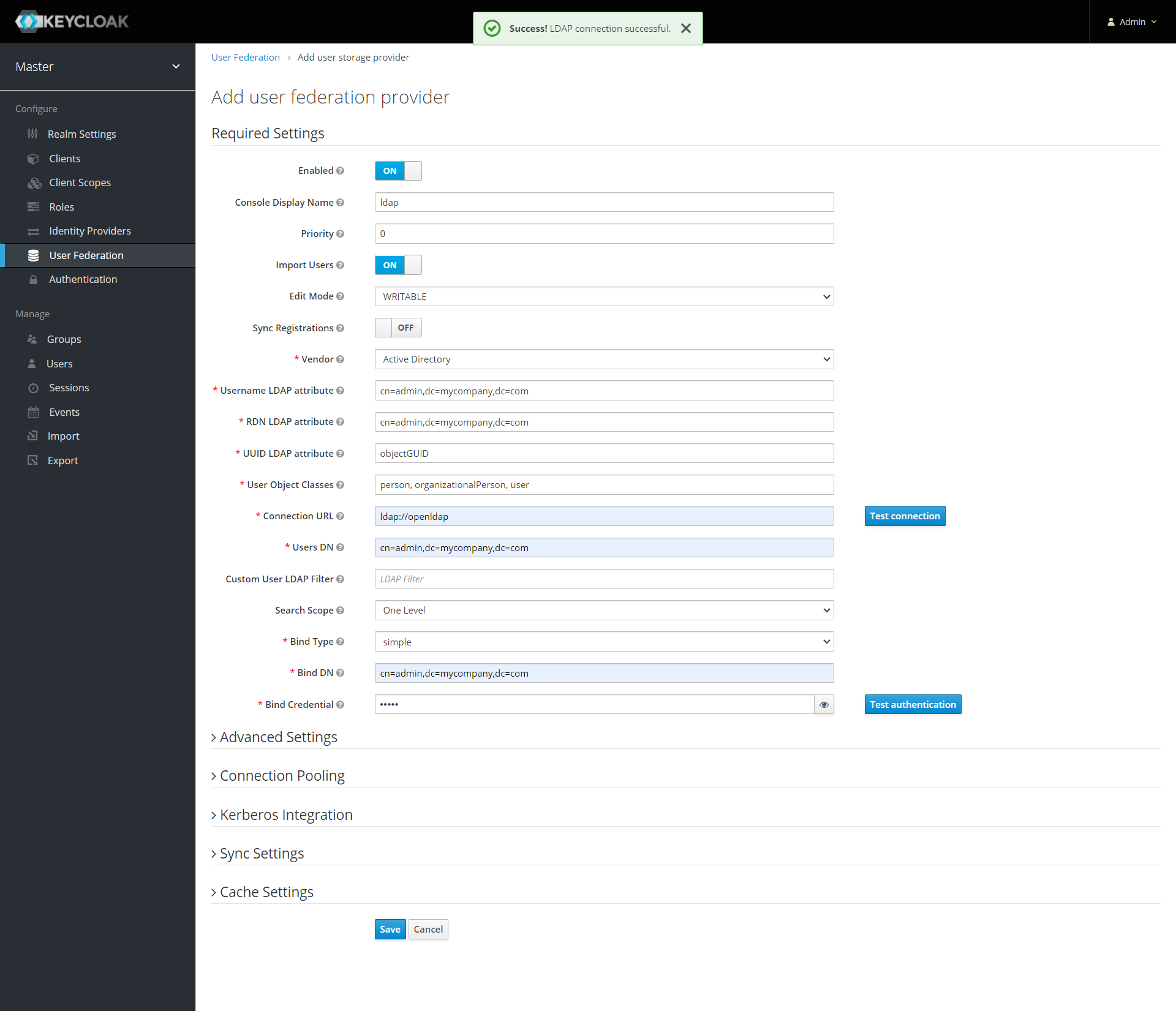
Phần này sẽ mô tả sơ lược về chức năng quản lý người dùng của Keycloak, chức năng liên kết người dùng với OpenLDAP.

**Chức năng quản lý người dùng của KeyCloak:**

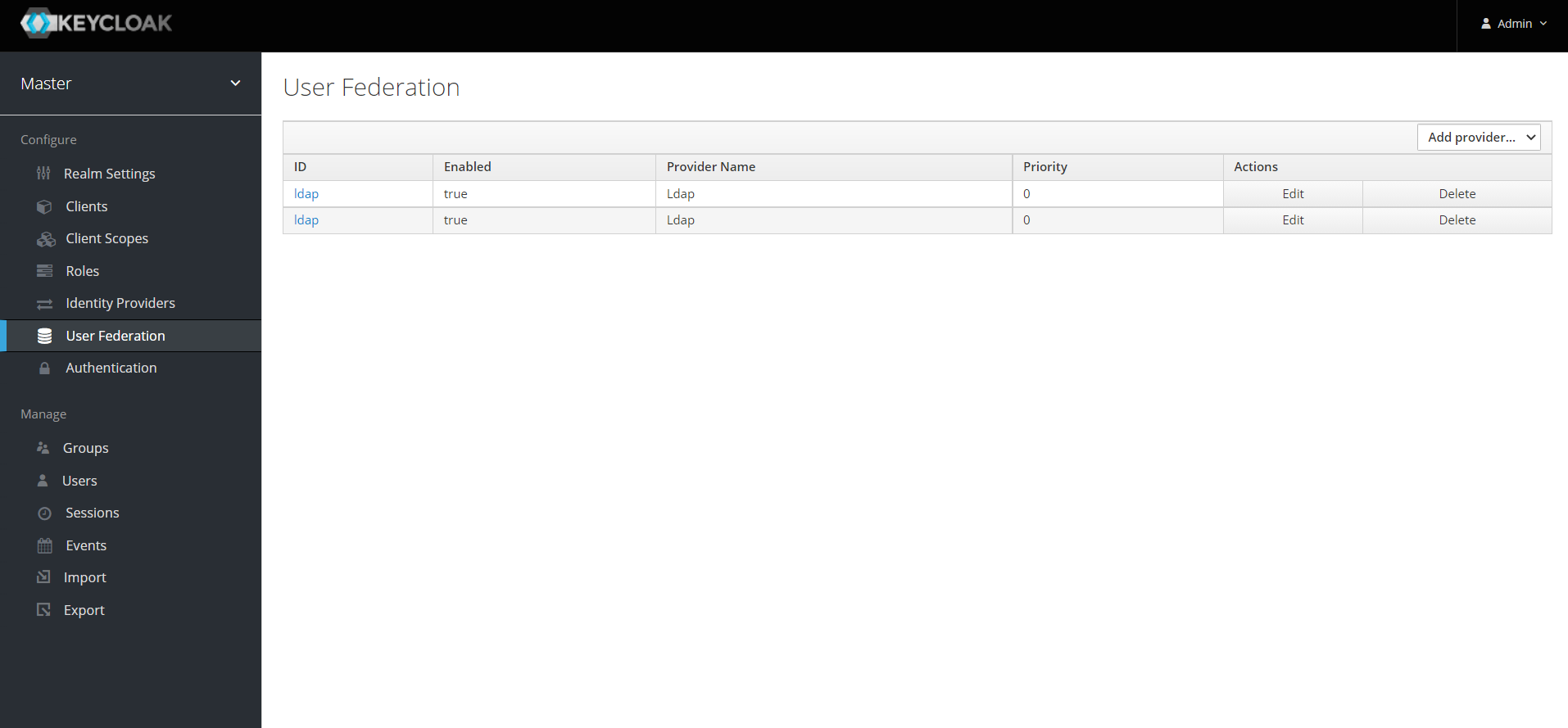


**Liên kết người dùng với OpenLDAP:**

Từ menu, chọn “User Federation”, ở page content, chọn add provider, chọn ldap, điền thông tin OpenLdap server, test connection, save.



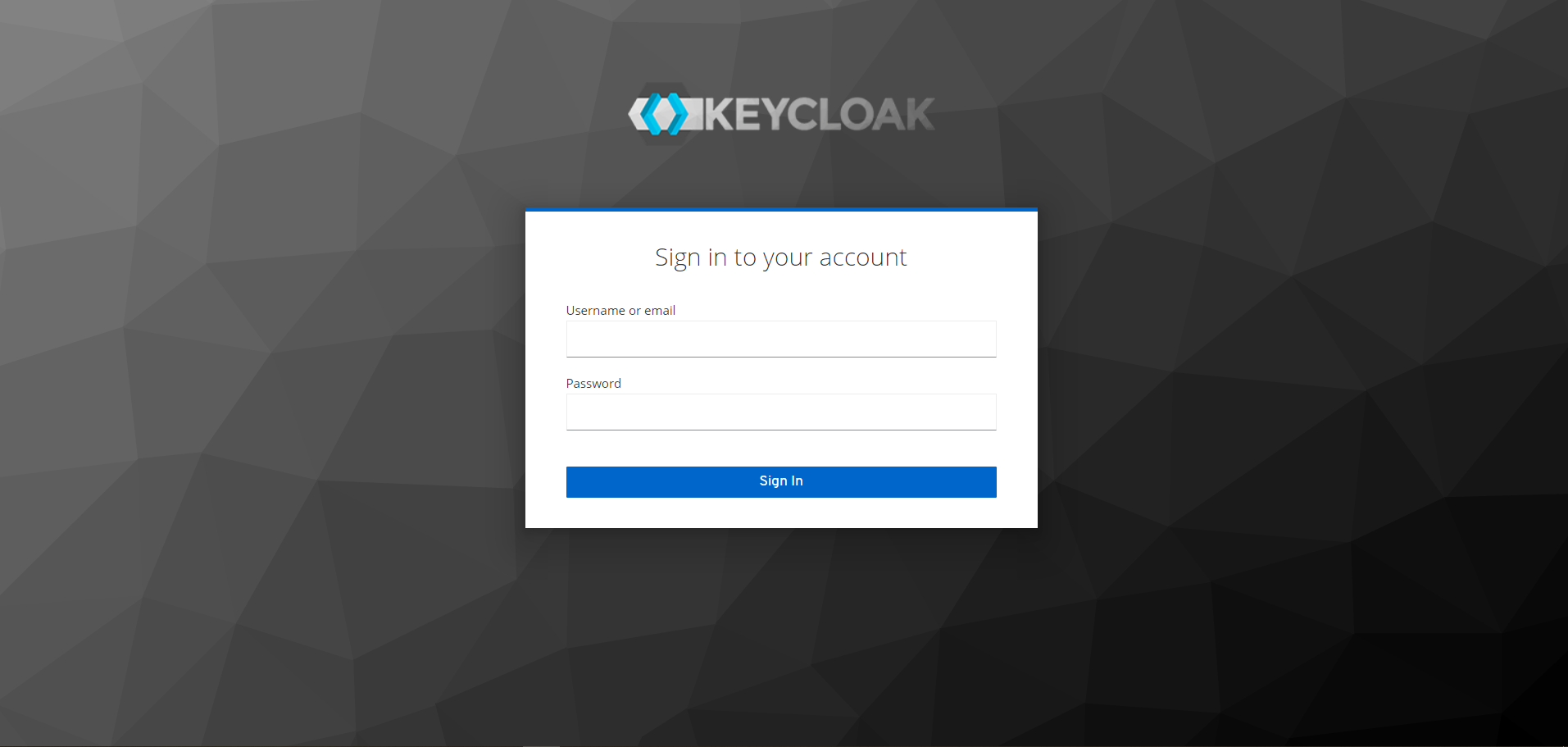
Provider đã được tạo.



1. **Các chức năng xác thực với các phương pháp: username và password, smart card và PIN(Personal Identification Number).**

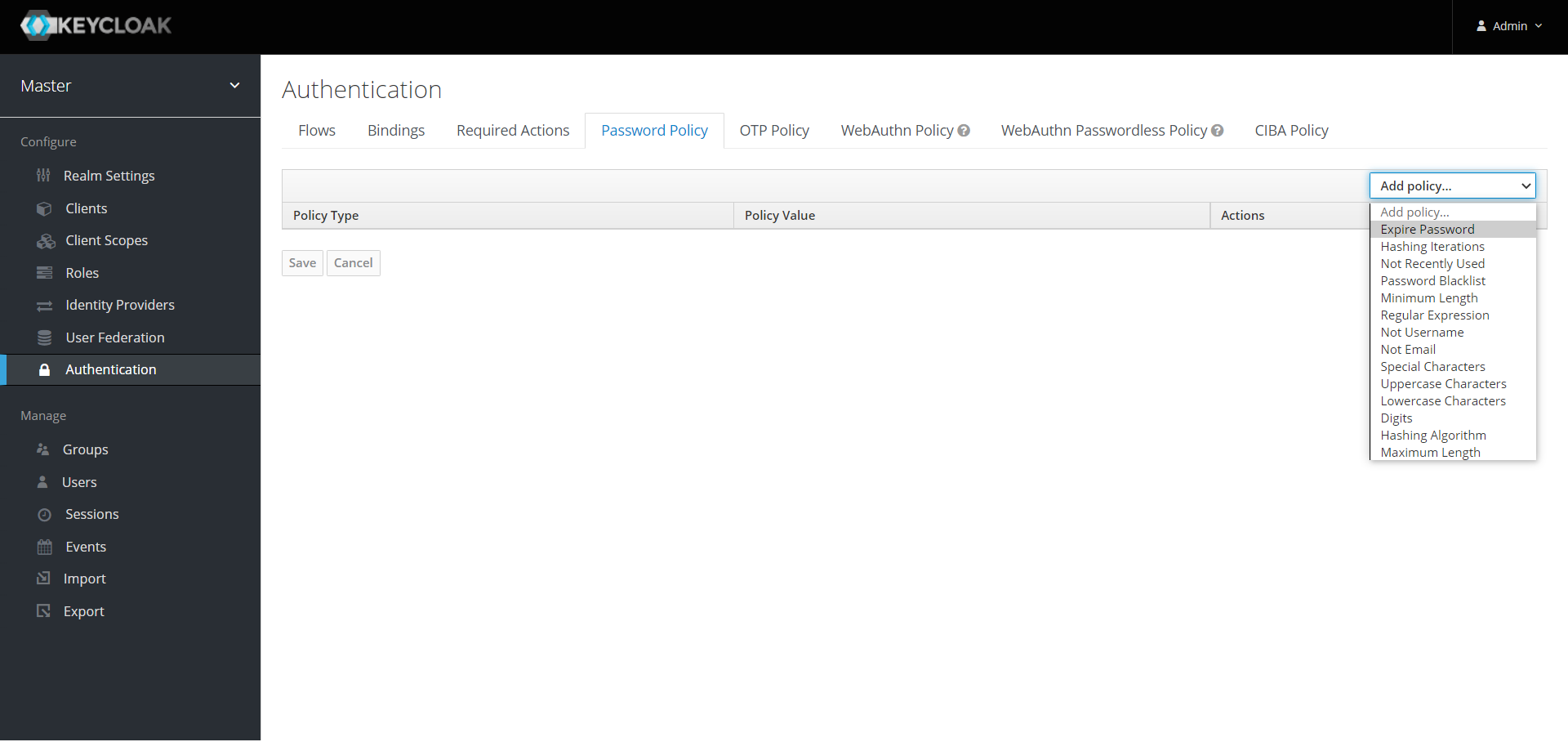
Keycloak mặc định chỉ hỗ trợ login với username và password. Để xác thực bằng các phương pháp như smart cart, PIN, vân tay, … Cần phải viết plugin cho KeyCloak.   
Phần này chỉ giới thiệu chức năng xác thực với phương pháp username và password, và cách quản lý chính sách password của Keycloak.

Keycloak admin có khả năng tạo user, và reset password của user. Bên dưới là màn hình login của Keycloak:



**Password policy của Keycloak**

Bên dưới là chức năng quản lý “password policy” của Keycloak.



Danh sách password policy của Keycloak

|  |  |
| --- | --- |
| Password policy | Chú thích |
| Expire Password | Thời gian hết hạn của password |
| Not Recently Used | Chính sách này lưu lịch sử của các mật khẩu trước đó. Số lượng mật khẩu cũ được lưu trữ có thể cấu hình được. Khi người dùng thay đổi mật khẩu, họ không thể sử dụng bất kỳ mật khẩu nào được lưu trữ. |
| Minimum Length | Chiều dài tối thiểu |
| Not Username | Khi được đặt, mật khẩu không được phép giống với tên người dùng. |
| Special Characters | Số ký tự đặc biệt như '?! #% $' Bắt buộc phải có trong chuỗi mật khẩu. |
| Uppercase Characters | Số lượng các chữ cái viết hoa bắt buộc phải có trong chuỗi mật khẩu. |
| Hashing Iterations | Giá trị này chỉ định số lần mật khẩu sẽ được băm trước khi nó được lưu trữ hoặc xác minh. Giá trị mặc định là 20.000. Quá trình băm này được thực hiện trong trường hợp hiếm hoi mà một hacker có thể truy cập vào cơ sở dữ liệu mật khẩu của bạn. Khi họ có quyền truy cập vào cơ sở dữ liệu, họ có thể thiết kế ngược lại mật khẩu của người dùng. Giá trị khuyến nghị của ngành cho thông số này thay đổi hàng năm khi sức mạnh của CPU được cải thiện. Giá trị lặp lại băm cao hơn sẽ tốn nhiều năng lượng hơn của CPU để băm và có thể ảnh hưởng đến hiệu suất. Bạn sẽ phải cân nhắc xem điều gì quan trọng hơn đối với bạn. Hiệu suất hoặc bảo vệ các cửa hàng mật khẩu của bạn. Có thể có nhiều cách hiệu quả hơn về chi phí để bảo vệ kho mật khẩu của bạn. |
| Password Blacklist | Chính sách này kiểm tra xem mật khẩu đã cho có nằm trong tệp danh sách đen hay không, tệp này có thể là một tệp rất lớn. Danh sách đen mật khẩu là các tệp văn bản thuần túy UTF-8 với phần cuối là dòng Unix trong đó mỗi dòng đại diện cho một mật khẩu nằm trong danh sách cấm. Tên tập tin của tập tin danh sách đen phải được cung cấp như là giá trị chính sách mật khẩu |
| Regular Expression | Xác định một hoặc nhiều mẫu biểu thức chính quy (được định nghĩa trong **java.util.regex.Pattern**) mà mật khẩu phải khớp. |
| Not Email | Khi được đặt, mật khẩu không được phép giống với địa chỉ email. |
| Lowercase Characters | Số lượng các chữ cái viết thường bắt buộc phải có trong chuỗi mật khẩu. |
| Digits | Số chữ số bắt buộc phải có trong chuỗi mật khẩu. |
| Hashing Algorithm | Mật khẩu không được lưu trữ dưới dạng văn bản rõ ràng. Thay vào đó, chúng được băm bằng cách sử dụng các thuật toán băm tiêu chuẩn trước khi chúng được lưu trữ hoặc xác thực. Thuật toán mặc định và tích hợp duy nhất có sẵn là PBKDF2. Xem [Hướng dẫn dành cho nhà phát triển](https://www.keycloak.org/docs/4.8/server_development/) máy chủ về cách cắm thuật toán của riêng bạn. Lưu ý rằng nếu bạn thay đổi thuật toán, hàm băm mật khẩu sẽ không thay đổi trong bộ nhớ cho đến lần người dùng đăng nhập tiếp theo. |
| Maximum Length | Chiều dài tối đa |

1. **Các chức năng quản trị thông qua Web cho việc quản lý người dùng và nhật ký.**
2. **Cài đặt hệ thống**

## Kết luận

|  |  |
| --- | --- |
|  | List agencies, stakeholders or divisions which will be impacted by this project and describe how they will be affected by the project. |

# Danh sách thành viên và phân công

We approve the project as described above, and authorize the team to proceed.

|  |  |  |
| --- | --- | --- |
| STT | MSSV | Name |
| 1 | 2033009 | Lưu Quốc Bình |
| 2 |  |  |
|  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |
| Approved By |  |  | Date |  | Approved By |  |  | Date |