KEYCLOAK

Vre

Configure

- Realm Settings
- Clients
- Client Scopes
- Roles
- Identity Providers
- User Federation
- Authentication

Manage

- Groups
- Users
- Sessions
- Events
- Import
- Export

Minio 🗑

| Settings | Credentials | Roles | Client Scopes @ | Mappers @ | Scope @ | Revocation | Sessions @ | Offline Access @ | Clustering | Installation @ |

Field	Value
Client ID @	minio
Name @	
Description @	
Enabled @	ON
Consent Required @	OFF
Login Theme @	
Client Protocol @	openid-connect
Access Type @	confidential
Standard Flow Enabled @	ON
Implicit Flow Enabled @	ON
Direct Access Grants Enabled @	ON
Service Accounts Enabled @	OFF
Authorization Enabled @	OFF
Root URL @	
* Valid Redirect URIs @	*
Base URL @	
Admin URL @	

KEYCLOAK

Vre

Configure

- Realm Settings
- **Clients**
- Client Scopes
- Roles
- Identity Providers
- User Federation
- Authentication

Manage

- Groups
- Users
- Sessions
- Events
- Import
- Export

Minio 🗑

Protocol ❓	openid-connect
ID	c83ab853-61ab-46e5-b1b6-
Name ❓	minio
Mapper Type ❓	User Attribute
User Attribute ❓	policy
Token Claim Name ❓	policy
Claim JSON Type ❓	String
Add to ID token ❓	ON
Add to access token ❓	ON
Add to userinfo ❓	ON
Multivalued ❓	OFF
Aggregate attribute values ❓	OFF

Save Cancel

Vre ⌄

Configure

⚙ Realm Settings

📦 **Clients**

👥 Client Scopes

🗒 Roles

⇄ Identity Providers

🗄 User Federation

🔒 Authentication

Manage

👥 Groups

👤 Users

🕐 Sessions

🗓 Events

⬐ Import

↗ Export

Minio_policy_script_mapper 🗑

Protocol ❔	openid-connect
ID	2348631d-de67-48f9-8a4f█████████
Name ❔	minio_policy_script_mapper
Mapper Type ❔	Script Mapper

Script ❔

```
 1  /**
 2   * Available variables:
 3   * user - the current user
 4   * realm - the current realm
 5   * token - the current token
 6   * userSession - the current userSession
 7   * keycloakSession - the current keycloakSession
 8   */
 9
10
11  //insert your code here...
12
13  var ArrayList = Java.type("java.util.ArrayList");
14  var roles = new ArrayList();
15  // roles.add("readwrite")
16
17  rr = user.getRealmRoleMappings()
18  for each (var r in rr){
19      role_name = r.getName()
20      if(role_name == "platform-admin") roles.add("readwrite")
21      else roles.add(role_name)
22
23  }
24
25
26  // give the system admin with the read/write policy
27  if(user.getUsername() =="admin") roles.add("readwrite")
28
29  // also attach the username as policy since the dataset
30  // is checked per user
31  roles.add(user.getUsername())
32
33  token.setOtherClaims("policy", roles)
34  // token.setOtherClaims("policy", "readwrite")
```