Configurez ODBC sur ISE 2.3 avec la base de données d'Oracle

Contenu

Introduction

Ce document décrit comment configurer le Cisco Identity Services Engine (ISE) avec la base de données d'Oracle pour l'authentification ISE utilisant la Connectivité de base de données ouverte (ODBC).

L'authentification de la Connectivité de base de données ouverte (ODBC) exige d'ISE de pouvoir chercher un mot de passe utilisateur de texte brut. Le mot de passe peut être chiffré dans la base de données, mais doit être déchiffré par la procédure stockée.

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Logiciel Cisco Identity Services Engine 2.3
- Base de données et concepts ODBC

- Oracle

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Cisco Identity Services Engine 2.3.0.298
- Centos 7
- Base de données 12.2.0.1.0 d'Oracle
- Développeur 4.1.5 d'Oracle SQL

Configurez

> **Note**: Traitez les procédures SQL présentées dans ce document comme exemples. Ce n'est pas un fonctionnaire et une manière recommandée de configuration de DB d'Oracle. Assurez-vous que vous comprenez le résultat et l'incidence de chaque requête SQL que vous commettez.

Étape 1. Configuration de base d'Oracle

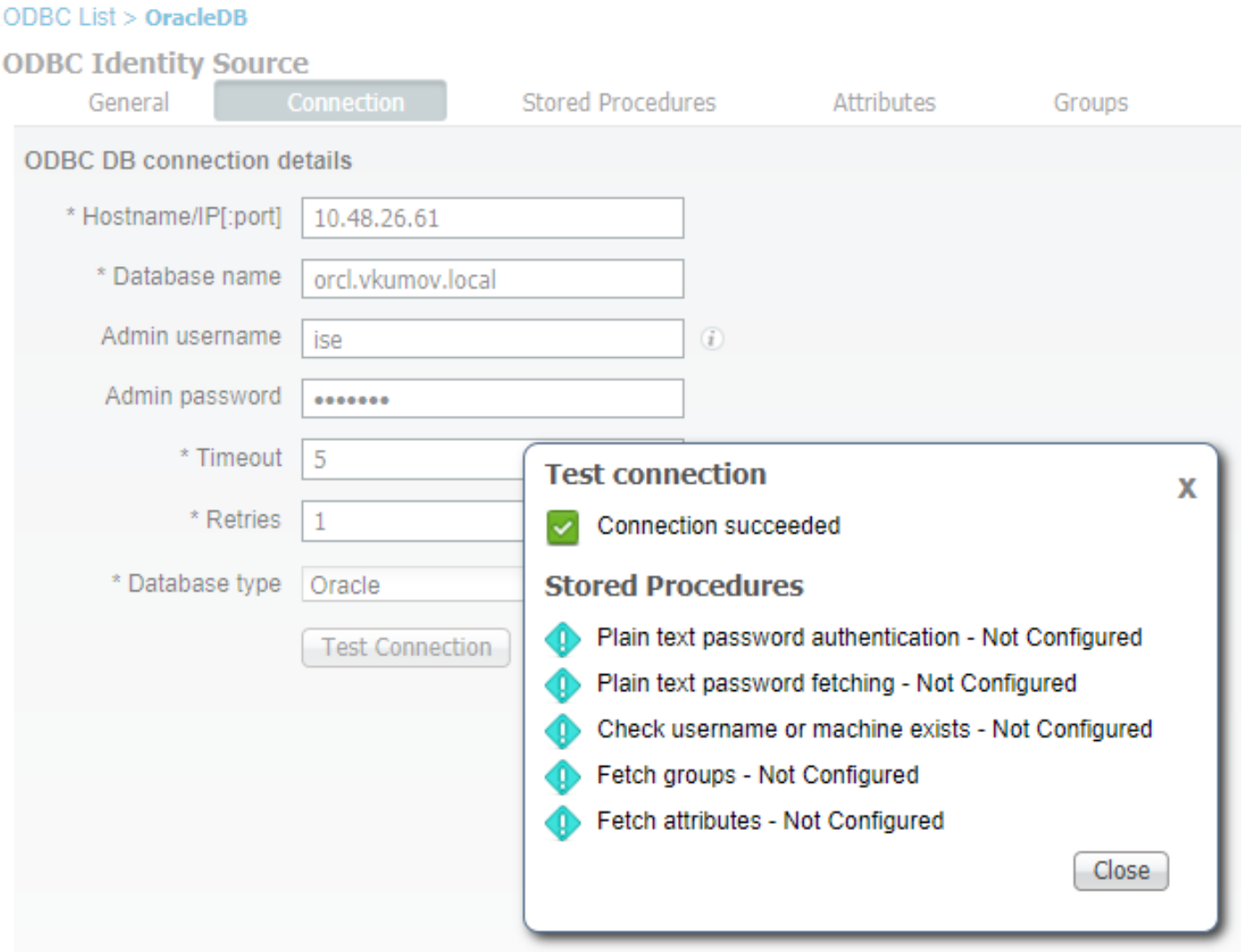
Dans cet exemple Oracle a été configuré avec des paramètres suivants :

- Nom de DB : **ORCL**
- Nom de service : **orcl.vkumov.local**
- Port : **1521** (par défaut)
- Créé expliquez ISE avec l'**ise de** nom d'utilisateur

Configurez votre base de données d'Oracle avant de commencer plus loin.

Étape 2. Configuration de base ISE

Créez une source d'identité ODBC à la *gestion > source extérieure d'identité > ODBC* et connexion de test :



Note: ISE se connecte à Oracle utilisant le nom de service, par conséquent [le champ de nom de la base de données] devrait être rempli de nom de service qui existe à Oracle, nom pas SID (ou de DB). En raison des points de la bogue CSCvf06497 (.) ne peut pas être utilisé dans [le domaine de nom de la base de données]. Cette bogue est réparée dans ISE 2.3.

Étape 3. Configurez l'authentification de l'utilisateur

L'authentification ISE à ODBC utilise le stored procedures. Il est possible au type de sélection de procédures. Dans cet exemple nous utilisons des recordsets en tant que retour.

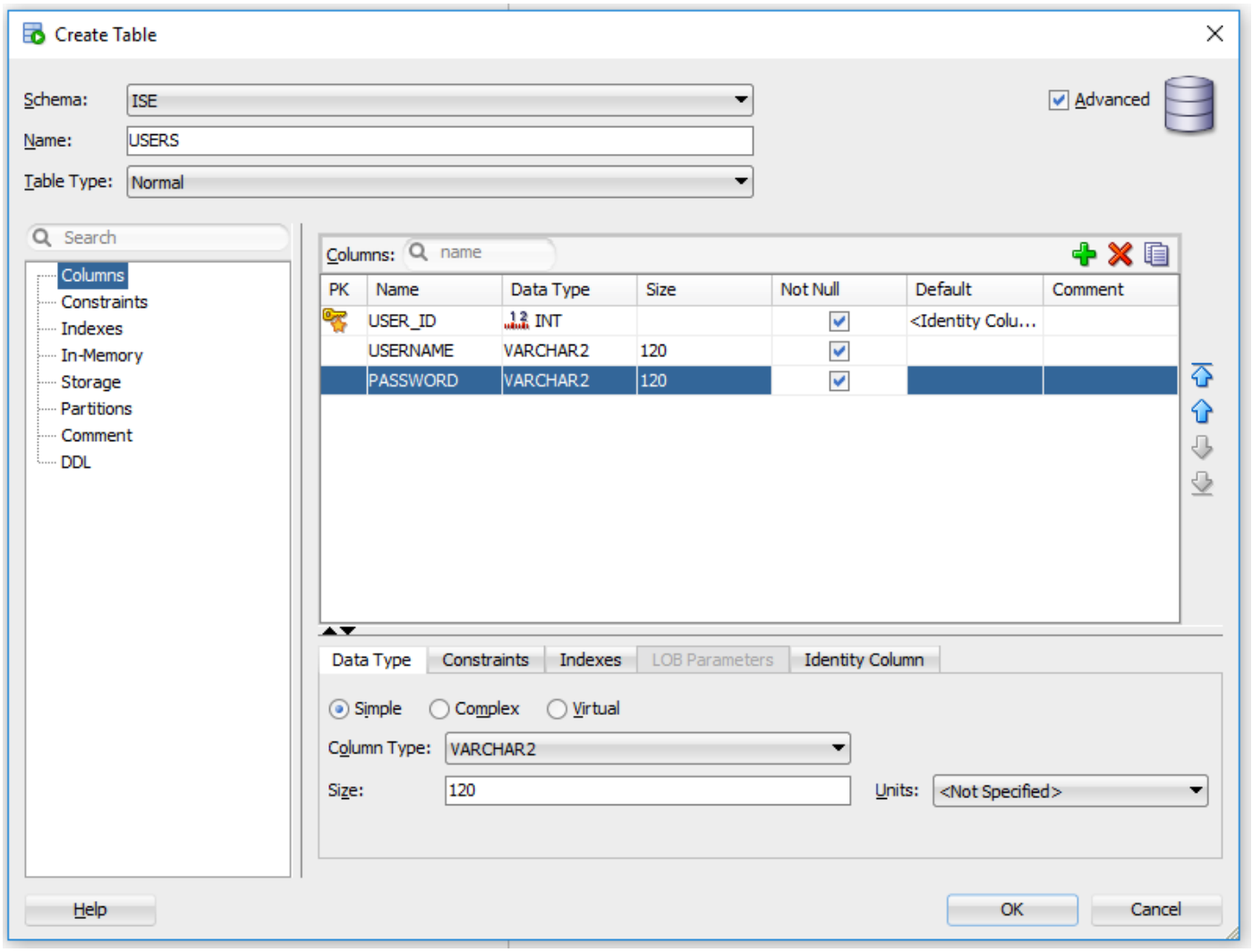
Pour d'autres procédures, référez-vous au guide de l'administrateur de Logiciel Cisco Identity Services Engine, version 2.3

Conseil : Il est possible de renvoyer des paramètres Désignés au lieu du resultSet. C'est juste un type différent de sortie, fonctionnalité est identique.

1. Créez la table avec les qualifications des utilisateurs. Assurez-vous que vous avez placé les configurations d'identité sur la **clé primaire.**

```
--------------------------------------------------------
--  DDL for Table USERS
--------------------------------------------------------

  CREATE TABLE "ISE"."USERS"
   ("USER_ID" NUMBER(*,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE
9999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER  NOCYCLE  NOKEEP
NOSCALE ,
"USERNAME" VARCHAR2(120 BYTE),
"PASSWORD" VARCHAR2(120 BYTE)
   ) SEGMENT CREATION IMMEDIATE
  PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS" ;
--------------------------------------------------------
--  DDL for Index USERS_PK
--------------------------------------------------------

  CREATE UNIQUE INDEX "ISE"."USERS_PK" ON "ISE"."USERS" ("USER_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS" ;
--------------------------------------------------------
--  Constraints for Table USERS
--------------------------------------------------------

  ALTER TABLE "ISE"."USERS" MODIFY ("USER_ID" NOT NULL ENABLE);
  ALTER TABLE "ISE"."USERS" MODIFY ("USERNAME" NOT NULL ENABLE);
  ALTER TABLE "ISE"."USERS" MODIFY ("PASSWORD" NOT NULL ENABLE);
  ALTER TABLE "ISE"."USERS" ADD CONSTRAINT "USERS_PK" PRIMARY KEY ("USER_ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS"  ENABLE;
```

Ou du GUI de développeur SQL :



2. Ajoutez les utilisateurs

```
INSERT INTO "ISE"."USERS" (USERNAME, PASSWORD) VALUES ('alice', 'password1')
INSERT INTO "ISE"."USERS" (USERNAME, PASSWORD) VALUES ('bob', 'password1')
INSERT INTO "ISE"."USERS" (USERNAME, PASSWORD) VALUES ('admin', 'password1')
```

3. Créez une procédure pour l'authentification de mot de passe de texte brut (utilisée pour la méthode intérieure PAP, EAP-GTC, TACACS)

```
create or replace function ISEAUTH_R
(
  ise_username IN VARCHAR2,
  ise_userpassword IN VARCHAR2
) return sys_refcursor AS
BEGIN
  declare
    c integer;
    resultSet SYS_REFCURSOR;
  begin
    select count(*) into c from USERS where USERS.USERNAME = ise_username and USERS.PASSWORD =
ise_userpassword;
    if c > 0 then
      open resultSet for select 0 as code, 11, 'good user', 'no error' from dual;
    ELSE
      open resultSet for select 3, 0, 'odbc','ODBC Authen Error' from dual;
    END IF;
```

```
    return resultSet;
  end;
END ISEAUTH_R;
```

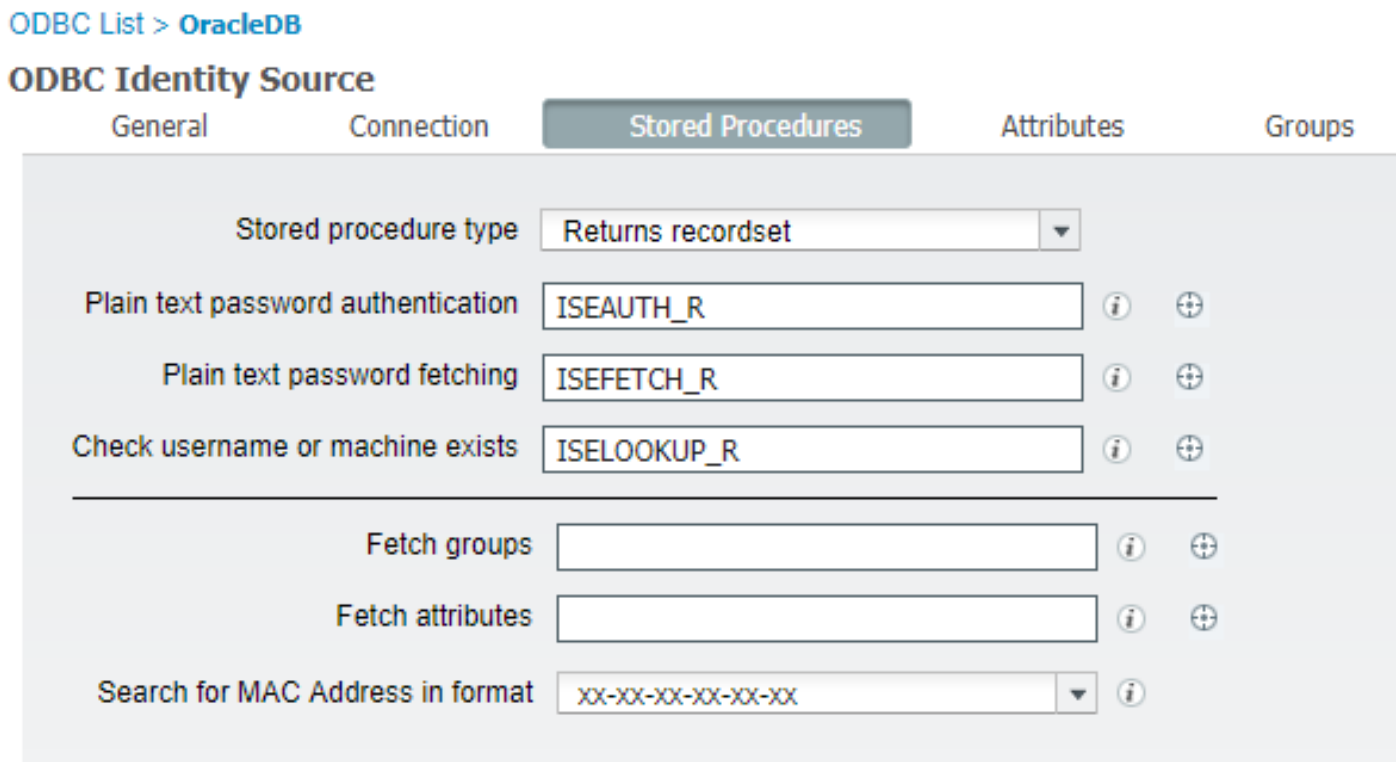
4. Créez une procédure pour chercher de mot de passe de texte brut (utilisé pour le CHAP, MSCHAPv1/v2, EAP-MD5, LEAP, méthode EAP-MSCHAPv2 intérieure, TACACS)

```
create or replace function ISEFETCH_R
(
  ise_username IN VARCHAR2
) return sys_refcursor AS
BEGIN
  declare
    c integer;
    resultSet SYS_REFCURSOR;
  begin
    select count(*) into c from USERS where USERS.USERNAME = ise_username;
    if c > 0 then
      open resultSet for select 0, 11, 'good user', 'no error', password from USERS where
USERS.USERNAME = ise_username;
      DBMS_OUTPUT.PUT_LINE('found');
    ELSE
      open resultSet for select 3, 0, 'odbc','ODBC Authen Error' from dual;
      DBMS_OUTPUT.PUT_LINE('not found');
    END IF;
    return resultSet;
  end;
END;
```

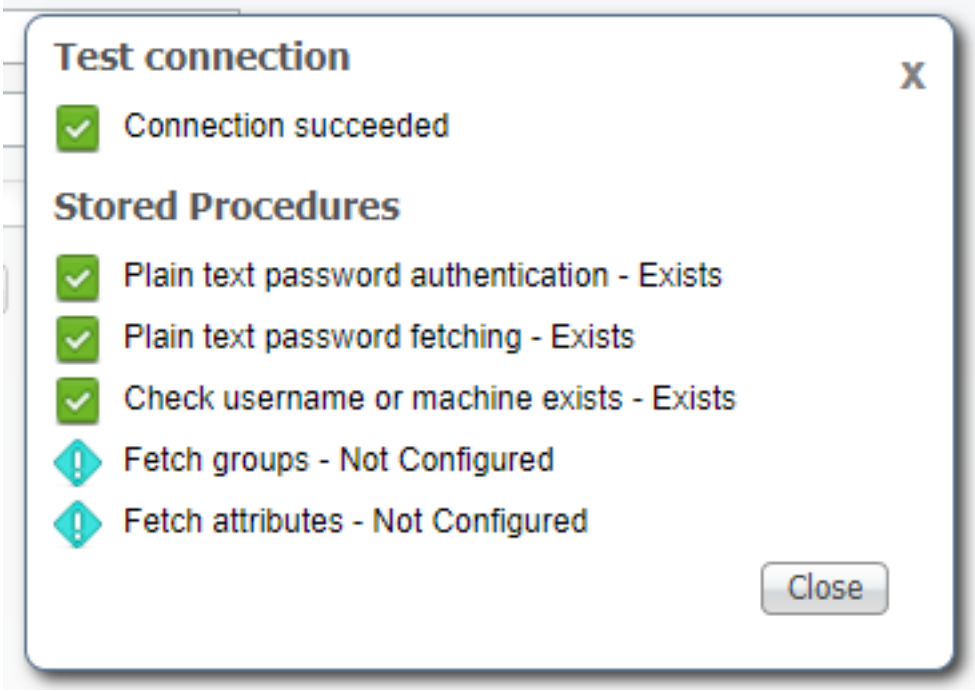
5. Créez une procédure pour le nom d'utilisateur de contrôle ou l'ordinateur existe (utilisé pour le MAB, rapide rebranchez du PEAP, de l'EAP-FAST et de l'EAP-TTLS)

```
create or replace function ISELOOKUP_R
(
  ise_username IN VARCHAR2
) return sys_refcursor AS
BEGIN
  declare
    c integer;
    resultSet SYS_REFCURSOR;
  begin
    select count(*) into c from USERS where USERS.USERNAME = ise_username;
    if c > 0 then
      open resultSet for select 0, 11, 'good user', 'no error' from USERS where USERS.USERNAME =
ise_username;
    ELSE
      open resultSet for select 3, 0, 'odbc','ODBC Authen Error' from dual;
    END IF;
    return resultSet;
  end;
END;
```

6. Configurez les procédures sur ISE et les sauvegardez



7. Retournez à l'onglet Connection et cliquez sur le bouton de connexion de test



Étape 4. Configurez la récupération de groupe

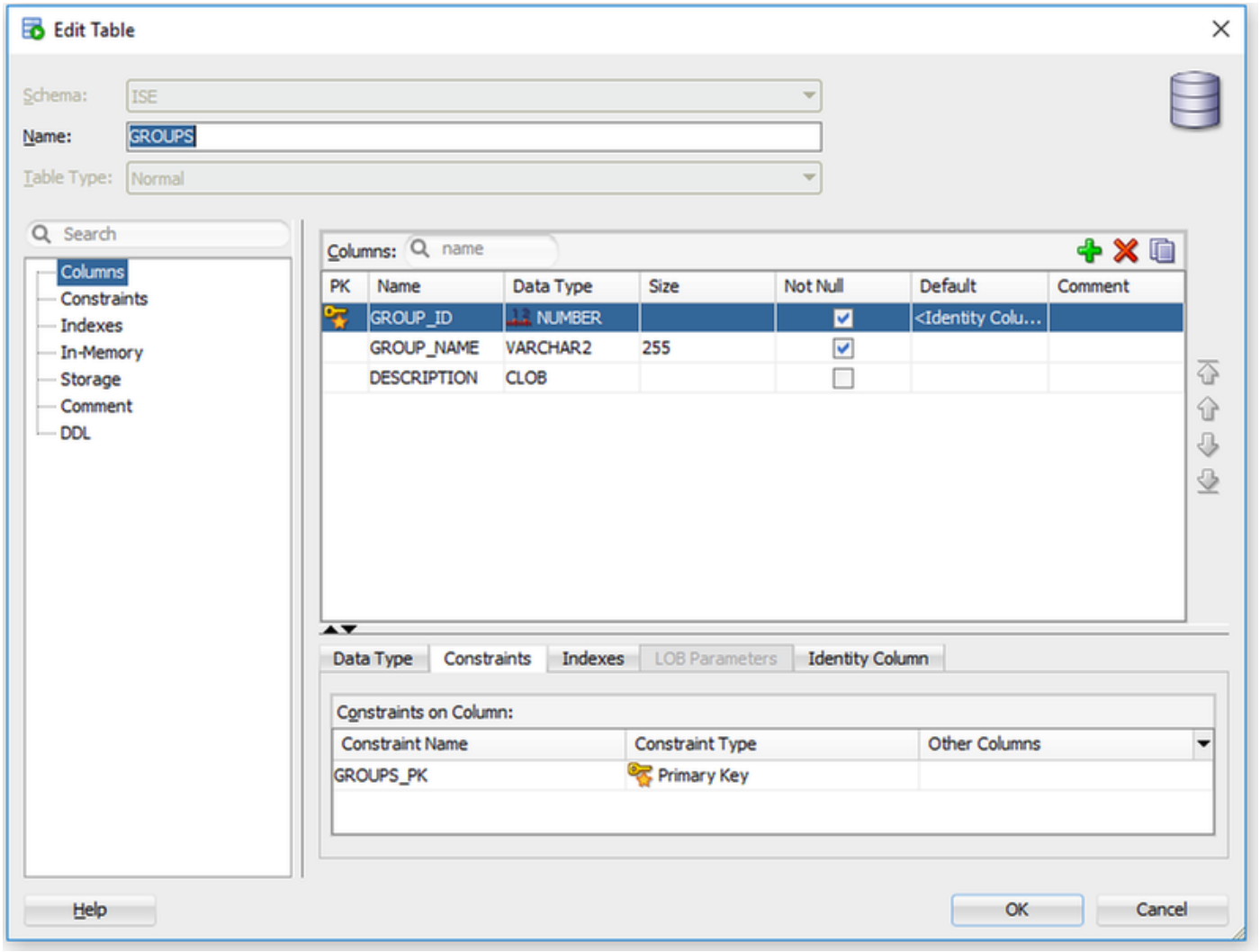
1. Créez les tables contenant des groupes d'utilisateurs et des autres utilisées pour le mappage multiple

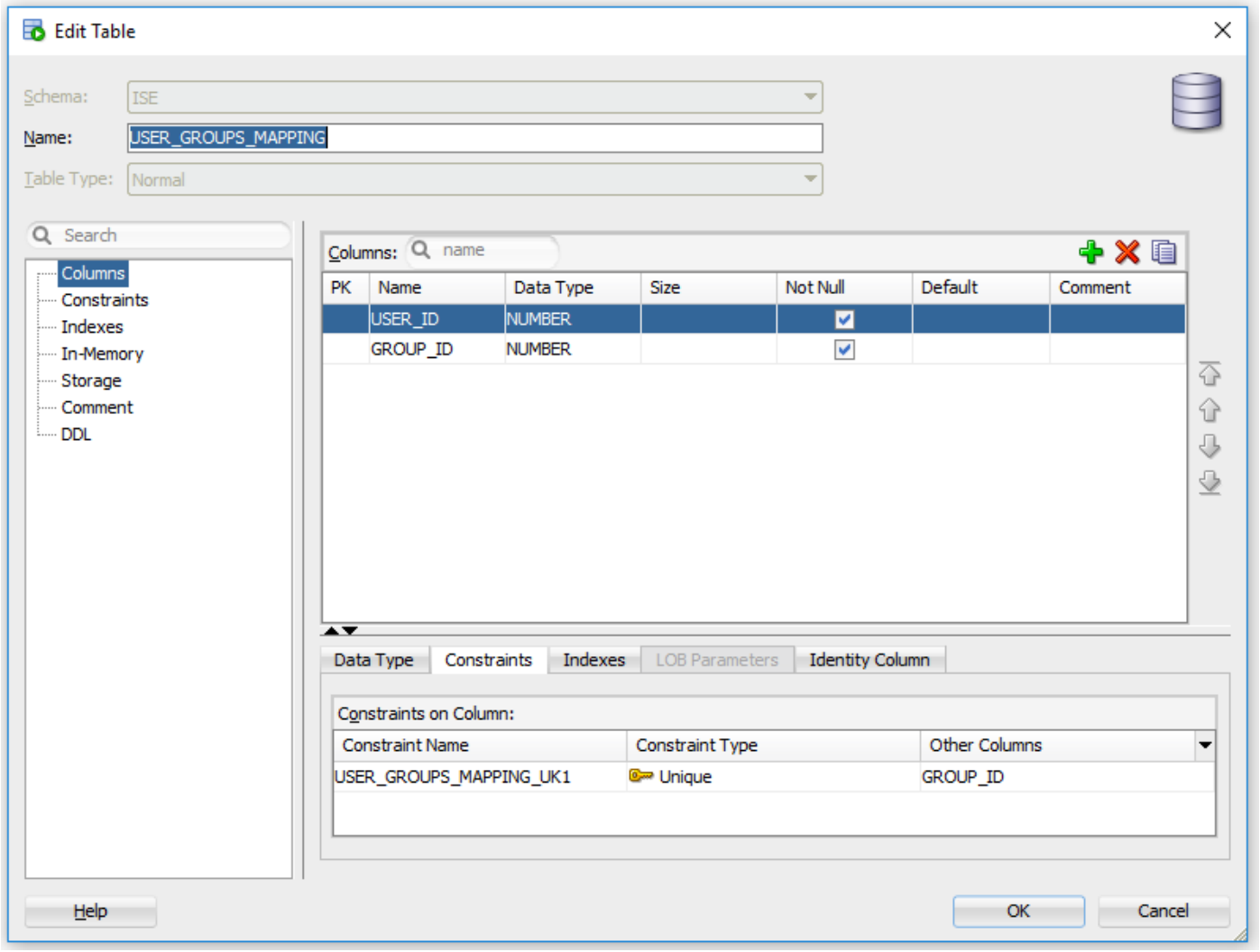
```
--------------------------------------------------------
--  DDL for Table GROUPS
--------------------------------------------------------

  CREATE TABLE "ISE"."GROUPS"
   ("GROUP_ID" NUMBER(*,0) GENERATED ALWAYS AS IDENTITY MINVALUE 1 MAXVALUE
```

```
99999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER  NOCYCLE  NOKEEP
NOSCALE ,
"GROUP_NAME" VARCHAR2(255 BYTE),
"DESCRIPTION" CLOB
   ) SEGMENT CREATION IMMEDIATE
  PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS"
 LOB ("DESCRIPTION") STORE AS SECUREFILE (
  TABLESPACE "USERS" ENABLE STORAGE IN ROW CHUNK 8192
  NOCACHE LOGGING  NOCOMPRESS  KEEP_DUPLICATES
  STORAGE(INITIAL 106496 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)) ;
--------------------------------------------------------
--  DDL for Table USER_GROUPS_MAPPING
--------------------------------------------------------

  CREATE TABLE "ISE"."USER_GROUPS_MAPPING"
   ("USER_ID" NUMBER(*,0),
"GROUP_ID" NUMBER(*,0)
   ) SEGMENT CREATION IMMEDIATE
  PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS" ;
--------------------------------------------------------
--  DDL for Index GROUPS_PK
--------------------------------------------------------

  CREATE UNIQUE INDEX "ISE"."GROUPS_PK" ON "ISE"."GROUPS" ("GROUP_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS" ;
--------------------------------------------------------
--  DDL for Index USER_GROUPS_MAPPING_UK1
--------------------------------------------------------

  CREATE UNIQUE INDEX "ISE"."USER_GROUPS_MAPPING_UK1" ON "ISE"."USER_GROUPS_MAPPING" ("USER_ID",
"GROUP_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS" ;
--------------------------------------------------------
--  Constraints for Table GROUPS
--------------------------------------------------------

  ALTER TABLE "ISE"."GROUPS" MODIFY ("GROUP_ID" NOT NULL ENABLE);
  ALTER TABLE "ISE"."GROUPS" MODIFY ("GROUP_NAME" NOT NULL ENABLE);
  ALTER TABLE "ISE"."GROUPS" ADD CONSTRAINT "GROUPS_PK" PRIMARY KEY ("GROUP_ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS"  ENABLE;
```

```
--------------------------------------------------------
--  Constraints for Table USER_GROUPS_MAPPING
--------------------------------------------------------

  ALTER TABLE "ISE"."USER_GROUPS_MAPPING" MODIFY ("USER_ID" NOT NULL ENABLE);
  ALTER TABLE "ISE"."USER_GROUPS_MAPPING" MODIFY ("GROUP_ID" NOT NULL ENABLE);
  ALTER TABLE "ISE"."USER_GROUPS_MAPPING" ADD CONSTRAINT "USER_GROUPS_MAPPING_UK1" UNIQUE
("USER_ID", "GROUP_ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS"  ENABLE;
```

Du GUI :





2. Ajoutez les groupes et les mappages, de sorte qu'**Alice** et **plomb** appartiennent pour grouper des **utilisateurs** et l'admin appartienne pour grouper des **admins**

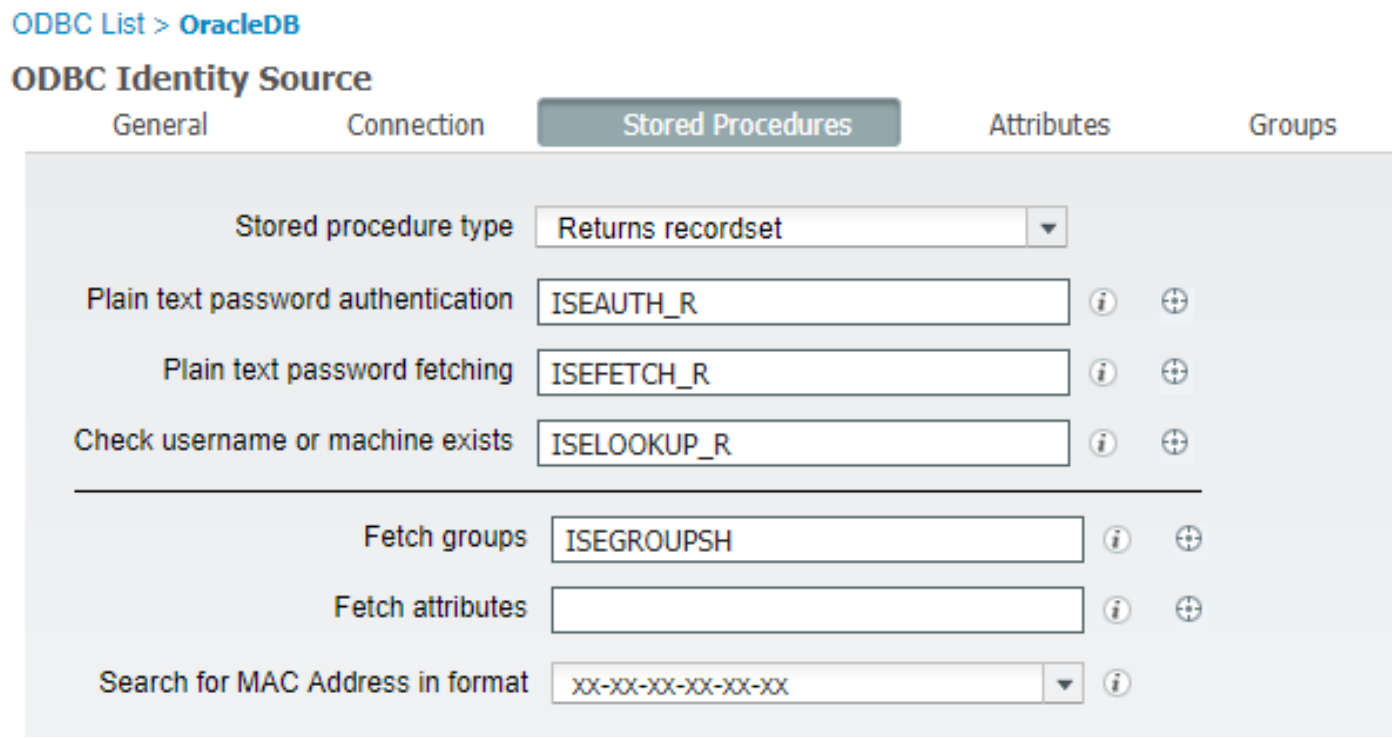
```
-- Adding groups
INSERT INTO "ISE"."GROUPS" (GROUP_NAME, DESCRIPTION) VALUES ('Admins', 'Group for
administrators')
INSERT INTO "ISE"."GROUPS" (GROUP_NAME, DESCRIPTION) VALUES ('Users', 'Corporate users')

-- Alice and Bob are users
INSERT INTO "ISE"."USER_GROUPS_MAPPING" (USER_ID, GROUP_ID) VALUES ('1', '2')
INSERT INTO "ISE"."USER_GROUPS_MAPPING" (USER_ID, GROUP_ID) VALUES ('2', '2')

-- Admin is in Admins group
INSERT INTO "ISE"."USER_GROUPS_MAPPING" (USER_ID, GROUP_ID) VALUES ('3', '1')
```

3. Créez une procédure de récupération de groupe. Il retourne tous les groupes si le nom d'utilisateur est « * »

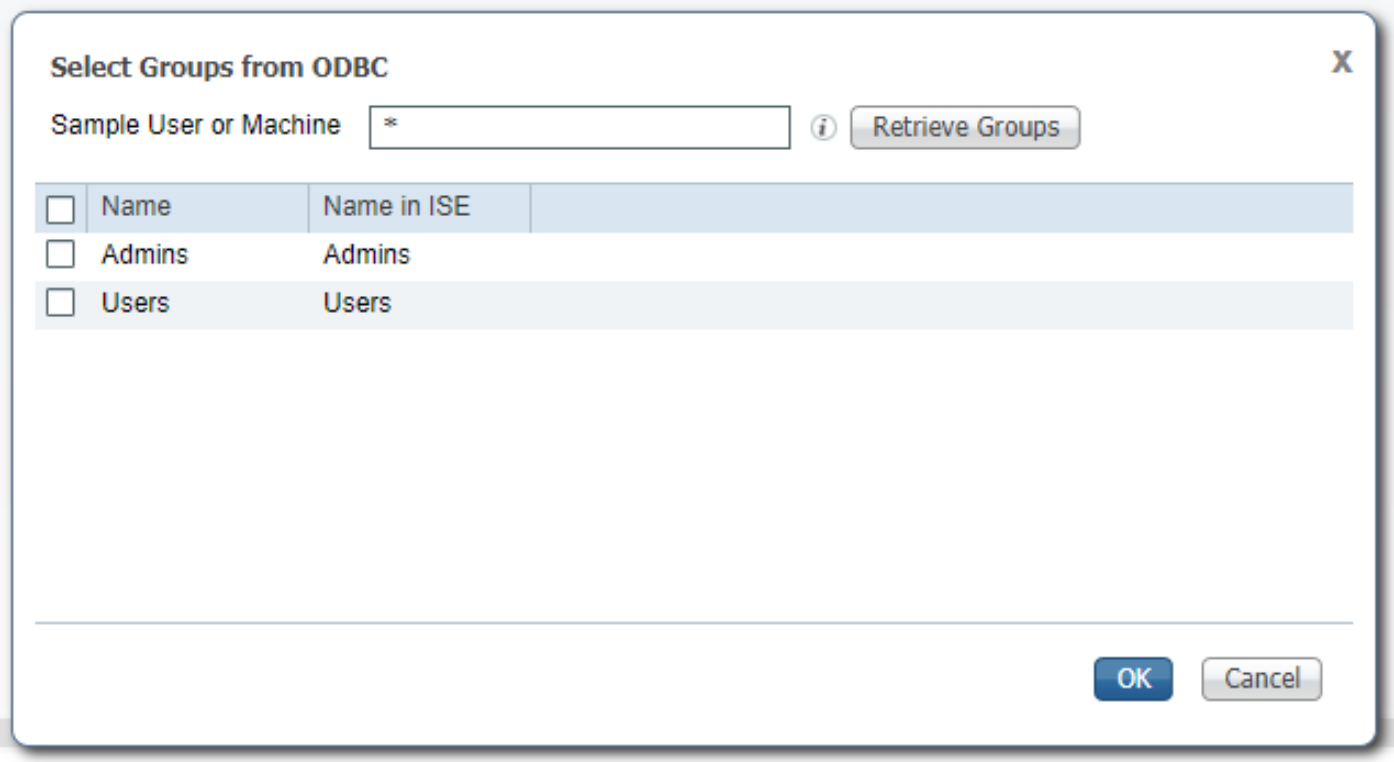
```
create or replace function ISEGROUPSH
(
  ise_username IN VARCHAR2,
  ise_result OUT int
) return sys_refcursor as
BEGIN
  declare
    c integer;
```

```
    userid integer;
    resultSet SYS_REFCURSOR;
  begin
    IF ise_username = '*' then
      ise_result := 0;
      open resultSet for select GROUP_NAME from GROUPS;
    ELSE
      select count(*) into c from USERS where USERS.USERNAME = ise_username;
      select USER_ID into userid from USERS where USERS.USERNAME = ise_username;
      IF c > 0 then
          ise_result := 0;
          open resultSet for select GROUP_NAME from GROUPS where GROUP_ID IN ( SELECT m.GROUP_ID
from USER_GROUPS_MAPPING m where m.USER_ID = userid );
      ELSE
          ise_result := 3;
          open resultSet for select 0 from dual where 1=2;
      END IF;
    END IF;
    return resultSet;
  end;
END ;
```

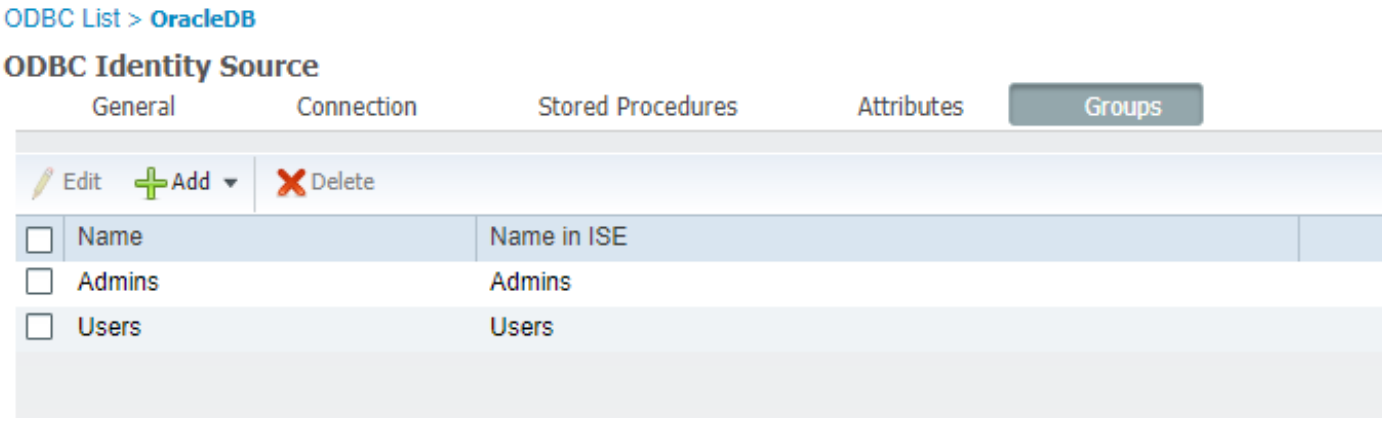
4. Tracez-le **pour chercher des groupes**



5. Cherchez les groupes et ajoutez-les dans la **source d'identité ODBC**



Select a eu besoin de groupes et clique sur OK, ils apparaîtra sur l'onglet de **groupes**



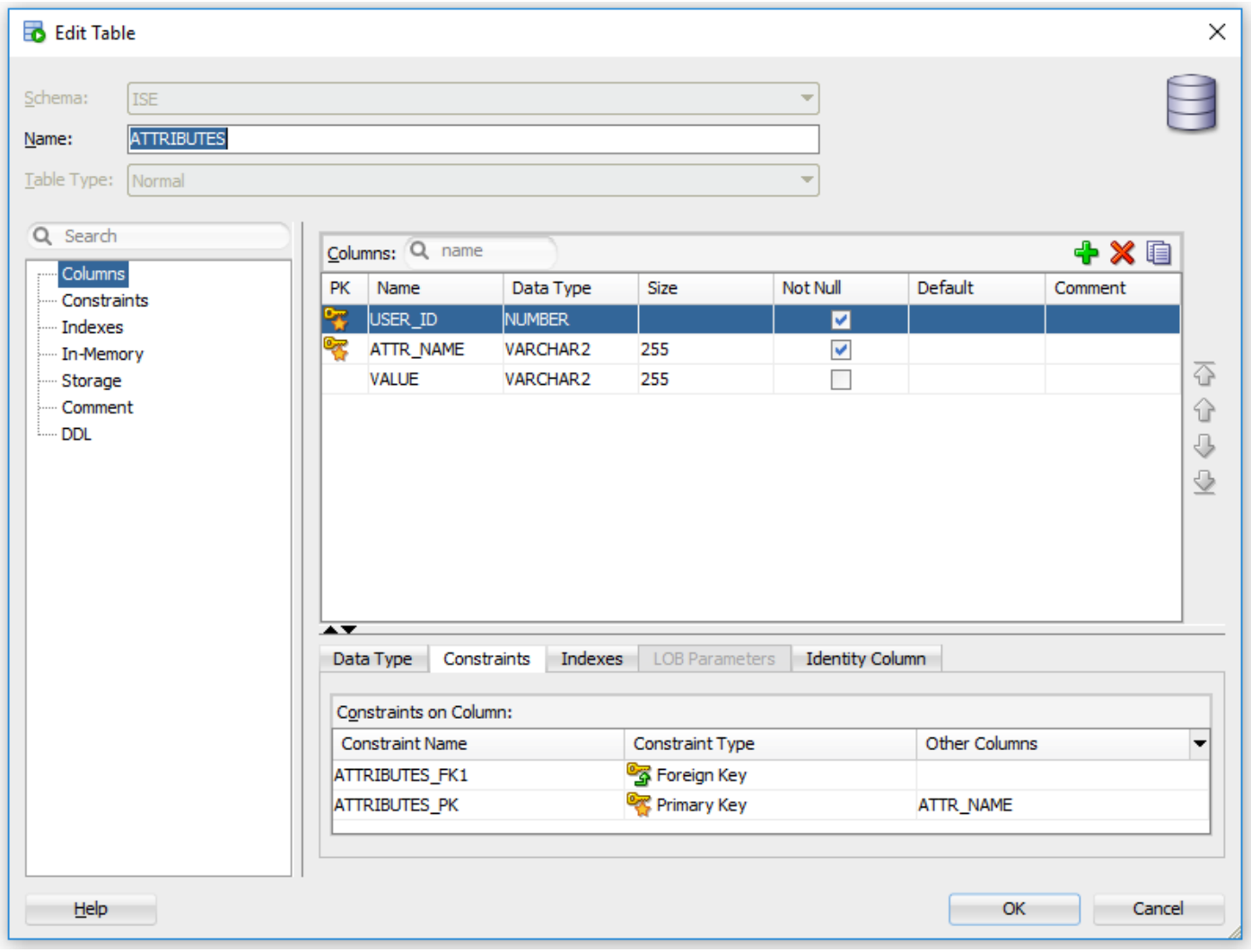
Étape 5. Configurez la récupération d'attributs

1. Afin de simplifier cet exemple, une table plate est utilisée pour des attributs

```
-------------------------------------------------------
--  DDL for Table ATTRIBUTES
-------------------------------------------------------

  CREATE TABLE "ISE"."ATTRIBUTES"
   ("USER_ID" NUMBER(*,0),
"ATTR_NAME" VARCHAR2(255 BYTE),
"VALUE" VARCHAR2(255 BYTE)
   ) SEGMENT CREATION IMMEDIATE
  PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS" ;
```

```
--------------------------------------------------------
--  DDL for Index ATTRIBUTES_PK
--------------------------------------------------------

  CREATE UNIQUE INDEX "ISE"."ATTRIBUTES_PK" ON "ISE"."ATTRIBUTES" ("ATTR_NAME", "USER_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS" ;
--------------------------------------------------------
--  Constraints for Table ATTRIBUTES
--------------------------------------------------------

  ALTER TABLE "ISE"."ATTRIBUTES" MODIFY ("USER_ID" NOT NULL ENABLE);
  ALTER TABLE "ISE"."ATTRIBUTES" MODIFY ("ATTR_NAME" NOT NULL ENABLE);
  ALTER TABLE "ISE"."ATTRIBUTES" ADD CONSTRAINT "ATTRIBUTES_PK" PRIMARY KEY ("ATTR_NAME",
"USER_ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1
  BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT)
  TABLESPACE "USERS"  ENABLE;
```

Du GUI :



2. Créez quelques attributs pour des utilisateurs

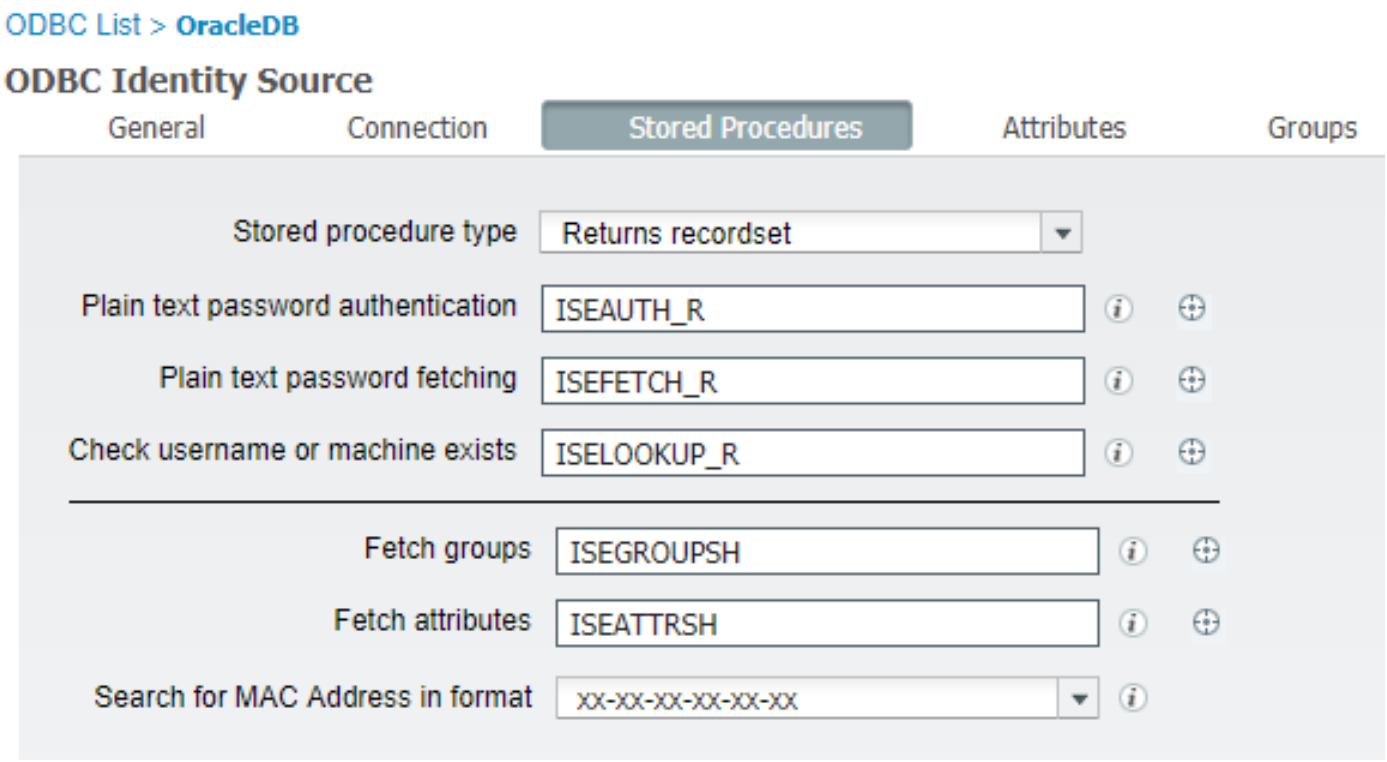
```
INSERT INTO "ISE"."ATTRIBUTES" (USER_ID, ATTR_NAME, VALUE) VALUES ('3', 'SecurityLevel', '15')
```

```
INSERT INTO "ISE"."ATTRIBUTES" (USER_ID, ATTR_NAME, VALUE) VALUES ('1', 'SecurityLevel', '5')
INSERT INTO "ISE"."ATTRIBUTES" (USER_ID, ATTR_NAME, VALUE) VALUES ('2', 'SecurityLevel', '10')
```

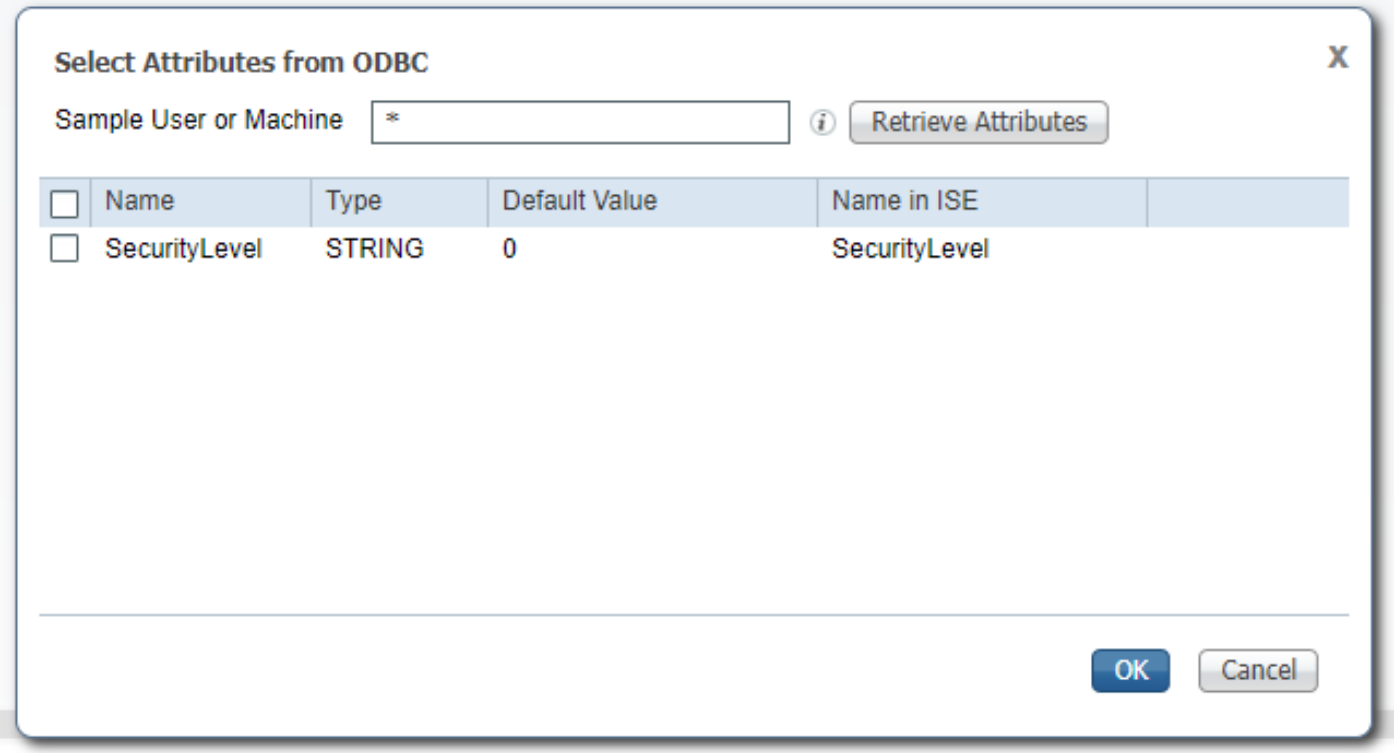
3. Créez une procédure. Mêmes qu'avec la récupération de groupes, il renverra tous les attributs distincts si le nom d'utilisateur est « * »

```
create or replace function ISEATTRSH
(
  ise_username IN VARCHAR2,
  ise_result OUT int
) return sys_refcursor as
BEGIN
  declare
    c integer;
    userid integer;
    resultSet SYS_REFCURSOR;
  begin
    IF ise_username = '*' then
      ise_result := 0;
      open resultSet for select DISTINCT ATTR_NAME, '0' as "VAL" from ATTRIBUTES;
    ELSE
      select count(*) into c from USERS where USERS.USERNAME = ise_username;
      select USER_ID into userid from USERS where USERS.USERNAME = ise_username;
      if c > 0 then
          ise_result := 0;
          open resultSet for select ATTR_NAME, VALUE from ATTRIBUTES where USER_ID = userid;
      ELSE
          ise_result := 3;
          open resultSet for select 0 from dual where 1=2;
      END IF;
    END IF;
    return resultSet;
  end;
END ;
```

4. Tracez-le **pour chercher des attributs**



5. Cherchez les attributs



Sélectionnez les attributs et cliquez sur OK.

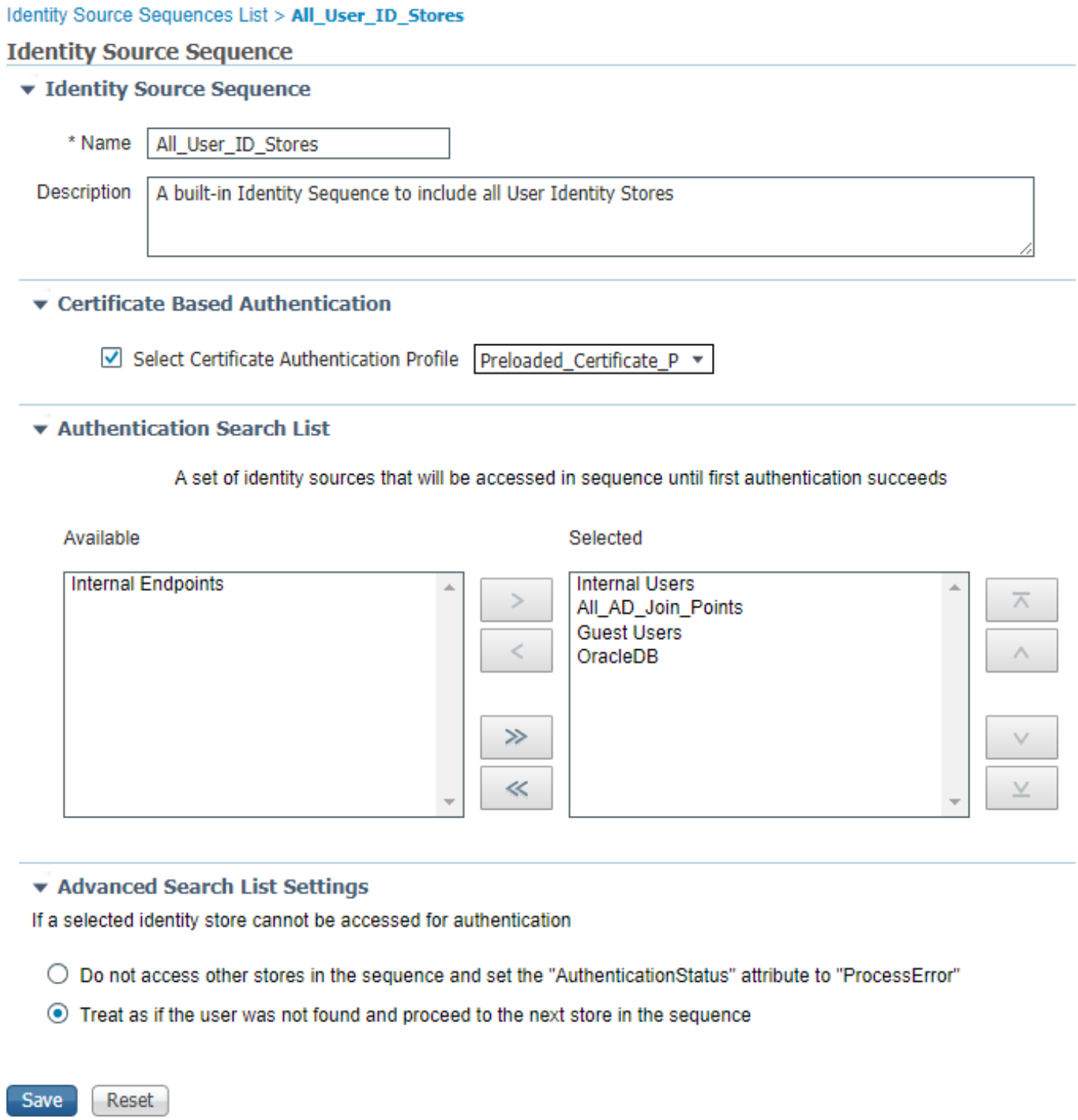
Étape 6. Configurez les stratégies d'authentification/autorisation

Dans cet exemple les stratégies simples suivantes d'autorisation ont été configurées :

Des utilisateurs avec **SecurityLevel = 5** seront refusés.

Étape 7. Ajoutez Oracle ODBC aux ordres de source d'identité

Naviguez vers des *ordres de source de gestion > de Gestion de l'identité > d'identité,* sélectionnez votre ordre et ajoutez ODBC à l'ordre :



Sauvegardez-le.

Vérifiez

Vous devriez maintenant pouvoir authentifier des utilisateurs contre ODBC à ce jour et récupérer leurs groupes et attributs.

Logs vivants de RADIUS

Exécutez quelques authentifications et naviguez vers des *exécutions > RADIUS > vivent des logs*

Time	Status	Details	Repeat ...	Identity	Endpoint ID	Endpoint P...	Authenticat...	Authorizati...	Authorizati...	IP Address	Network Device
				Identity	Endpoint ID	Endpoint Prof	Authenticatior	Authorization	Authorization	IP Address	Network Device
Aug 08, 2017 04:31:32.545 PM				badUser	92:77:F1:E4:D2:53		Default >> D...	Default			SWITCH
Aug 08, 2017 04:31:32.465 PM			0	admin	61:AD:77:0F:DF:CF	FreeBSD-W...	Default >> D...	Default >> A...	PermitAccess	83.133.106.96	
Aug 08, 2017 04:31:32.460 PM				admin	61:AD:77:0F:DF:CF		Default >> D...	Default >> A...	PermitAccess		SWITCH
Aug 08, 2017 04:31:32.365 PM			0	bob	FC:F4:97:F2:F5:4F		Default >> D...	Default >> A...	PermitAccess	241.97.134.20	
Aug 08, 2017 04:31:32.359 PM				bob	FC:F4:97:F2:F5:4F		Default >> D...	Default >> A...	PermitAccess		SWITCH
Aug 08, 2017 04:31:32.237 PM				alice	42:27:B1:C6:F9:A4		Default >> D...	Default >> S...	DenyAccess		SWITCH

Comme vous pouvez voir, l'utilisateur Alice a **SecurityLevel = 5,** par conséquent l'accès a été rejeté.

État de détail

Cliquez sur en fonction l'**état de détail** dans la colonne de **détails** pour la session intéressante pour vérifier l'écoulement.

Rapport détaillé pour l'utilisateur Alice (en raison rejeté de bas SecurityLevel) :