

# Configurazione di ISE 2.2 per l'integrazione con il server MySQL

## Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Premesse](#)

[Configurazione](#)

[Esempio di rete](#)

[Configurazioni](#)

[1. Configurare MySQL su Ubuntu:](#)

[2. Configurare il database e le tabelle:](#)

[3. Configurare le stored procedure](#)

[4. Integrare ISE con MySQL:](#)

[5. Configurare i criteri di autenticazione e autorizzazione:](#)

[Verifica](#)

[Risoluzione dei problemi](#)

[Debug su ISE](#)

[Informazioni correlate](#)

## Introduzione

In questo documento viene descritto come configurare Cisco Identity Services Engine (ISE) 2.2 per l'integrazione con l'origine esterna ODBC (Open Database Connectivity) di MySQL. Questo documento è valido per le configurazioni che usano MySQL come origine identità esterna per l'autenticazione e l'autorizzazione ISE.

## Prerequisiti

### Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Configurazione Identity Services Engine (ISE)
- Configurazione MySQL di base

### Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

- Cisco ISE versione 2.2
- Ubuntu Linux con MySQL installato
- Cisco Wireless LAN Controller (WLC) versione 8.0.100.0
- Microsoft Windows versione 7x64

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

## Premesse

ISE 2.2 supporta più origini esterne ODBC, una delle quali è MySQL. È possibile utilizzare ODBC come origine identità esterna per autenticare utenti ed endpoint simili ad Active Directory (AD). L'origine identità ODBC può essere utilizzata in una sequenza di archivio identità e per le autenticazioni di Guest e Sponsor.

Questo è un elenco dei motori di database supportati in ISE 2.2:

- MySQL
- Oracle
- PostgreSQL
- Microsoft SQL Server
- Sybase

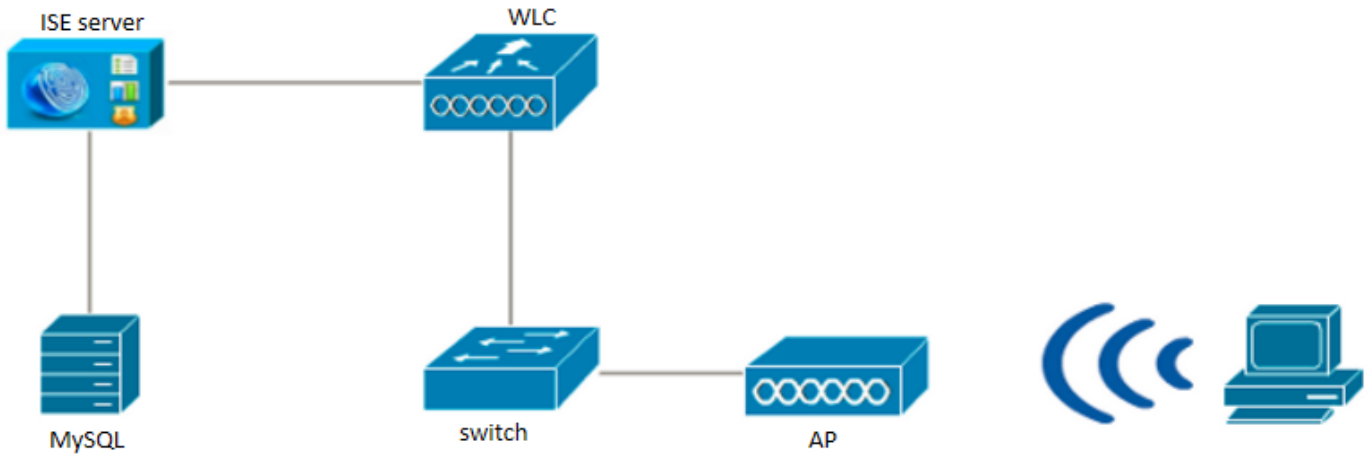
Ulteriori informazioni sono disponibili all'indirizzo:

[https://www.cisco.com/c/en/us/td/docs/security/ise/2-2/admin\\_guide/b\\_ise\\_admin\\_guide\\_22/b\\_ise\\_admin\\_guide\\_22\\_chapter\\_01101.html#concept\\_6EB9B4875CBB47D79168E329696E2C65](https://www.cisco.com/c/en/us/td/docs/security/ise/2-2/admin_guide/b_ise_admin_guide_22/b_ise_admin_guide_22_chapter_01101.html#concept_6EB9B4875CBB47D79168E329696E2C65)

## Configurazione

### Esempio di rete

In questo esempio di configurazione, l'endpoint utilizza una scheda wireless per l'associazione alla rete wireless. La LAN wireless (WLAN) sul WLC è configurata in modo da autenticare gli utenti tramite l'ISE. In ISE, MySQL è configurato come un archivio identità esterno. Nell'immagine è illustrata la topologia di rete utilizzata:



## Configurazioni

La configurazione MySQL presentata è un esempio. Non trattare è come un consiglio di Cisco.

### 1. Configurare MySQL su Ubuntu:

Aggiornare il sistema:

```
sudo apt-get update
sudo apt-get upgrade
```

Installa MySQL (durante l'installazione viene richiesta una password per l'utente root):

```
sudo apt-get install mysql-server
```

Per accedere al database MySQL:

```
mysql -u root -p
```

### 2. Configurare il database e le tabelle:

Crea database:

```
mysql>
mysql> CREATE DATABASE demo_db;
Query OK, 1 row affected (0.00 sec)
mysql>
mysql> use demo_db;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Database changed
```

Creare un utente del database e concedergli i privilegi riportati di seguito.

```
mysql>
mysql> CREATE USER 'cisco' IDENTIFIED BY 'cisco';
mysql> GRANT USAGE ON *.* TO 'cisco'@'%';
mysql> GRANT ALL PRIVILEGES ON `demo_db`.* TO 'cisco'@'%';
mysql> GRANT SELECT ON *.* TO 'cisco'@'%';
```

## Crea tabella di utenti:

```
mysql>
mysql> CREATE TABLE `users` (
-> `user_id` int(10) unsigned NOT NULL AUTO_INCREMENT,
-> `username` varchar(50) NOT NULL,
-> `password` varchar(50) NOT NULL,
-> PRIMARY KEY (`user_id`),
-> UNIQUE KEY `username_UNIQUE` (`username`)
-> ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Query OK, 0 rows affected (0.01 sec)
```

## Creare gli utenti e aggiungerli alla tabella:

```
mysql>
mysql> INSERT INTO users
-> (user_id, username, password)
-> VALUES
-> (1, "alice", "Krakow123");
Query OK, 1 row affected (0.00 sec)
```

È possibile aggiungere altri utenti in modo simile ed elencare il contenuto della tabella (allo stesso modo degli utenti, aggiungere l'indirizzo MAC per l'autenticazione MAB - la password può rimanere vuota):

```
mysql>
mysql> select * from users;
+-----+-----+-----+
| user_id | username | password |
+-----+-----+-----+
| 1       | alice    | Krakow123 |
| 2       | bob      | Krakow123 |
| 3       | oscar    | Krakow123 |
+-----+-----+-----+
```

## Crea tabella di gruppi:

```
mysql>
mysql> CREATE TABLE `groups` (
-> `group_id` int(10) unsigned NOT NULL AUTO_INCREMENT,
-> `groupname` varchar(50) NOT NULL,
-> PRIMARY KEY (`group_id`),
-> UNIQUE KEY `groupname_UNIQUE` (`groupname`)
-> ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Query OK, 0 rows affected (0.01 sec)
```

## Creare i gruppi e aggiungerli alla tabella:

```
mysql>
mysql> INSERT INTO groups
```

```
-> (group_id, groupname)
-> VALUES
-> (1, "everyone");
Query OK, 1 row affected (0.00 sec)
```

È possibile aggiungere altri gruppi in modo simile ed elencare il contenuto della tabella:

```
mysql>
mysql> select * from groups;
+-----+-----+
| group_id | groupname |
+-----+-----+
| 3        | contractor |
| 2        | employee   |
| 1        | everyone   |
+-----+-----+
```

Crea tabella per i mapping tra utenti e gruppi

```
mysql>
mysql> CREATE TABLE `user_group` (
-> `user_id` int(10) unsigned NOT NULL,
-> `group_id` int(10) unsigned NOT NULL,
-> PRIMARY KEY (`user_id`,`group_id`),
-> KEY `group_id` (`group_id`),
-> CONSTRAINT `user_group_ibfk_1` FOREIGN KEY (`user_id`) REFERENCES `users` (`user_id`)
-> ON DELETE CASCADE,
-> CONSTRAINT `user_group_ibfk_2` FOREIGN KEY (`group_id`) REFERENCES `groups`
-> (`group_id`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Query OK, 0 rows affected (0.01 sec)
```

Compilare la tabella per i mapping tra utenti e gruppi

```
mysql>
mysql> INSERT INTO user_group
-> (user_id, group_id)
-> VALUES
-> (1, 1);
Query OK, 1 row affected (0.00 sec)
```

È possibile aggiungere altri mapping in modo simile ed elencare il contenuto della tabella:

```
mysql>
mysql> select * from user_group;
+-----+-----+
| user_id | group_id |
+-----+-----+
| 1        | 1        |
| 2        | 1        |
| 1        | 2        |
| 2        | 3        |
+-----+-----+
```

4 rows in set (0.00 sec)

### 3. Configurare le stored procedure

È necessario configurare le stored procedure necessarie per autenticare gli utenti in base a un'origine identità ODBC. I task eseguiti in base alla procedura variano in base al protocollo di autenticazione. ISE supporta tre diversi tipi di controllo delle credenziali rispetto all'archivio esterno ODBC. È necessario configurare una stored procedure separata per ogni tipo di controllo. ISE chiama la stored procedure appropriata con parametri di input e riceve l'output. Il database può restituire un recordset o un set di parametri denominati in risposta a una query ODBC.

- **Autenticazione password in testo normale nel database ODBC:** l'autenticazione per PAP e PEAP viene eseguita all'interno del database. Se viene individuata una combinazione nome utente/password corrispondente all'input, l'utente viene autenticato.
- **Recupero della password in testo normale dal database ODBC:** autenticazione per CHAP, MS-CHAPv1/v2, EAP-MD5, LEAP e EAP-MSCHAPv2 (come metodo interno di PEAP o EAP-FAST) eseguita in Cisco ISE (ISE verifica la password fornita dall'utente e la confronta con la password ricevuta dalla stored procedure). La stored procedure restituisce la password se il nome utente è corretto. Se il nome utente non viene trovato, verrà restituito un codice di errore.
- **Ricerca:** l'autenticazione per MAB viene eseguita nel database. Se viene trovato il nome utente richiesto, i parametri rilevanti vengono restituiti all'ISE.

Ognuna di queste procedure deve essere definita con delimitatore affinché MySQL accetti la sintassi della query:

```
DELIMITER //
CREATE DEFINER=`root`@`localhost` PROCEDURE `ISEGroups`(username varchar(64), OUT result INT)
begin
CASE username
WHEN '*' THEN
select distinct groupname from groups;
ELSE
select groupname from user_group
inner join users ON users.user_id = user_group.user_id
inner join groups ON groups.group_id = user_group.group_id
where users.username = username;
END CASE;
SET result = 0;
end //
```

```
DELIMITER //
CREATE DEFINER=`root`@`localhost` PROCEDURE `ISEAuthUserPlainReturnsRecordset`(username
varchar(64), password varchar(255))
begin
IF EXISTS (select * from users where users.username = username and users.password = password )
THEN
select 0,11,'This is a very good user, give him all access','no error';
ELSE
select 3, 0, 'odbc','ODBC Authen Error';
END IF;
end //
```

```
DELIMITER //
CREATE DEFINER=`root`@`localhost` PROCEDURE `ISEFetchPasswordReturnsRecordset`(username
varchar(64))
begin
```

```

IF EXISTS (select * from users where users.username = username) THEN
select 0,11,'This is a very good user, give him all access','no error',password from users where
users.username = username;
ELSE
select 3, 0, 'odbc','ODBC Authen Error';
END IF;
end //

```

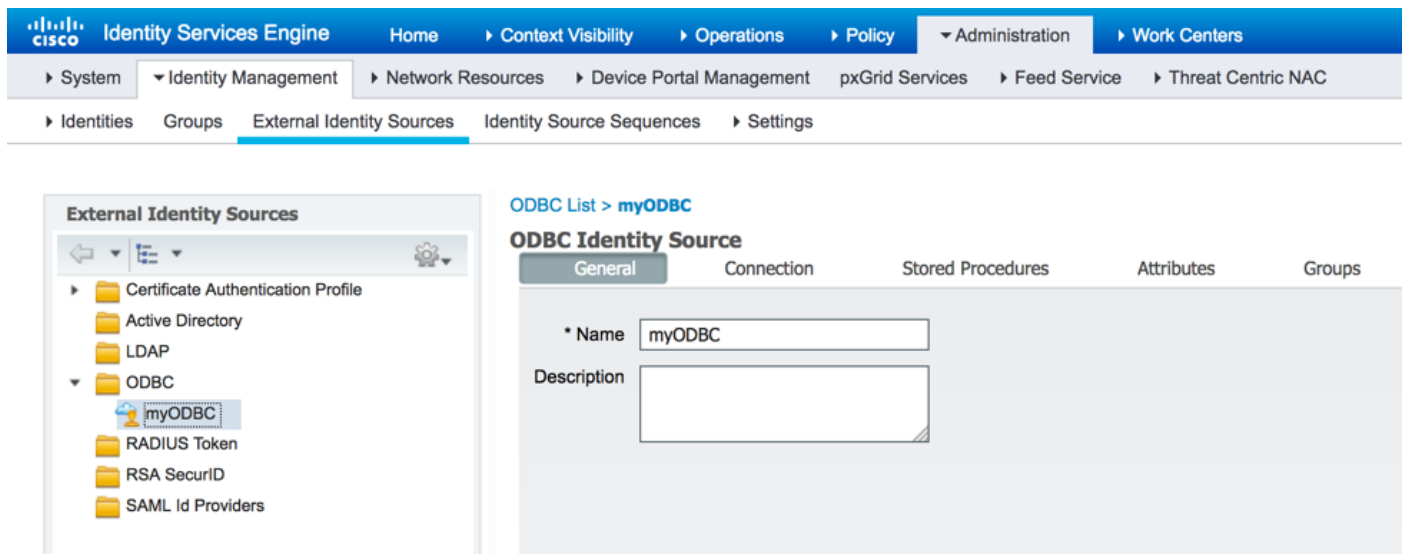
```

DELIMITER //
CREATE DEFINER=`root`@`localhost` PROCEDURE `ISEUserLookupReturnsRecordset`(username
varchar(64))
begin
IF EXISTS (select * from users where users.username = username) THEN
select 0,11,'This is a very good user, give him all access','no error';
ELSE
select 3, 0, 'odbc','ODBC Authen Error';
END IF;
end //

```

#### 4. Integrare ISE con MySQL:

Per integrare MySQL con Cisco ISE, usare le informazioni riportate di seguito. Passare a **Amministrazione > Gestione delle identità > Origini identità esterne > ODBC** e aggiungere un nuovo archivio:



Utilizzare l'indirizzo IP di Ubuntu che esegue il database MySQL come nome host/indirizzo IP di seguito. Specificare il tipo di database (in questa situazione viene utilizzato MySQL), inserire anche il nome del database e le credenziali utente del database create in precedenza:

**External Identity Sources**

- Certificate Authentication Profile
- Active Directory
- LDAP
- ODBC
  - myODBC
- RADIUS Token
- RSA SecurID
- SAML Id Providers

ODBC List > myODBC

**ODBC Identity Source**

General **Connection** Stored Procedures Attributes Groups

**ODBC DB connection details**

\* Hostname/IP[:port]

\* Database name

Admin username  ⓘ

Admin password

\* Timeout  Seconds (1-30)

\* Retries  (0-3)

\* Database type

Specificare i nomi delle routine create in MySQL. È necessario prestare attenzione al formato dell'indirizzo MAC (in questo esempio è stato modificato in un formato diverso):

**External Identity Sources**

- Certificate Authentication Profile
- Active Directory
- LDAP
- ODBC
  - myODBC
- RADIUS Token
- RSA SecurID
- SAML Id Providers

ODBC List > myODBC

**ODBC Identity Source**

General Connection **Stored Procedures** Attributes Groups

Stored procedure type

Plain text password authentication  ⓘ ⊕

Plain text password fetching  ⓘ ⊕

Check username or machine exists  ⓘ ⊕

---

Fetch groups  ⓘ ⊕

Fetch attributes  ⓘ ⊕

Search for MAC Address in format  ⓘ

Al termine, tornare alla scheda **Connessione** e verificare la connessione:



Identity Services Engine Home > Context Visibility > Operations > Policy > Administration > Work Centers

System > Identity Management > Network Resources > Device Portal Management pxGrid Services > Feed Service > Threat Centric NAC

Identities Groups External Identity Sources Identity Source Sequences Settings

External Identity Sources

- Certificate Authentication Profile
- Active Directory
- LDAP
- ODBC
  - myODBC
  - RADIUS Token
  - RSA SecurID
  - SAML Id Providers

ODBC List > myODBC

ODBC Identity Source

General Connection Stored Procedures Attributes Groups

ODBC DB connection details

\* Hostname/IP[port] 10.48.26.137

\* Database name demo\_db

Admin username veronika

Admin password .....

\* Timeout 5

\* Retries 1

\* Database type MySQL

Test Connection

**Test connection**

✓ Connection succeeded

**Stored Procedures**

✓ Plain text password authentication - Exists

✓ Plain text password fetching - Exists

✓ Check username or machine exists - Exists

✓ Fetch groups - Exists

✓ Fetch attributes - Exists

Close

Recuperare gli attributi da MySQL, fare clic sulla scheda **Attributi**:

Identity Services Engine Home > Context Visibility > Operations > Policy > Administration > Work Centers License Warning

System > Identity Management > Network Resources > Device Portal Management pxGrid Services > Feed Service > Threat Centric NAC

Identities Groups External Identity Sources Identity Source Sequences Settings

External Identity Sources

- Certificate Authentication Profile
- Active Directory
- LDAP
- ODBC
  - myODBC
  - RADIUS Token
  - RSA SecurID
  - SAML Id Providers

ODBC List > myODBC

ODBC Identity Source

General Connection Stored Procedures Attributes Groups

Edit + Add - Delete

<input type="checkbox"/> Name	Type	Default Value	Name in ISE
<input type="checkbox"/> eye_color	STRING	green	eye_color
<input type="checkbox"/> floor	STRING	1	floor
<input type="checkbox"/> is_certified	STRING	true	is_certified

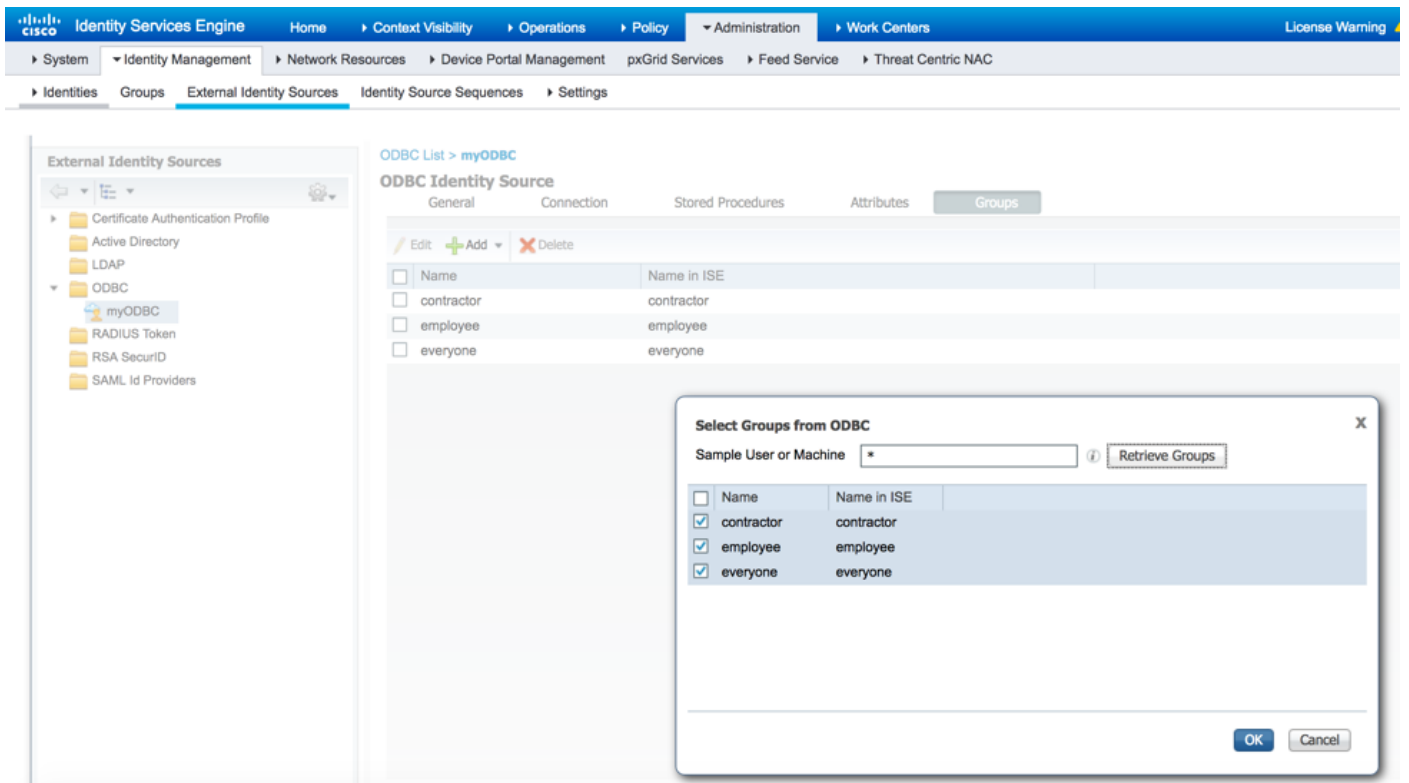
**Select Attributes from ODBC**

Sample User or Machine  Retrieve Attributes

<input type="checkbox"/> Name	Type	Default Value	Name in ISE
<input checked="" type="checkbox"/> eye_color	STRING	green	eye_color
<input checked="" type="checkbox"/> floor	STRING	1	floor
<input checked="" type="checkbox"/> is_certified	STRING	true	is_certified

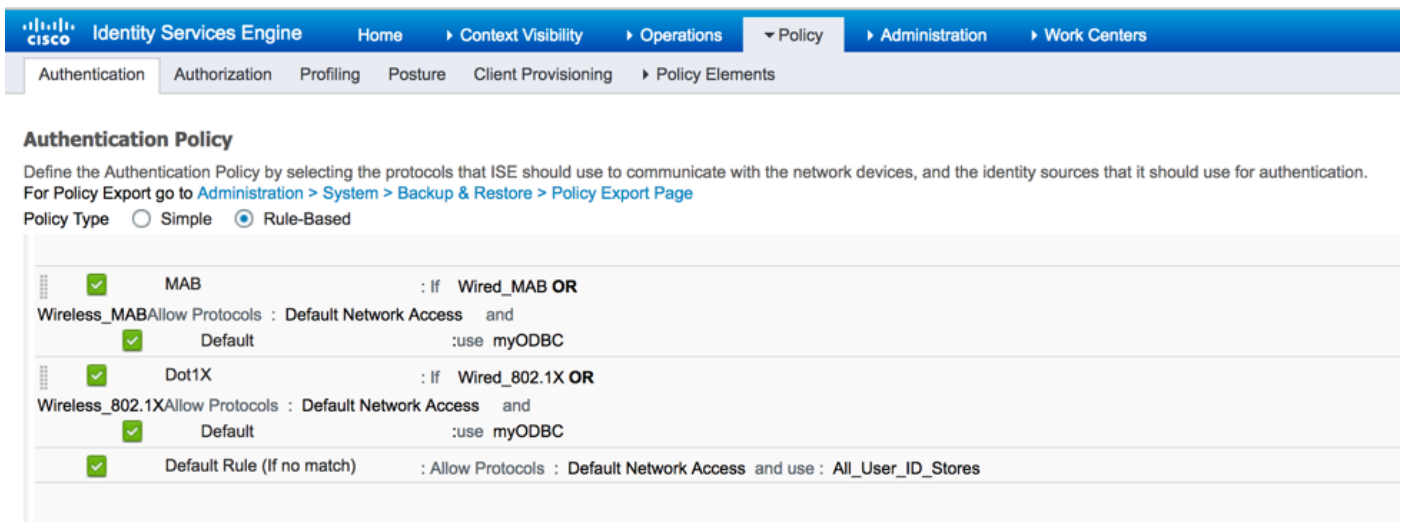
OK Cancel

Recupera gruppi nello stesso modo:



## 5. Configurare i criteri di autenticazione e autorizzazione:

Configurare ISE per autenticare e autorizzare gli utenti dal database MySQL. Selezionare **Criterio > Autenticazione** e **Criterio > Autorizzazione**:



Identity Services Engine Home Context Visibility Operations Policy Administration Work Centers License Warning

Authentication Authorization Profiling Posture Client Provisioning Policy Elements

### Authorization Policy

Define the Authorization Policy by configuring rules based on identity groups and/or other conditions. Drag and drop rules to change the order.  
For Policy Export go to Administration > System > Backup & Restore > Policy Export Page

First Matched Rule Applies

Exceptions (0)

Standard

Status	Rule Name	Conditions (identity groups and other conditions)	Permissions	
<input checked="" type="checkbox"/>	Wireless Black List Default	if Blacklist AND Wireless_Access	then Blackhole_Wireless_Access	Edit   v
<input checked="" type="checkbox"/>	Contractor	if myODBC-ExternalGroups EQUALS contractor	then for_contractor	Edit   v
<input checked="" type="checkbox"/>	Employee	if myODBC-ExternalGroups EQUALS employee	then for_employee	Edit   v
<input checked="" type="checkbox"/>	Profiled Cisco IP Phones	if Cisco-IP-Phone	then Cisco_IP_Phones	Edit   v
<input checked="" type="checkbox"/>	Profiled Non Cisco IP Phones	if Non_Cisco_Profiled_Phones	then Non_Cisco_IP_Phones	Edit   v
<input type="checkbox"/>	Compliant_Devices_Access	if (Network_Access_Authentication_Passed AND Compliant_Devices )	then PermitAccess	Edit   v
<input type="checkbox"/>	Employee_EAP-TLS	if (Wireless_802.1X AND BYOD_is_Registered AND EAP-TLS AND MAC_in_SAN )	then PermitAccess AND BYOD	Edit   v
<input type="checkbox"/>	Employee_Onboarding	if (Wireless_802.1X AND EAP-MSCHAPv2 )	then NSP_Onboard AND BYOD	Edit   v
<input type="checkbox"/>	Wi-Fi_Guest_Access	if (Guest_Flow AND Wireless_MAB )	then PermitAccess AND Guests	Edit   v
<input type="checkbox"/>	Wi-Fi_Redirect_to_Guest_Login	if Wireless_MAB	then Cisco_WebAuth	Edit   v

Save Reset

## Verifica

Sono stati testati due flussi di autenticazione: PEAP-MSCHAPv2 e MAB. Alice fa parte del gruppo di dipendenti in MySQL, Bob fa parte del gruppo di collaboratori esterni:

Identity Services Engine Home Context Visibility Operations Policy Administration Work Centers License Warning

RADIUS Threat-Centric NAC Live Logs TACACS Troubleshoot Adaptive Network Control Reports

Live Logs Live Sessions

Misconfigured Supplicants 0 Misconfigured Network Devices 0 RADIUS Drops 0 Client Stopped Responding 1 Repeat Counter 0

Refresh Never Show Latest 20 records Within Last 3 hours

Refresh Reset Repeat Counts Export To Filter

Time	Status	Details	Identity	Authentication ...	Authorization P...	Authorization P...	Event	Failure Reason
Feb 18, 2017 02:13:38.0...	<input type="checkbox"/>		bob	Default >> Dot1X...	Default >> Contr...	for_contractor	Session State is Started	
Feb 18, 2017 02:13:37.8...	<input checked="" type="checkbox"/>		bob	Default >> Dot1X...	Default >> Contr...	for_contractor	Authentication succeeded	
Feb 18, 2017 02:12:19.4...	<input checked="" type="checkbox"/>		alice	Default >> Dot1X...	Default >> Emplo...	for_employee	Authentication succeeded	

## Risoluzione dei problemi

### Debug su ISE

Per abilitare i debug su ISE, selezionare **Amministrazione > Sistema > Registrazione > Configurazione log di debug**, selezionare il nodo PSN e modificare il livello di log del componente **odbc-id-store** in **DEBUG**:

Registri da controllare: prrt-server.log e prrt-management.log. È possibile archivarli direttamente dalla CLI di ISE:

```
vchrenek-ise22-1/admin# show logging application port-management.log tail
```

Durante l'autenticazione dell'utente bob, ISE deve recuperare la password in testo normale e viene utilizzata la seguente stored procedure **ISEFetchPasswordReturnsRecordset**:

```
2017-02-18 14:13:37,565 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC ID Store Operation: Fetch Plain Text Password. Username=bob, SessionID=0a3e9466000090658a8487f
2017-02-18 14:13:37,566 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.CustomerLog -:::- Write customer log message: 24861
2017-02-18 14:13:37,567 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:::- OdbcConnectionPool - get connection
2017-02-18 14:13:37,567 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:::- OdbcConnectionPool - use existing connection
2017-02-18 14:13:37,568 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:::- OdbcConnectionPool - connections in use: 1
2017-02-18 14:13:37,568 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::- Fetch plain text password
2017-02-18 14:13:37,568 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::- Prepare stored procedure call, procname=ISEFetchPasswordReturnsRecordset
2017-02-18 14:13:37,568 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::- Using recordset to obtain stored procedure result values
2017-02-18 14:13:37,568 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.CustomerLog -:::- Write customer log message: 24855
2017-02-18 14:13:37,568 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::- Text: {call ISEFetchPasswordReturnsRecordset(?)}
```

odbcStoredProcedureCustomerErrorString=null, accountInfo=This is a very good user, give him all access, group=11

Poiché ISE deve controllare l'assegnazione dei gruppi ODBC, deve recuperare i gruppi:

```
2017-02-18 14:13:37,572 DEBUG [Thread-493][] cisco.cpm.odbcidstore.impl.CustomerLog -::::- Write customer log message: 24862
2017-02-18 14:13:37,728 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -::::- ODBC ID Store Operation: Get all user groups. Username=bob, SessionID=0a3e9466000090658a8487f
2017-02-18 14:13:37,728 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -::::- ODBC ID Store Operation: Fetch user groups. Username=bob, SessionID=0a3e9466000090658a8487f
2017-02-18 14:13:37,728 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.CustomerLog -::::- Write customer log message: 24869
2017-02-18 14:13:37,729 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -::::- OdbcConnectionPool - get connection
2017-02-18 14:13:37,729 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -::::- OdbcConnectionPool - use existing connection
2017-02-18 14:13:37,729 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -::::- OdbcConnectionPool - connections in use: 1
2017-02-18 14:13:37,729 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Fetch user groups
2017-02-18 14:13:37,729 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Prepare stored procedure call, procname=ISEGroups
2017-02-18 14:13:37,729 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Text: {call ISEGroups(?,?) }
2017-02-18 14:13:37,733 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Setup stored procedure input parameters, username=bob
2017-02-18 14:13:37,733 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Execute stored procedure call
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Process stored procedure results
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Received result recordset, total number of columns=1
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- According to column number expect multiple rows (vertical attributes/groups returned result)
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Fetch data: ExternalGroup=everyone
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Fetch data: ExternalGroup=contractor
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Results successfully parsed from recordset
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -::::- Result code indicates success
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -::::- OdbcConnectionPool - release connection
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -::::- OdbcConnectionPool - connections in use: 0
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -::::- Call to ODBC DB succeeded
2017-02-18 14:13:37,740 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.CustomerLog -::::- Write customer log message: 24870
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -::::- ODBC ID Store Operation: Get all user groups. Got groups...
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -::::- ODBC ID Store Operation: Get all user groups. Got groups(0) = everyone
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -::::- ODBC ID Store Operation: Get all user groups. Setting Internal groups(0) = everyone
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -::::- ODBC ID Store Operation: Get all user groups. Got groups(1) = contractor
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -::::- ODBC ID Store Operation: Get all user groups. Setting Internal groups(1) = contractor
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -::::- ODBC ID Store Operation: Get all user groups. Username=bob, ExternalGroups=[everyone, contractor]
```

```
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC
ID Store Operation: Fetch user attributes. Username=bob, SessionID=0a3e9466000090658a8487f
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.CustomerLog -:::- Write
customer log message: 24872
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -
:::- OdbcConnectionPool - get connection
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -
:::- OdbcConnectionPool - use existing connection
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -
:::- OdbcConnectionPool - connections in use: 1
```

Lo stesso vale per gli attributi:

```
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Fetch user attributes
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Prepare stored procedure call, procname=ISEAttrSH
2017-02-18 14:13:37,741 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Text: {call ISEAttrSH(?,?) }
2017-02-18 14:13:37,745 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Setup stored procedure input parameters, username=bob
2017-02-18 14:13:37,746 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Execute stored procedure call
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Process stored procedure results
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Received result recordset, total number of columns=3
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
According to column number expect multiple columns (horizontal attributes/groups returned result)
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Fetch data: eye_color=green
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Fetch data: floor=1
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Fetch data: is_certified=true
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Results successfully parsed from recordset
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Result code indicates success
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -
:::- OdbcConnectionPool - release connection
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -
:::- OdbcConnectionPool - connections in use: 0
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- Call
to ODBC DB succeeded
2017-02-18 14:13:37,749 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.CustomerLog -:::- Write
customer log message: 24873
2017-02-18 14:13:37,750 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC
ID Store Operation: Get all user attrs. Username=bob, Setting myODBC.eye_color to green
2017-02-18 14:13:37,750 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC
ID Store Operation: Get all user attrs. Username=bob, Setting myODBC.floor to 1
2017-02-18 14:13:37,750 DEBUG [Thread-259][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC
ID Store Operation: Get all user attrs. Username=bob, Setting myODBC.is_certified to true
```

## Informazioni correlate

- [Documentazione e supporto tecnico – Cisco Systems](#)
- [Note sulla release di ISE 2.2](#)
- [Guida all'installazione dell'hardware ISE 2.2](#)
- [Guida all'aggiornamento a ISE 2.2](#)
- [Guida per l'amministratore di ISE 2.2 Engine](#)