Introduction

This document describes how to configure Identity Services Engine (ISE) with Microsoft Standard Query Language (SQL) Server for ISE authentication using

**Note:** Open Database Connectivity (ODBC) authentication requires ISE to be able to fetch a plain text user password. The password can be encrypted in the database, but has to be decrypted by the *stored procedure*.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Database and ODBC concepts
- Microsoft SQL Server

Components Used

The information in this document is based on these software and hardware versions:

- Identity Services Engine 2.1
- MSSQL Server 2008 R2

Configure

Step 1. MS SQL Basic Configuration

Configuration steps include creating a database and one user for ISE with permissions to access
that database.

**Note**: ISE supports only SQL authentication, not the Windows account. If you need to change authentication mode, please refer to [Change Server Authentication Mode](#).

1. Open SQL Server Management Studio (Start menu > Microsoft SQL Server 2008 R2) and create a database:

![Microsoft SQL Server Management Studio](image1)

2. Leave default options or adjust database settings as shown in this image:

![Database Settings](image2)
3. Create a user and set permissions as shown in the images below:
Login - New

Select a page
- General
- Server Roles
- User Mapping
- Securables
- Status

Connection
- Server: localhost
- Connection: BABA\LAND\administrator

Progress
- Ready

Login name: ISEDBUser

- Windows authentication
- SQL Server authentication

Password: ********

Confirm password: ********

- Specify old password
- Old password: 

- Enforce password policy
- Enforce password expiration
- User must change password at next login

- Mapped to certificate
- Mapped to asymmetric key
- Map to Credential
- Mapped Credentials

Default database: ISEDB

Default language: <default>

Add

OK
Cancel

Login Properties - ISEDBUser

Selected page
- General
- Server Roles
- User Mapping
- Securables
- Status

Connection
- Server: localhost
- Connection: BABA\LAND\administrator

Progress
- Ready

Users mapped to this login:

<table>
<thead>
<tr>
<th>Map</th>
<th>Database</th>
<th>User</th>
<th>Default Schema</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISEDB</td>
<td>ISEDB</td>
<td>ISEDBUser</td>
<td></td>
</tr>
<tr>
<td>master</td>
<td>model</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>mssql</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>tempdb</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Guest account enabled for ISEDB

Database role membership for: ISEDB

- db_accessadmin
- db_backupsoperator
- db_dataroleadmin
- db_datarolereader
- db_developer
- db_devdatarole
- db_devdataroleadmin
- db_devdatawriter
- db_monitor
- db_securityadmin
- public

OK
Cancel
Step 2. ISE Basic Configuration

Create an **ODBC Identity Source** at Administration > External Identity Source > ODBC and test connection:

**ODBC List > ISE_ODBC**

**ODBC Identity Source**

Create an ODBC Identity Source at Administration > External Identity Source > ODBC and test connection:

**ODBC DB connection details**

- **Hostname/IP[:port]**: bast-ad-ca.cisco.com
- **Database name**: ISEDB
- **Admin username**: ISEDBUser
- **Admin password**: ********
- **Timeout**: 5
- **Retries**: 1
- **Database type**: Microsoft SQL Server

**Test connection**

- **Connection succeeded**

**Stored Procedures**

- Plain text password authentication - Not Configured
- Plain text password fetching - Not Configured
- Check username or machine exists - Not Configured
- Fetch groups - Not Configured
- Fetch attributes - Not Configured

**Step 3. Configure User Authentication**

ISE authentication to ODBC uses stored procedures. The **resultset** with this syntax:

<table>
<thead>
<tr>
<th>Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Result</td>
<td>Integer</td>
</tr>
<tr>
<td>Group (for compatibility with ACS 4.2 only)</td>
<td>Integer or varchar(255)</td>
</tr>
<tr>
<td>Account Info</td>
<td>varchar(255)</td>
</tr>
<tr>
<td>Error String</td>
<td>varchar(255)</td>
</tr>
</tbody>
</table>

For other procedures, refer to [Cisco Identity Services Engine 2.1 Administration Guide](#)

**Tip**: It is possible to return named parameters instead of resultset. It is just a different type of output, functionality is the same.

1. Navigate to options and uncheck **Prevent saving change that require table re-creation** check box (optional):
2. Create the table. Make sure you set the identity settings on the **primary key**. To set user_id as
primary key, right click the column name:

Final SQL:

3. Run this query to insert one user:

4. Create a procedure for plain text password authentication (used for PAP, EAP-GTC inner method, TACACS):

5. Create a procedure for plain text password fetching (used for CHAP, MSCHAPv1/v2, EAP-MD5, LEAP, EAP-MSCHAPv2 inner method, TACACS)

6. Create a procedure for check username or machine exists

7. Test created procedures:
Test other procedures in the same way.

8. Configure procedures on ISE and save:

9. Create a simple authentication rule using ODBC and test it:
Step 4. Configure Group Retrieval

1. Create tables containing user groups and another used for many-to-many mapping:

2. Add groups and mappings, so that **ODBCUSER1** belongs to both groups:

3. Create group retrieval procedure:

4. Map it to **Fetch groups**:

**ODBC List > ISE_ODBC**

**ODBC Identity Source**

<table>
<thead>
<tr>
<th>Stored procedure type</th>
<th>Returns recordset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plain text password authentication</td>
<td>ISEAuthUserPlainReturnsRecordset</td>
</tr>
<tr>
<td>Plain text password fetching</td>
<td>ISEFetchPasswordReturnsRecordset</td>
</tr>
<tr>
<td>Check username or machine exists</td>
<td>ISEUserLookupReturnsRecordset</td>
</tr>
</tbody>
</table>

5. Fetch the groups and add them into the **ODBC Identity Source**:

Step 5. Configure Attributes Retrieval

1. In order to simplify this example, a flat table is used for attributes:

2. Create an attribute for one of the users:

3. Create a **stored procedure**:

**ODBC List > ISE_ODBC**

**ODBC Identity Source**

<table>
<thead>
<tr>
<th>Stored procedure type</th>
<th>Returns recordset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check username or machine exists</td>
<td>ISEUserLookupReturnsRecordset</td>
</tr>
</tbody>
</table>

4. Add another user which does not belong to any group:

5. Create a specific **Policy Set** and test:
4. Map it to Fetch attributes:

ODBC List > ISE_ODBC

ODBC Identity Source

<table>
<thead>
<tr>
<th>General</th>
<th>Connection</th>
<th>Stored Procedures</th>
<th>Attributes</th>
<th>Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stored procedure type</td>
<td>Returns recordset</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plain text password authentication</td>
<td>ISEAuthUserPlainReturnsRecordset</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plain text password fetching</td>
<td>ISEFetchPasswordReturnsRecordset</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Check username or machine exists</td>
<td>ISEUserLookupReturnsRecordset</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fetch groups</td>
<td>ISEGroupsRetrieval</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fetch attributes</td>
<td>ISEAttrsRetrieval</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Search for MAC Address in format</td>
<td>xx-xx-xx-xx-xx-xx</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5. Fetch the attributes:

Select Attributes from ODBC

Sample User or Machine: odbcuser2

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Default Value</th>
<th>Name in ISE</th>
</tr>
</thead>
<tbody>
<tr>
<td>AwsomenessLevel</td>
<td>STRING</td>
<td>100</td>
<td>AwsomenessLevel</td>
</tr>
<tr>
<td>UserType</td>
<td>STRING</td>
<td>admin</td>
<td>UserType</td>
</tr>
</tbody>
</table>

OK Cancel

6. Adjust ISE rules:
Troubleshoot

If the connection is not successful, check windows event log. On ISE use command `show logging application prrt-management.log tail` while attempting to connect.

Example of bad authentication mode:

![Event Viewer - Bad Authentication Mode](image1)

Example of user missing permissions to open database:

![Event Viewer - User Permissions](image2)
In order to troubleshoot DB operations, enable logging components `odbc-id-store` to DEBUG level under Administration > System > Logging > Debug Log Configuration.

Logs are placed in `prrt-management.log` file.

Example for `odbuser2`: 