

ODBC を使って MS SQL の ISE 2.1 を設定する

目次

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[設定](#)

[ステップ 1：MS SQL の基本設定](#)

[ステップ 2：ISE の基本設定](#)

[ステップ 3：ユーザ認証の設定](#)

[ステップ 4：グループ取得の設定](#)

[ステップ 5：属性取得の設定](#)

[トラブルシューティング](#)

概要

このドキュメントでは、Open Database Connectivity (ODBC) を使用して Identity Services Engine (ISE) 認証を行うための Microsoft Standard Query Language (SQL) サーバでの ISE の設定方法について説明します。

注: Open Database Connectivity (ODBC) 認証では、ISE がプレーン テキストのパスワードを取得できることが必要です。データベース内でパスワードを暗号化できますが、ストア プロシージャで復号する必要があります。

前提条件

要件

次の項目に関する知識が推奨されます。

- データベースと ODBC の概念
- Microsoft SQL Server

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づくものです。

- Identity Services Engine 2.1
- MSSQL Server 2008 R2

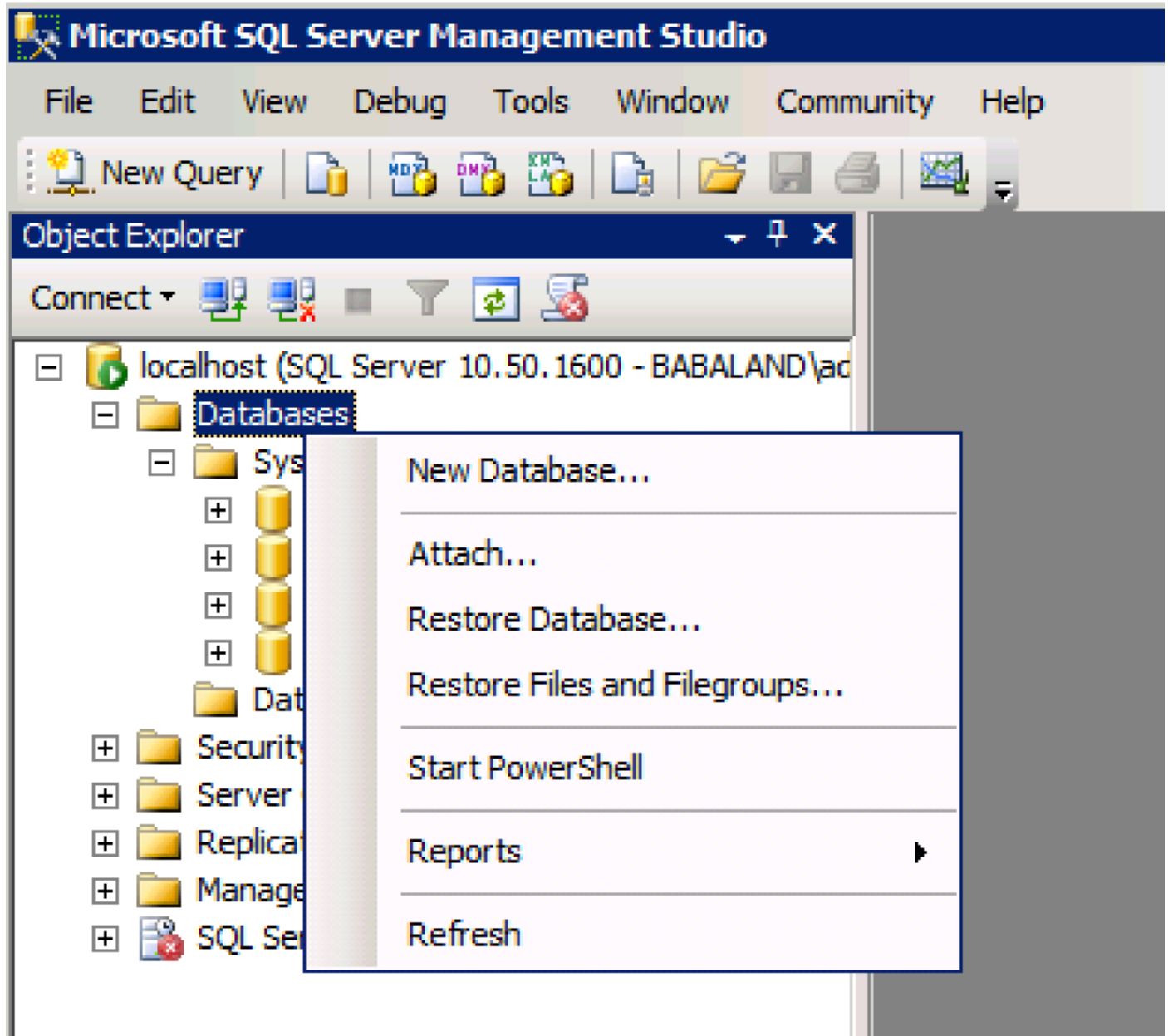
設定

ステップ 1 : MS SQL の基本設定

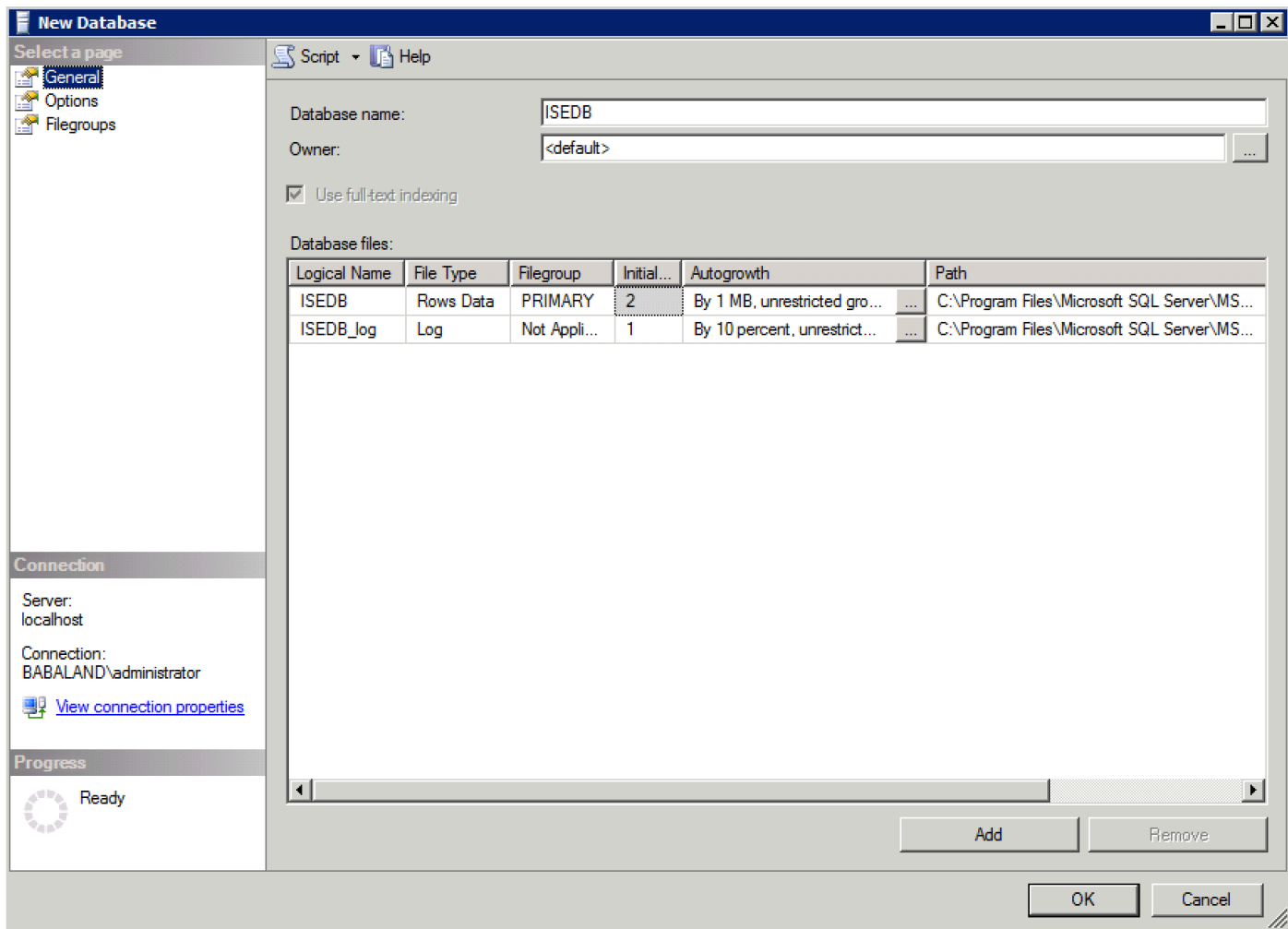
設定手順には、データベースと、そのデータベースにアクセスする権限を持つ 1 つの ISE ユーザの作成が含まれます。

注: ISE は SQL サポートします。Windows アカウントはサポートしません。認証モードを変更する必要がある場合は、「[サーバ認証モードの変更](#)」を参照してください。

1. SQL Server Management Studio を開き ([Start] メニュー [Microsoft SQL Server 2008 R2])、データベースを作成します。



2. デフォルト オプションをそのままにするか、次のイメージに示すようにデータベース設定を調整します。



3. 次のイメージに示すように、ユーザを作成し、権限を設定します。

Microsoft SQL Server Management Studio

File Edit View Debug Tools Window Community



Object Explorer



- [-] localhost (SQL Server 10.50.1600 - BABALAND\ad...)
 - + Databases
 - [-] Security
 - [-] Logins
 - [New Login...]
 - Filter
 - Start PowerShell
 - Reports
 - Refresh
 - + Servers
 - + Credentials

New Login...

Filter

Start PowerShell

Reports

Refresh

Login - New [Minimize] [Maximize] [Close]

Select a page

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

Script Help

Login name: Search...

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 Add

Mapped Credentials

Credential	Provider

Remove

Default database:

Default language:

OK Cancel

Connection

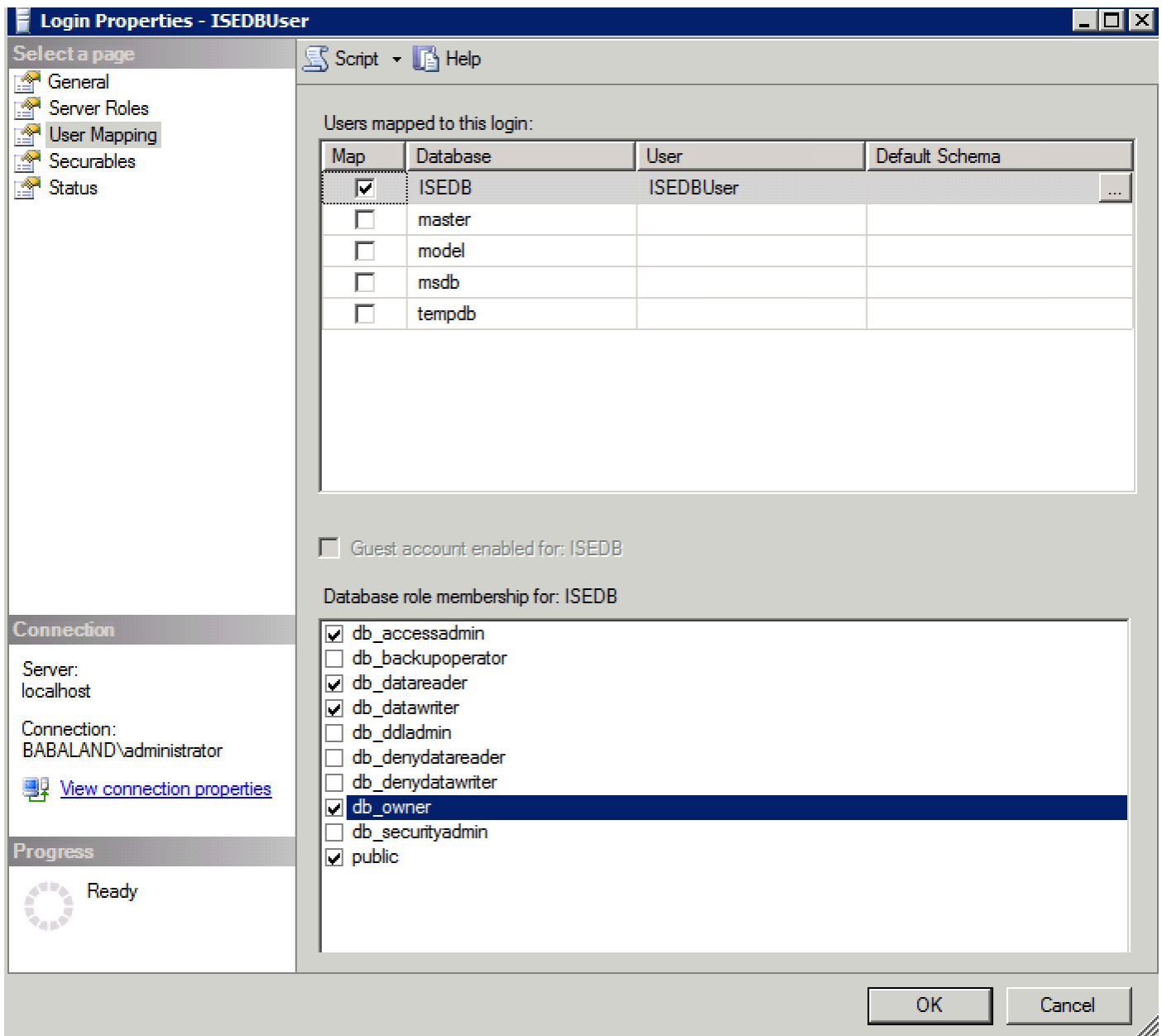
Server: localhost

Connection: BABALAND\Administrator

[View connection properties](#)

Progress

Ready



ステップ 2 : ISE の基本設定

[Administration] > [External Identity Source] > [ODBC] で ODBC Identity Source を作成し、接続をテストします。

ODBC Identity Source

General

Connection

Stored Procedures

Attributes

Groups

ODBC DB connection details

* Hostname/IP[:port] * Database name Admin username ⓘAdmin password * Timeout * Retries * Database type

Test connection

X

 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

ステップ 3 : ユーザ認証の設定

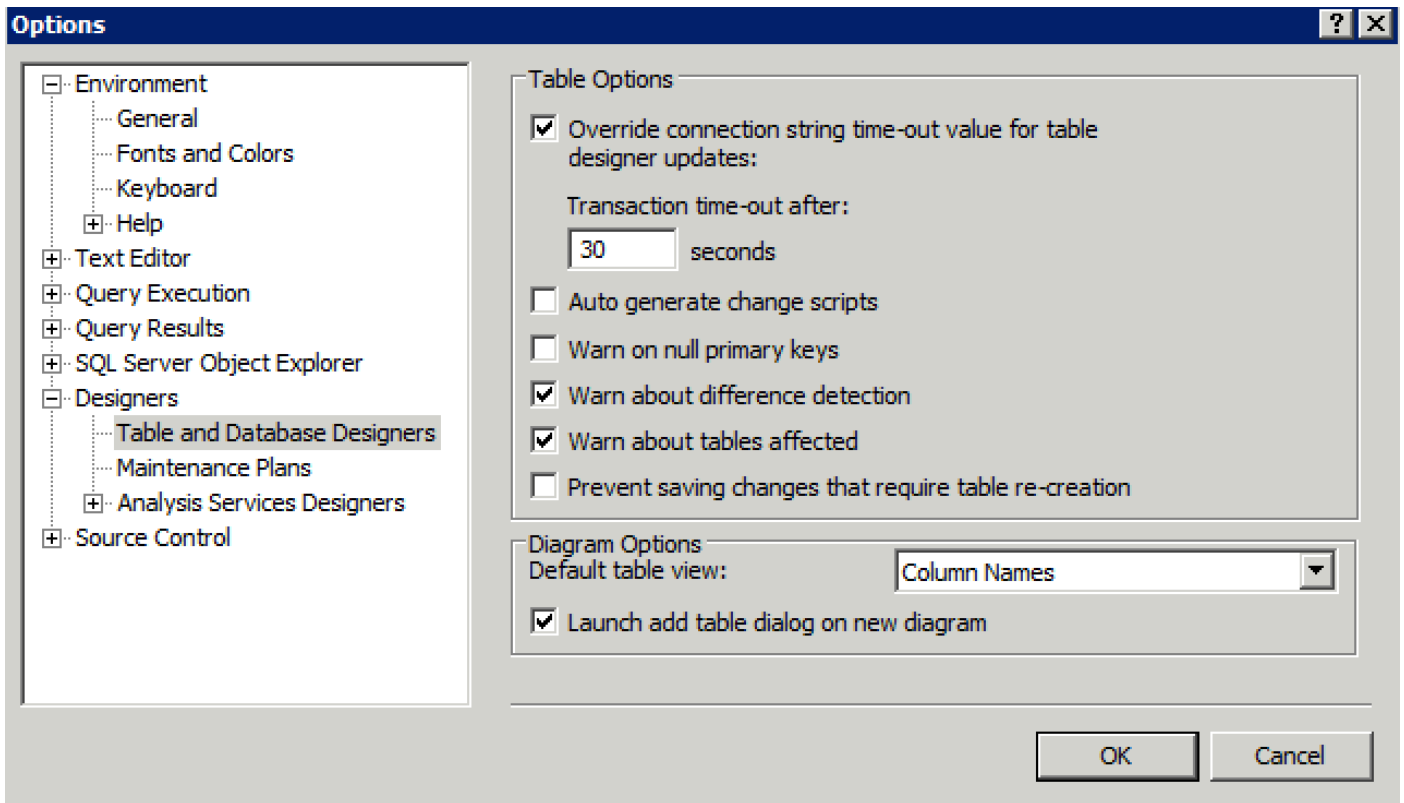
ODBC の ISE 認証では、ストアド プロシージャを使用します。認証のストアド プロシージャは次のシンタックスで **resultset** を返します。

値	タイプ
結果	Integer
グループ (ACS 4.2 のみと互換)	整数または varchar(255)
アカウント情報	varchar(255)
エラー文字列	varchar(255)

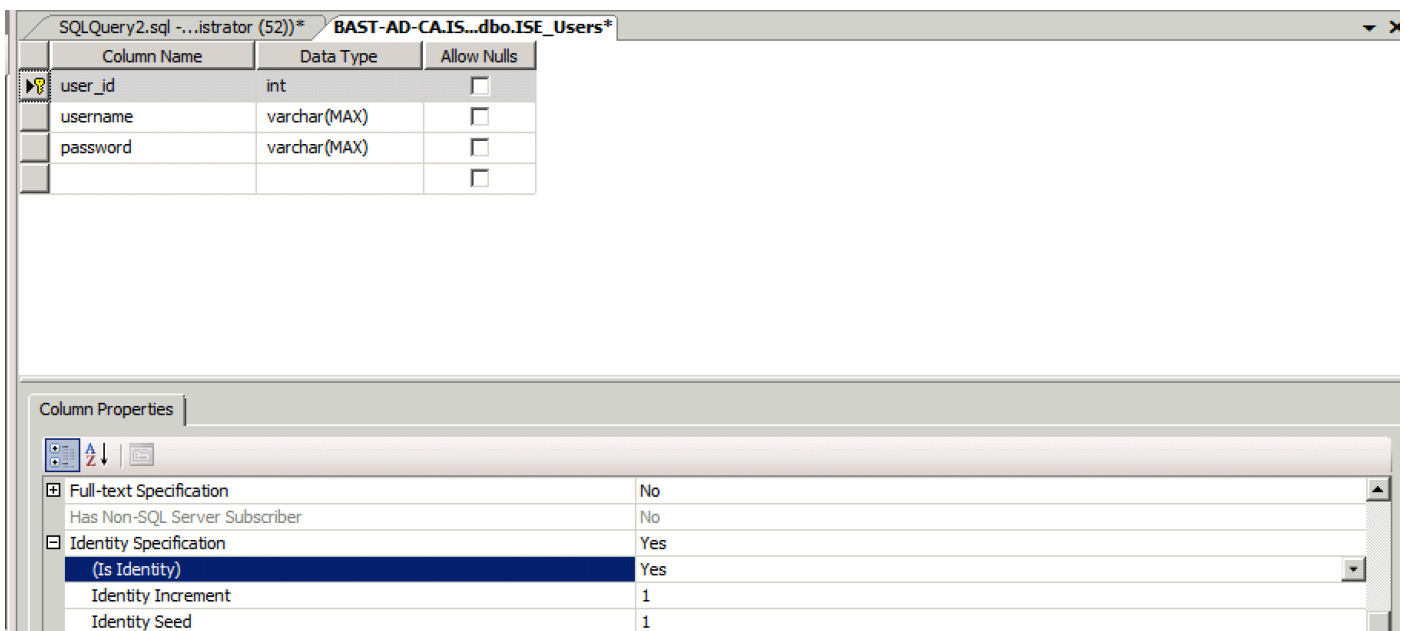
他の手順については、『[Cisco Identity Services Engine 2.1 アドミニストレーションガイド](#)』を参照してください。

ヒント : resultset の代わりに名前付きパラメータが返されることがあります。これは別のタイプの出力ですが、機能は同じです。

1. オプションに移動し、[Prevent saving change that require table re-creation] チェックボックスをオフにします (オプション)。



2. テーブルを作成します。プライマリ キーに ID 設定が行われていることを確認します。
user_id をプライマリ キーとして設定するには、[column name] を右クリックします。



最後の SQL :

```
CREATE TABLE [dbo].[ISE_Users](
[user_id] [int] IDENTITY(1,1) NOT NULL,
[username] [varchar](max) NOT NULL,
[password] [varchar](max) NOT NULL,
CONSTRAINT [PK_ISE_Users] PRIMARY KEY CLUSTERED
```



```
(  
[user_id] ASC  
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS =  
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY]
```

3. 1人のユーザを挿入するには、このクエリーを実行します。

```
insert into ISE_Users(username,password) values('odbcuser1','odbcpass');
```

4. プレーン テキストのパスワード認証のプロシージャを作成します (PAP、EAP-GTC 内部メソッド、TACACS に使用)。

```
CREATE PROCEDURE [dbo].[ISEAuthUserPlainReturnsRecordset]  
@username varchar(255), @password varchar(255)  
AS  
BEGIN  
IF EXISTS( SELECT username  
FROM ISE_Users  
WHERE username = @username  
AND password = @password )  
SELECT 0,11,'This is a very good user, give him all access','No Error'  
FROM ISE_Users  
WHERE username = @username  
ELSE  
SELECT 3,0,'odbc','ODBC Authen Error'  
END
```

5. プレーン テキストのパスワードの取得プロシージャを作成します (CHAP、MSCHAPv1/v2、EAP-MD5、LEAP、EAP-MSCHAPv2 内部メソッド、TACACS に使用)。

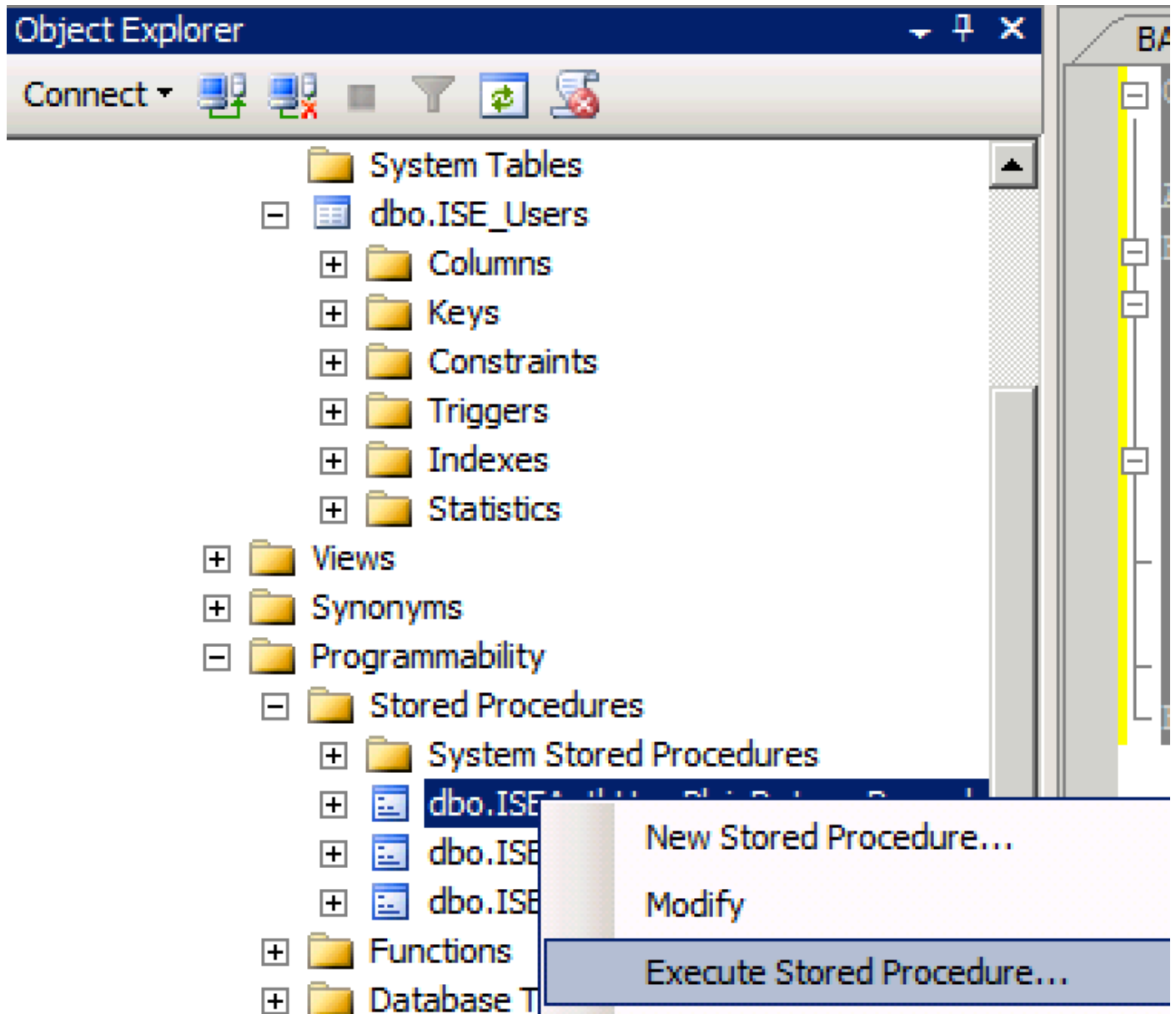
```
CREATE PROCEDURE [dbo].[ISEFetchPasswordReturnsRecordset]  
@username varchar(255)  
AS  
BEGIN  
IF EXISTS( SELECT username  
FROM ISE_Users  
WHERE username = @username)  
SELECT 0,11,'This is a very good user, give him all access','No Error',password  
FROM ISE_Users  
WHERE username = @username  
ELSE  
SELECT 3,0,'odbc','ODBC Authen Error'  
END
```

6. ユーザ名またはマシンの存在を確認するプロシージャを作成します (MAB、REAP の高速接続、EAP-FAST、および EAP-TTLS に使用)。

```
CREATE PROCEDURE [dbo].[ISEUserLookupReturnsRecordset]  
@username varchar(255)  
AS
```

```
BEGIN
IF EXISTS( SELECT username
FROM ISE_Users
WHERE username = @username)
SELECT 0,11,'This is a very good user, give him all access','No Error'
FROM ISE_Users
WHERE username = @username
ELSE
SELECT 3,0,'odbc','ODBC Authen Error'
END
```

7. 作成したプロシージャをテストします。



Execute Procedure - [dbo].[ISEAuthUserPlainReturnsRecordset]

Select a page: General

Script Help

Parameter	Data Type	Output Parameter	Pass Null Value	Value
@username	varchar(255)	No	<input type="checkbox"/>	odbcuser1
@password	varchar(255)	No	<input type="checkbox"/>	odbcpass

Connection

Server: localhost

Connection: BABALAND\administrator

[View connection properties](#)

Progress

Ready

OK Cancel

SQLQuery5.sql -...inistrator (57)) BAST-AD-CA.IS...dbo.ISE_Users SQLQuery2.sql -...istrator (52))* BAST-AD-CA.IS...dbo.ISE_Users

```

USE [ISEDB]
GO

DECLARE @return_value int

EXEC @return_value = [dbo].[ISEAuthUserPlainReturnsRecordset]
    @username = N'odbcuser1',
    @password = N'odbcpass'

SELECT 'Return Value' = @return_value

GO

```

Results Messages

	(No column name)	(No column name)	(No column name)	(No column name)
1	0	11	This is a very good user, give him all access	No Error

同じ方法で他のプロシージャをテストします。

8. ISE でプロシージャを設定し、保存します。

[ODBC List](#) > [ISE_ODBC](#)

ODBC Identity Source

General	Connection	Stored Procedures	Attributes	Groups
Stored procedure type		Returns recordset		
Plain text password authentication	ISEAuthUserPlainReturnsRecordset			
Plain text password fetching	ISEFetchPasswordReturnsRecordset			
Check username or machine exists	ISEUserLookupReturnsRecordset			
Fetch groups				
Fetch attributes				
Search for MAC Address in format		xx-xx-xx-xx-xx-xx		

9. ODBC を使用して簡単な認証ルールを作成し、テストします。

Authentication Policy

<input checked="" type="checkbox"/>	MAB	: If Wired_MAB OR Wireless_MAB	Allow Protocols : Default Network Access	and	Edit
<input checked="" type="checkbox"/>	Default	:use Internal Endpoints			
<input checked="" type="checkbox"/>	Dot1X	: If Wired_802.1X OR Wireless_802.1X	Allow Protocols : Default Network Access	and	Edit
<input checked="" type="checkbox"/>	Default	:use All_User_ID_Stores			
<input checked="" type="checkbox"/>	test_aaa	: If Radius:Service-Type EQUALS Login	Allow Protocols : Default Network Access	and	Edit
<input checked="" type="checkbox"/>	Default	:use ISE_ODBC			

```
b3560#test aaa group ISE236 odbcuser1 odbcpass legacy
Attempting authentication test to server-group ISE236 using radius
User was successfully authenticated.
```

Overview

Event	5200 Authentication succeeded
Username	odbcuser1
Endpoint Id	
Endpoint Profile	
Authentication Policy	Default >> test_aaa >> Default
Authorization Policy	Default >> Default
Authorization Result	PermitAccess

Authentication Details

Source Timestamp	2016-06-08 11:04:07.004
Received Timestamp	2016-06-08 11:04:07.005
Policy Server	bise236
Event	5200 Authentication succeeded
Username	odbcuser1
Authentication Identity Store	ISE_ODBC

Steps

```
11001 Received RADIUS Access-Request
11017 RADIUS created a new session
11117 Generated a new session ID for a 3rd party NAD
15049 Evaluating Policy Group
15008 Evaluating Service Selection Policy
15048 Queried PIP - Radius.NAS-Port-Type
15048 Queried PIP - Normalised Radius.RadiusFlowType (4 times)
15048 Queried PIP - Radius.Service-Type
15004 Matched rule - test_aaa
15041 Evaluating Identity Policy
15006 Matched Default Rule
15013 Selected Identity Source - ISE_ODBC
24852 Perform plain text password authentication in external ODBC database - ISE_ODBC
24849 Connecting to external ODBC database - ISE_ODBC
24850 Successfully connected to external ODBC database - ISE_ODBC
24855 Expect external ODBC database stored procedure to return results in a recordset - ISE_ODBC
22037 Authentication Passed
15036 Evaluating Authorization Policy
15048 Queried PIP - Radius.User-Name
15048 Queried PIP - Network Access.UseCase
15048 Queried PIP - Normalised Radius.RadiusFlowType (5 times)
15004 Matched rule - Default
```

ステップ4 : グループ取得の設定

1. ユーザグループグループが含まれているテーブルと、多対多マッピングに使用する別のテーブルを作成します。

```
CREATE TABLE [dbo].[Groups](
  [Group_ID] [int] IDENTITY(1,1) NOT NULL,
  [Group_Name] [varchar](max) NOT NULL,
  [Group_Desc] [text] NOT NULL,
  CONSTRAINT [PK_Groups] PRIMARY KEY CLUSTERED
  (
    [Group_ID] ASC
  )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

```
CREATE TABLE [dbo].[User_Groups_Mapping](
  [user_id] [int] NOT NULL,
  [group_id] [int] NOT NULL
) ON [PRIMARY]
```

```
ALTER TABLE dbo.User_Groups_Mapping ADD CONSTRAINT
FK_User_Groups_Mapping_Groups FOREIGN KEY
(
  group_id
) REFERENCES dbo.Groups
(
  Group_ID
) ON UPDATE CASCADE
ON DELETE CASCADE
```

```
GO
ALTER TABLE dbo.User_Groups_Mapping ADD CONSTRAINT
FK_User_Groups_Mapping_ISE_Users FOREIGN KEY
(
```

```

user_id
) REFERENCES dbo.ISE_Users
(
user_id
) ON UPDATE CASCADE
ON DELETE CASCADE

```

2. ODBCUSER1 が両方のグループに属するように、グループとマッピングを追加します。

```

INSERT [dbo].[Groups] ([Group_ID], [Group_Name], [Group_Desc]) VALUES (1, N'ODBCGroup1', N'My
Nice Group1')
INSERT [dbo].[User_Groups_Mapping] ([user_id], [group_id]) VALUES (1, 1)
INSERT [dbo].[Groups] ([Group_ID], [Group_Name], [Group_Desc]) VALUES (2, N'ODBCGroup2', N'My
Nice Group2')
INSERT [dbo].[User_Groups_Mapping] ([user_id], [group_id]) VALUES (1, 2)

```

3. グループ取得プロシージャを作成します。

```

CREATE PROCEDURE [dbo].[ISEGroupsRetrieval]
@username varchar(255), @result int output
AS
BEGIN
if exists (select * from ISE_Users where username = @username)
begin
set @result = 0
select Group_Name from Groups where group_id in (select group_ID from User_Groups_Mapping where
User_Groups_Mapping.USER_ID IN (select USER_ID from ISE_Users where username=@username ) )
end
else
set @result = 1
END

```

4. そのプロシージャを [Fetch groups] にマップします。

[ODBC List > ISE_ODBC](#)

ODBC Identity Source

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

5. グループを取得し、それらを [ODBC Identity Source] に追加します。

ODBC Identity Source

General Connection Stored Procedures Attributes **Groups**

Edit + Add - Delete

Name	Name in ISE
No data available	

Select Groups from ODBC X

Sample User or Machine Retrieve Groups

Name	Name in ISE
<input checked="" type="checkbox"/> ODBCGroup1	ODBCGroup1
<input checked="" type="checkbox"/> ODBCGroup2	ODBCGroup2

OK Cancel

6. どのグループにも属していない別のユーザを追加します。

```
insert into ISE_Users(username,password) values('odbcuser2','odbcpass');
```

7. 特定のポリシーセットを作成し、テストします。

Policy Sets Profiling Posture Client Provisioning Policy Elements

Policy Sets

Search policy names & descriptions.

Summary of Policies

Global Exceptions

TestAAA

VPN

Default

Save Order Reset Order

Define the Policy Sets by configuring rules based on conditions. Drag and drop sets on the left hand side to change the order.

For Policy Export go to Administration > System > Backup & Restore > Policy Export Page

Status	Name	Description	Conditions	Permissions
<input checked="" type="checkbox"/>	TestAAA		Radius:Service-Type EQUALS Login	

Authentication Policy

<input checked="" type="checkbox"/>	Default Rule (if no match)	Allow Protocols : Default Network Access	and use : ISE_ODBC	
-------------------------------------	----------------------------	--	--------------------	--

Authorization Policy

Exceptions (0)

Standard

Status	Rule Name	Conditions (identity groups and other conditions)	Permissions
<input checked="" type="checkbox"/>	Group1Access	if ISE_ODBC:ExternalGroups EQUALS ODBCGroup1	then PermitAccess
<input checked="" type="checkbox"/>	Default	if no matches, then	DenyAccess

```
b3560#test aaa group ISE236 odbcuser2 odhcpass legacy
Attempting authentication test to server-group ISE236 using radius
User authentication request was rejected by server.
```

```
b3560#test aaa group ISE236 odbcuser1 odhcpass legacy
Attempting authentication test to server-group ISE236 using radius
User was successfully authenticated.
```

AuthorizationPolicyMatchedRule	Group1Access
CPMSessionID	0a3027eci0HeVTM3/bn5vLXkWMcJ0em5rzUDaOSnbMmAvL7jcfY
ISEPolicySetName	TestAAA
AllowedProtocolMatchedRule	Default
IdentitySelectionMatchedRule	Default
Network Device Profile	Cisco
Location	Location#All Locations
Device Type	Device Type#All Device Types
ExternalGroups	ODBCGroup1
ExternalGroups	ODBCGroup2
RADIUS Username	odbcuser1

ステップ 5 : 属性取得の設定

1. この例を簡略化するため、属性にフラット テーブルを使用しています。

```
CREATE TABLE [dbo].[User_Attributes](
[user_id] [int] NOT NULL,
[Attribute_Name] [varchar](max) NOT NULL,
[Attribute_Value] [varchar](max) NOT NULL
) ON [PRIMARY]
```

GO

```
ALTER TABLE [dbo].[User_Attributes] WITH CHECK ADD CONSTRAINT [FK_User_Attributes_ISE_Users]
FOREIGN KEY([user_id])
REFERENCES [dbo].[ISE_Users] ([user_id])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
```

2. ユーザの 1 人の属性を作成します。

```
INSERT [dbo].[User_Attributes] ([user_id], [Attribute_Name], [Attribute_Value]) VALUES (2,
N'AwsomenessLevel', N'100')
INSERT [dbo].[User_Attributes] ([user_id], [Attribute_Name], [Attribute_Value]) VALUES (2,
N'UserType', N'admin')
```

3. ストアド プロシージャを作成します。

```
CREATE PROCEDURE [dbo].[ISEAttrRetrieval]
@username varchar(255), @result int output
AS
BEGIN
if exists (select * from ISE_Users where username = @username)
begin
```



```

set @result = 0
select attribute_name , attribute_value from user_attributes where USER_ID in(SELECT USER_ID
from ISE_Users where username = @username)
end
else
set @result = 1
END

```

4. そのプロシージャを [Fetch attributes] にマップします。

[ODBC List > ISE_ODBC](#)

ODBC Identity Source

General Connection **Stored Procedures** Attributes Groups

Stored procedure type Returns recordset ▼

Plain text password authentication ISEAuthUserPlainReturnsRecordset ⓘ ⊕

Plain text password fetching ISEFetchPasswordReturnsRecordset ⓘ ⊕

Check username or machine exists ISEUserLookupReturnsRecordset ⓘ ⊕

Fetch groups ISEGroupsRetrieval ⓘ ⊕

Fetch attributes ISEAttrsRetrieval ⓘ ⊕

Search for MAC Address in format xx-xx-xx-xx-xx-xx ▼ ⓘ

5. 属性を取得します。

Select Attributes from ODBC X

Sample User or Machine ⓘ

<input type="checkbox"/>	Name	Type ▲	Default Value	Name in ISE
<input type="checkbox"/>	AwsomenessLevel	STRING	100	AwsomenessLevel
<input type="checkbox"/>	UserType	STRING	admin	UserType

6. ISE ルールを調整します。

Status	Rule Name	Conditions (identity groups and other conditions)	Permissions	
✓	Group1Access	if ISE_ODBC:ExternalGroups EQUALS ODBCGroup1	then PermitAccess	Edit ▾
✓	AwesomeUser	if ISE_ODBC:AwesomenessLevel EQUALS 100	then PermitAccess	Edit ▾
✓	Default	if no matches, then	DenyAccess	Edit ▾

Refresh Reset Repeat Counts Export To Filter

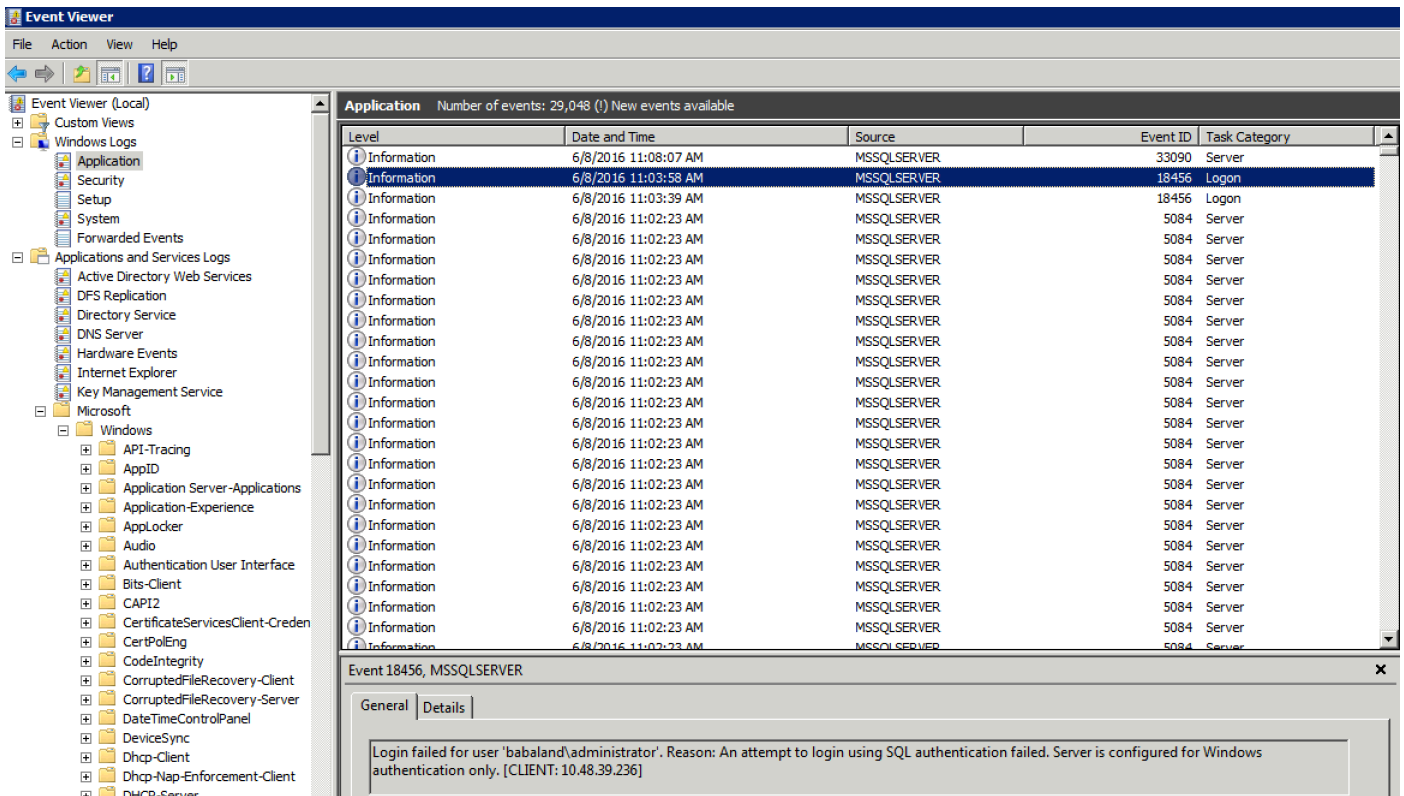
Time	Status	Details	Repeat ...	Identity	Endpoint ID	Endpoint Pr...	Authenticat...	Authorization Policy	Authorizatio...
Jun 08, 2016 12:21:45.596 PM	✓			Identity	Endpoint ID	Endpoint Prof	Authenticator	Authorization Policy	Authorization
				odbcuser2			TestAAA >> ...	TestAAA >> AwesomeUser	PermitAccess

トラブルシューティング

接続がうまくいかない場合、Windows のイベント ログを確認します。接続を試行している間に ISE で `show logging application prrt-management.log tail` コマンドを使用します。

不正な認証モードの例：

```
bise236/admin# sh logg app prrt-management.log tail
2016-06-08 09:03:59,822 WARN [admin-http-pool1177][]
cisco.cpm.odbcidstore.impl.MSSQLServerDbAccess -:bastien::- Connection to ODBC DB failed.
Exception: com.microsoft.sqlserver.jdbc.S
QLServerException: Login failed for user 'babaland\administrator'. ClientConnectionId:c74ade15-
4f34-415a-9a94-4c54c58c0fc3
com.microsoft.sqlserver.jdbc.SQLServerException: Login failed for user 'babaland\administrator'.
ClientConnectionId:c74ade15-4f34-415a-9a94-4c54c58c0fc3
at
com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:21
6)
at com.microsoft.sqlserver.jdbc.TDSTokenHandler.onEOF(tdsparser.java:254)
at com.microsoft.sqlserver.jdbc.TDSParser.parse(tdsparser.java:84)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.sendLogon(SQLServerConnection.java:2908)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.logon(SQLServerConnection.java:2234)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.access$000(SQLServerConnection.java:41)
at
com.microsoft.sqlserver.jdbc.SQLServerConnection$LogonCommand.doExecute(SQLServerConnection.java
:2220)
at com.microsoft.sqlserver.jdbc.TDSCommand.execute(IOBuffer.java:5696)
at
com.microsoft.sqlserver.jdbc.SQLServerConnection.executeCommand(SQLServerConnection.java:1715)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.connectHelper(SQLServerConnection.java:1326)
```

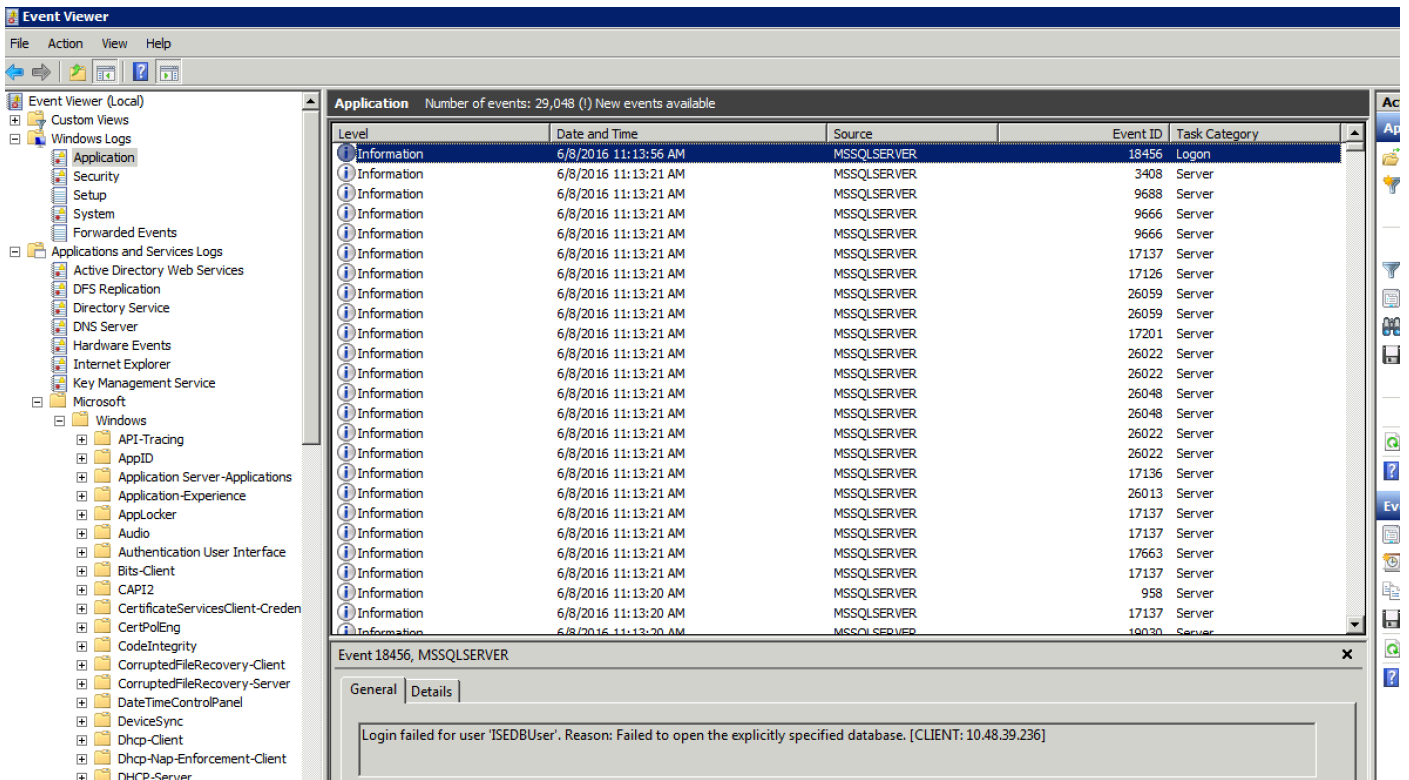


データベースを開く権限を失っているユーザの例：

```

2016-06-08 09:13:57,842 WARN [admin-http-pool159][
cisco.cpm.odbcidstore.impl.MSSQLServerDbAccess -:bastien::- Connection to ODBC DB failed.
Exception: com.microsoft.sqlserver.jdbc.SQLServerException: Cannot open database "ISEDB"
requested by the login. The login failed. ClientConnectionId:299c2956-6946-4282-b3ca-
2aa86642a821
com.microsoft.sqlserver.jdbc.SQLServerException: Cannot open database "ISEDB" requested by the
login. The login failed. ClientConnectionId:299c2956-6946-4282-b3ca-2aa86642a821
at
com.microsoft.sqlserver.jdbc.SQLServerException.makeFromDatabaseError(SQLServerException.java:21
6)
at com.microsoft.sqlserver.jdbc.TDSTokenHandler.onEOF(tdsparser.java:254)
at com.microsoft.sqlserver.jdbc.TDSParser.parse(tdsparser.java:84)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.sendLogon(SQLServerConnection.java:2908)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.logon(SQLServerConnection.java:2234)
at com.microsoft.sqlserver.jdbc.SQLServerConnection.access$000(SQLServerConnection.java:41)

```



DB 操作をトラブルシューティングするには、[Administration]> [System] > [Logging] > [Debug Log Configuration] で、ロギング コンポーネント **odbc-id-store** を DEBUG レベルに有効にします

。

ログは prrt-management.log ファイルに配置されます。

odbcuser2 の例 :

```

2016-06-08 12:26:56,009 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC
ID Store Operation: Authenticate Plain Text Password. Username=odbcuser2,
SessionID=0a3027ecLA_rJLKsS5QAzurVluGWzdYe67rIgcG3MMQcpE8yKnw
2016-06-08 12:26:56,012 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.CustomerLog -:::- Write
customer log message: 24852
2016-06-08 12:26:56,012 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -
:::- OdbcConnectionPool - get connection
2016-06-08 12:26:56,012 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -
:::- OdbcConnectionPool - use existing connection
2016-06-08 12:26:56,013 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -
:::- OdbcConnectionPool - connections in use: 1
2016-06-08 12:26:56,013 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Authenticate plain text password
2016-06-08 12:26:56,013 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Prepare stored procedure call, procname=ISEAuthUserPlainReturnsRecordset
2016-06-08 12:26:56,013 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Using recordset to obtain stored procedure result values
2016-06-08 12:26:56,013 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.CustomerLog -:::- Write
customer log message: 24855
2016-06-08 12:26:56,013 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Text: {call ISEAuthUserPlainReturnsRecordset(?, ?)}
2016-06-08 12:26:56,013 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Setup stored procedure input parameters, username=odbcuser2, password=***
2016-06-08 12:26:56,014 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Execute stored procedure call

```

```
2016-06-08 12:26:56,017 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Process stored procedure results
2016-06-08 12:26:56,017 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Obtain stored procedure results from recordset
2016-06-08 12:26:56,017 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Received result recordset, number of columns=4
2016-06-08 12:26:56,017 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Results successfully parsed from recordset
2016-06-08 12:26:56,018 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -
:::- OdbcConnectionPool - release connection
2016-06-08 12:26:56,018 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -
:::- OdbcConnectionPool - connections in use: 0
2016-06-08 12:26:56,018 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- Call
to ODBC DB succeeded
2016-06-08 12:26:56,018 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.OdbcAuthResult -:::-
Authentication result: code=0, Connection succeeded=false, odbcDbErrorString=No Error,
odbcStoredProcedureCustomerErrorString=null, accountInfo=This is a very good user, give him all
access, group=11
2016-06-08 12:26:56,019 DEBUG [Thread-4051][] cisco.cpm.odbcidstore.impl.CustomerLog -:::- Write
customer log message: 24853
2016-06-08 12:26:56,026 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC ID
Store Operation: Get all user groups. Username=odbcuser2,
SessionID=0a3027ecLA_rJLKsS5QAzurVluGWzdYe67rIgcG3MMQcpE8yKnw
2016-06-08 12:26:56,029 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC ID
Store Operation: Fetch user groups. Username=odbcuser2,
SessionID=0a3027ecLA_rJLKsS5QAzurVluGWzdYe67rIgcG3MMQcpE8yKnw
2016-06-08 12:26:56,029 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.CustomerLog -:::- Write
customer log message: 24869
2016-06-08 12:26:56,029 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:::-
OdbcConnectionPool - get connection
2016-06-08 12:26:56,029 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:::-
OdbcConnectionPool - use existing connection
2016-06-08 12:26:56,029 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:::-
OdbcConnectionPool - connections in use: 1
2016-06-08 12:26:56,029 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Fetch user groups
2016-06-08 12:26:56,029 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Prepare stored procedure call, procname=ISEGroupsRetrieval
2016-06-08 12:26:56,029 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Text: {call ISEGroupsRetrieval(?,?) }
2016-06-08 12:26:56,029 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Setup stored procedure input parameters, username=odbcuser2
2016-06-08 12:26:56,029 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Execute stored procedure call
2016-06-08 12:26:56,031 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Process stored procedure results
2016-06-08 12:26:56,032 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Received empty result set, no groups/attributes data can be obtained
2016-06-08 12:26:56,032 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:::-
Result code indicates success
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:::-
OdbcConnectionPool - release connection
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:::-
OdbcConnectionPool - connections in use: 0
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- Call to
ODBC DB succeeded
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.CustomerLog -:::- Write
customer log message: 24870
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC ID
Store Operation: Get all user groups. Got groups...
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC ID
Store Operation: Get all user groups. Username=odbcuser2, ExternalGroups=[]
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:::- ODBC ID
Store Operation: Fetch user attributes. Username=odbcuser2,
```

SessionID=0a3027ecLA_rJLKsS5QAzurVluGWzdYe67rIgcG3MMQcpE8yKnw
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.CustomerLog -:- Write customer log message: 24872
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:- OdbcConnectionPool - get connection
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:- OdbcConnectionPool - use existing connection
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:- OdbcConnectionPool - connections in use: 1
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Fetch user attributes
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Prepare stored procedure call, procname=ISEAttrsRetrieval
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Text: {call ISEAttrsRetrieval(?,?)}
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Setup stored procedure input parameters, username=odbcuser2
2016-06-08 12:26:56,033 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Execute stored procedure call
2016-06-08 12:26:56,035 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Process stored procedure results
2016-06-08 12:26:56,035 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Received result recordset, total number of columns=2
2016-06-08 12:26:56,035 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- According to column number expect multiple rows (vertical attributes/groups returned result)
2016-06-08 12:26:56,035 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Fetched data: AwsomenessLevel=100
2016-06-08 12:26:56,035 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Fetched data: UserType=admin
2016-06-08 12:26:56,035 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Results successfully parsed from recordset
2016-06-08 12:26:56,035 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnection -:- Result code indicates success
2016-06-08 12:26:56,036 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:- OdbcConnectionPool - release connection
2016-06-08 12:26:56,036 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcConnectionPool -:- OdbcConnectionPool - connections in use: 0
2016-06-08 12:26:56,036 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:- Call to ODBC DB succeeded
2016-06-08 12:26:56,036 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.CustomerLog -:- Write customer log message: 24873
2016-06-08 12:26:56,036 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:- ODBC ID Store Operation: Get all user attrs. Username=odbcuser2, Setting ISE_ODBC.AwsomenessLevel to 100
2016-06-08 12:26:56,036 DEBUG [Thread-84][] cisco.cpm.odbcidstore.impl.OdbcIdStore -:- ODBC ID Store Operation: Get all user attrs. Username=odbcuser2, Setting ISE_ODBC.UserType to admin