BUSINESS OBJECTS

Crystal Reports XI Release 2

Using Integrated and SQL Authentication in .NET Applications

Overview

This technical document describes how to use integrated and SQL authentication in applications that use the Crystal Reports .NET SDK. Specifically, seven scenarios are discussed that involve changing authentication types, databases, and/or servers.

The code samples provided in this document apply to Crystal Reports XI (with Service Pack 1), Crystal Reports XI Release 2, and the bundled version of Crystal Reports in Visual Studio .NET 2005.

Contents

Introduction

A Crystal report may be designed to use either integrated or SQL authentication. The seven scenarios described in this technical document discuss how to change the authentication type, database, and/or server in your Crystal Reports .NET application. The code samples provided in this document apply to Crystal Reports XI (with Service Pack 1), Crystal Reports XI Release 2, and the bundled version of Crystal Reports in Visual Studio .NET 2005.

IMPORTANT	This document applies only to the scenarios where the application and the database server reside on the same computer.

Integrated and SQL authentication scenarios

When a report is created in the Crystal Reports Designer with a particular authentication, it will use that authentication at runtime, unless the authentication is changed. There are five basic scenarios that are considered and explained below

Scenario 1 – Changing from SQL to integrated authentication at runtime and using same data source

A Crystal report is designed to use SQL authentication and is connecting to the original data source. Use one of the following code samples to change the report to use integrated authentication:

Code Sample 1

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load


    Dim report As New SQLReport_LocalDBrpt


    'Using Crystal Reports XI, or XI r2, the following line
    returns 'False'

    MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
    ecurity.ToString())


    'Setting IntegratedSecurity = True will allow SSO and
    will not prompt

    report.DataSourceConnections(0).IntegratedSecurity =
    True


    'The following line will return 'True'

    MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
    ecurity.ToString())
```

```
    CrystalReportViewer1.ReportSource = report

End Sub
```

Code Sample 2

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load

    Dim report As New SQLReport_LocalDBrpt()

    'Using Crystal Reports XI, or XI r2, the following line
    returns 'False'
    MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
    ecurity.ToString())

    'Setting the third parameter to True will allow SSO and
    will not prompt
    report.DataSourceConnections(0).SetConnection("dbServer"
    , "Northwind", True)

    'The following line will return 'True'
    MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
    ecurity.ToString())
    CrystalReportViewer1.ReportSource = report

End Sub
```

Code Sample 3

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load

    Dim report As New SQLReport_LocalDBrpt.Load()

    'Using Crystal Reports XI, or XI r2, the following line
    returns 'False' even though Integrated Security is true
    in the 'Report. Crystal Reports 10.2 returns  'True'
    MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
    ecurity.ToString())
    report.Database.Tables(0).LogOnInfo.ConnectionInfo.Serve
    rName = "dbServer"
    report.Database.Tables(0).LogOnInfo.ConnectionInfo.Datab
    aseName = "Northwind"
```

```
        'Setting IntegratedSecurity = True will allow SSO and
        will not prompt
        report.Database.Tables(0).LogOnInfo.ConnectionInfo.Integ
        ratedSecurity = True
        Dim tLogonInfo As New
        CrystalDecisions.Shared.TableLogOnInfo
        tLogonInfo.ConnectionInfo =
        rpt.Database.Tables(0).LogOnInfo.ConnectionInfo
        report.Database.Tables(0).ApplyLogOnInfo(tLogonInfo)


        'The following line will return 'True'
        MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
        ecurity.ToString())


        CrystalReportViewer1.ReportSource = report

    End Sub
```

These code samples (without the message boxes) may also be used in web applications, with the addition of these steps:

1. Add this line of code to your Web.Config file:

   <identity impersonate=”true” />

2. Disable anonymous access in IIS, as follows:

   i. Go to **Start** > **Run** and then type “inetmgr” in the **Run** box.

   ii. Right-click your application virtual folder and then click Properties.

   iii. Click the **Directory Security** tab and then click the **Edit** button.

   iv. Clear the **Anonymous access** check box.

NOTE	If you are using Windows 2003, ensure that the application pool (SSOAppPool) associated with the web application is using the Network Service or Local System account.

Scenario 2 – Changing from SQL to integrated authentication at runtime and using a different data source

A Crystal report is designed to use SQL authentication and is connecting to a different data source. Use one of the following code samples to change the report to use integrated authentication:

Code Sample 1

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load


    Dim report As New SQLReport_LocalDBrpt


    'Both of the following lines must be used

    report.DataSourceConnections(0).IntegratedSecurity =
    True

    report.DataSourceConnections(0).SetConnection("myNewServ
    er", "Northwind", True)


    'The following line will now return ‘True’

    MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
    ecurity.ToString())

    CrystalReportViewer1.ReportSource = report


End Sub
```

Code Sample 2

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load


    Dim report As New SQLReport_LocalDBrpt()

    Dim tLogonInfo As New
    CrystalDecisions.Shared.TableLogOnInfo


    'Using Crystal Reports XI, or XI r2, the following line
    returns 'False'

    MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
    ecurity.ToString())

    report.DataSourceConnections(0).IntegratedSecurity =
    True

    report.Database.Tables(0).LogOnInfo.ConnectionInfo.Serve
    rName = "myNewServer"

    report.Database.Tables(0).LogOnInfo.ConnectionInfo.Datab
    aseName = "Northwind"
```

```
    report.Database.Tables(0).LogOnInfo.ConnectionInfo.Integ
    ratedSecurity = True

    tLogonInfo.ConnectionInfo =
    report.Database.Tables(0).LogOnInfo.ConnectionInfo

    report.Database.Tables(0).ApplyLogOnInfo(tLogonInfo)


    'The following line will return 'True'

    MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
    ecurity.ToString())

    CrystalReportViewer1.ReportSource = report

End Sub
```

These code samples (without the message boxes) may also be used in web applications, with the addition of these steps:

1. Add this line of code to your Web.Config file:

   <identity impersonate=”true” />

2. Disable anonymous access in IIS, as follows:

   i. Go to **Start** > **Run** and then type “inetmgr” in the **Run** box.

   ii. Right-click your application virtual folder and then click Properties.

   iii. Click the **Directory Security** tab and then click the **Edit** button.

   iv. Clear the **Anonymous access** check box.

NOTE	If you are using Windows 2003, ensure that the application pool (SSOAppPool) associated with the web application is using the Network Service or Local System account.

Scenario 3 – Changing from SQL to integrated authentication at runtime and connecting to original server

A Crystal report is designed to use SQL authentication and is connecting to the original server. In a Windows application no changes are required for this scenario; however, use the following code sample in your web application to change the report to use integrated authentication:

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load


    Dim report As New SSOReport_LocalDB
```

```
    'Using Crystal Reports XI, or XI r2, the following line
    returns 'False' even though Integrated Security is true
    in the 'Report. Crystal Reports 10.2 returns  'True'

    MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
    ecurity.ToString())


    'Setting IntegratedSecurity = True will allow SSO and
    will not prompt. This code must be ‘used even if the
    report was created with integrated authentication.

    report.DataSourceConnections(0).IntegratedSecurity =
    True


    'The following line will return 'True'

    MessageBox.Show(rpt.DataSourceConnections(0).IntegratedS
    ecurity.ToString())

    CrystalReportViewer1.ReportSource = report


End Sub
```

Scenario 4: Using integrated authentication connecting to different server but using same database name

A Crystal report is designed to use integrated authentication and is connecting to a different server but using the same database name. Use the following code sample in a web and Windows application:

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load


    Dim report As New SSOReport_LocalDB

    Dim MyDatabase As Database

    Dim MyTables As Tables

    Dim MyTable As Table

    Dim MyLogOnInfo As TableLogOnInfo

    Dim MyConnectionInfo As ConnectionInfo


    'Using Crystal Reports XI, or XI r2, the following line
    returns 'False' even though Integrated Security is true
    in the 'Report. Crystal Reports 10.2 returns  'True'

    MessageBox.Show(report.DataSourceConnections(0).Integrat
    edSecurity.ToString())


    'Setting IntegratedSecurity = True will allow SSO and
    will not prompt. This code must be ‘used even if the
    report was created with integrated authentication.
```

```
    report.DataSourceConnections(0).IntegratedSecurity =
    True

    'The following line will return 'True'
    MessageBox.Show(report.DataSourceConnections(0).Integrat
    edSecurity.ToString())

    MyConnectionInfo = New ConnectionInfo()
    With MyConnectionInfo
        .ServerName = "newServer"
    End With

    MyDatabase = Report.Database
    MyTables = MyDatabase.Tables

    For Each MyTable In MyTables
        MyLogOnInfo = MyTable.LogOnInfo
        MyLogOnInfo.ConnectionInfo = MyConnectionInfo
        MyTable.ApplyLogOnInfo(MyLogOnInfo)
    Next

    CrystalReportViewer1.ReportSource = report

End Sub
```

Scenario 5: Using integrated authentication and connecting to a different database but using the same server

A Crystal report is designed to use integrated authentication and is connecting to a different database but using the same database server. Use the following code sample in a web and Windows application:

```
Private Sub Page_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load

    Dim report New SSOReport_LocalDB
    Dim MyDatabase As Database
    Dim MyTables As Tables
    Dim MyTable As Table
    Dim MyLogOnInfo As TableLogOnInfo
    Dim MyConnectionInfo As ConnectionInfo
```

```
    'Setting IntegratedSecurity = True will allow SSO and
    will not prompt. This code must be used even if the
    ‘report was created with integrated authentication.
    report.DataSourceConnections(0).IntegratedSecurity =
    True


    MyConnectionInfo = New ConnectionInfo
    MyConnectionInfo =
    report.Database.Tables(0).LogOnInfo.ConnectionInfo


    With MyConnectionInfo
    .DatabaseName = "northwind"
    End With
    MyDatabase = report.Database
    MyTables = MyDatabase.Tables


    For Each MyTable In MyTables
        MyLogOnInfo = MyTable.LogOnInfo
        MyLogOnInfo.ConnectionInfo = MyConnectionInfo
        MyTable.ApplyLogOnInfo(MyLogOnInfo)
    Next


    CrystalReportViewer1.ReportSource = report

End Sub
```

Scenario 6: Changing to SQL authentication and connecting to the same or different data source

A Crystal report is designed to use integrated authentication. Use the following code sample to enable to the report to connect to the same or different data source.

```
Private Sub Page_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load


    Dim crDatabase As Database
    Dim crTables As Tables
    Dim crTable As Table
    Dim crtableLogOnInfo As TableLogOnInfo
    Dim crConnectionInfo As ConnectionInfo


    ' Create a new report
    crReportDocument = New CrystalReport1
```

```
        crConnectionInfo = New ConnectionInfo


        ' Set the connection information
        crReportDocument.DataSourceConnections(0).IntegratedSecu
        rity = False
        crConnectionInfo.ServerName = "dbServer"
        crConnectionInfo.DatabaseName = "northwind"
        crConnectionInfo.UserID = "dbUser"
        crConnectionInfo.Password = "dbPassword"


        ' Obtain a reference to the tables
        crDatabase = crReportDocument.Database
        crTables = crDatabase.Tables


        ' Loop through all the tables and set the logon
        information
        For Each crTable In crTables
            crtableLogOnInfo = crTable.LogOnInfo
            crtableLogOnInfo.ConnectionInfo = crConnectionInfo
            crTable.ApplyLogOnInfo(crtableLogOnInfo)


            ' This line is only necessary if you are changing the
            database location. It removes the database and owner
            ' from the location property and sets the location
            equal to just the table name
            crTable.Location =
            crTable.Location.Substring(crTable.Location.LastIndex
            Of(".") + 1)
        Next crTable


        CrystalReportViewer1.ReportSource = crReportDocument


    End Sub
```

Scenario 7: Using integrated authentication with Single Sign-On to a remote backend database server

A Crystal report uses integrated authentication to connect to a SQL Server database that is running on another server. The report resides on a computer that connects to a web server.

To use Single Sign-On (SSO) in your .NET web application to connect from the workstation to the web server to the SQL database, complete these steps:

1. Configure your system as described in the following Microsoft knowledge base article: http://www.microsoft.com/technet/prodtechnol/windowsserver2003/technologies/security/tkerbdel.mspx.

2. Add this line of code to your .NET application:

   Report.DataSourceConnections(0).IntegratedSecurity = True

NOTE	
	▪ All computers must be in the same Windows 2000 or Windows Server 2003 Active Directory forest. ▪ The service provider name must be created for the web service account. For Windows Server 2003, the **Delegation** tab appears in the Active Directory **Properties** dialog box. ▪ The service account must be set to be trusted for delegation.

Finding More Information

Reports connected through OLEDB to Microsoft Directory Services do not refresh

Changing from NT Authentication to SQL Authentication in .NET at Runtime

Err Msg: "Logon failed" when using NT authentication in an ASP.NET web application

For more information and resources, refer to the product documentation and visit the support area of the web site at www.businessobjects.com.

www.businessobjects.com