

SSRS

- [SSRS: Export Reports Without WebPortal](#)
- [SSRS Links](#)
- [SSRS Server Pages](#)
- [SSRS SSL Certificates](#)

SSRS: Export Reports Without WebPortal

If you're trying to export all the RDL, data sources, gif, and other files of a set of reports, but the SSRS web portal is inaccessible, this method will pull them from the report engine database back-end.

Taken from here: [Export of SSRS reports datasources and images](#)

```
<# .SYNOPSIS
    Export of SSRS reports datasources and images
.DESCRIPTION
    This PowerShell script exports all (or filtered) reports, data sources and images directly from the
    ReportServer database
    to a specified folder. For the file name the complete report path is used; for file name invalid characters are
    replaced with a -.
    Reports are exported with .rdl as extension, data sources with .rds and resources without any additional
    extension.
    Please change the "Configuration data" below to your enviroment.
    Works with SQL Server 2005 and higher versions in all editions.
    Requires SELECT permission on the ReportServer database.
.NOTES
    Author : Olaf Helper
    Requires: PowerShell Version 1.0, Ado.Net assembly
.LINK
    GetSqlBinary: http://msdn.microsoft.com/en-
    us/library/system.data.sqlclient.sqldatareader.getsqlbinary.aspx
#>
param(
    [Parameter(Mandatory=$true)] [string]$serverName, # server name and instance
    [Parameter(Mandatory=$true)] [string]$databaseName, # ReportServer Database.
    [Parameter()] [string]$outputDirectory = 'C:\Temp' # Path to export the reports to.
)

# Select-Statement for file name & blob data with filter.
```

```

$sql = "SELECT CT.[Path]
      ,CT.[Type]
      ,CONVERT(varbinary(max), CT.[Content]) AS BinaryContent
FROM dbo.[Catalog] AS CT
WHERE CT.[Type] IN (2, 3, 5)";

# Open ADO.NET Connection with Windows authentication.
$con = New-Object Data.SqlClient.SqlConnection;
$con.ConnectionString = "Data Source=$serverName;Initial Catalog=$databaseName;Integrated
Security=True;";
$con.Open();

Write-Output ((Get-Date -format yyyy-MM-dd-HH:mm:ss) + ": Started ...");

# New command and reader.
$cmd = New-Object Data.SqlClient.SqlCommand $sql, $con;
$rd = $cmd.ExecuteReader();

$invalids = [System.IO.Path]::GetInvalidFileNameChars();
# Looping through all selected datasets.
While ($rd.Read())
{
    Try
    {
        # Get the name and make it valid.
        $name = $rd.GetString(0);
        foreach ($invalid in $invalids)
        { $name = $name.Replace($invalid, "-"); }

        If ($rd.GetInt32(1) -eq 2)
        { $name = $name + ".rdl"; }
        Elseif ($rd.GetInt32(1) -eq 5)
        { $name = $name + ".rds"; }

        Write-Output ((Get-Date -format yyyy-MM-dd-HH:mm:ss) + ": Exporting {0}" -f $name);

        $name = [System.IO.Path]::Combine($outputDirectory, $name);

        # New BinaryWriter; existing file will be overwritten.
        $fs = New-Object System.IO.FileStream ($name), Create, Write;

```

```
$bw = New-Object System.IO.BinaryWriter($fs);

# Read of complete Blob with GetSqlBinary
$bt = $rd.GetSqlBinary(2).Value;
$bw.Write($bt, 0, $bt.Length);
$bw.Flush();
$bw.Close();
$fs.Close();
}
Catch
{
    Write-Output ($_.Exception.Message)
}
Finally
{
    $fs.Dispose();
}
}

# Closing & Disposing all objects
$rd.Close();
$cmd.Dispose();
$con.Close();
$con.Dispose();

Write-Output ((Get-Date -format yyyy-MM-dd-HH:mm:ss) + ": Finished");
```

SSRS Links

How to Customize the SSRS Report Viewer:

<https://github.com/MagnusJohansson/CustomSSRSReportViewer>

SSRS Server Pages

This is a bunch of rolling notes for SSRS.

Web Portal URL

The standard Web portal UI is here: <https://hostname/Reports>

Use this link to browse, manage, and view reports with the standard web interface.

Report Server URL

The Report Server is available here: <https://hostname/ReportServer>

Use this link to access the report server's backend, or in applications like Report Builder for deploying and managing reports.

Reporting Services Configuration Manager

This is a Windows executable, that provides high-level configuration for the report server.

It can be found in the Start menu, as *Reporting Services Configuration Manager*.

URL Reservations

Once configured in RSCM, URL reservations are stored in the RSReportServer.config file.

It is available here: "c:\Program Files\Microsoft SQL Server Reporting Services\SSRS\ReportServer\RSReportServer.config".

SSRS SSL Certificates

Here's some notes on how to add an SSL certificate to SSRS.

Certificate Conversion

When importing an SSL certificate (cer) with a separate private key (key), it needs to be in PFX format, for Windows to accept it.

Follow this page to convert a key and cer pair to pfx: [Create PFX Cert File \(for IIS\)](#)

The conversion is pasted, here, for clarity:

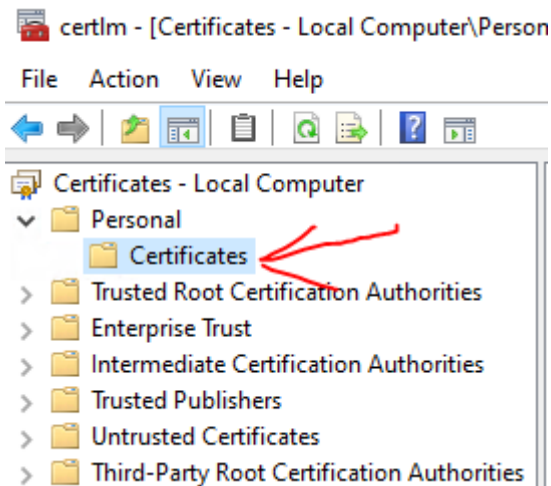
```
"C:\Program Files\Git\usr\bin\openssl.exe"  
pkcs12 -export  
-out myhostname.companyname.com-cert.pfx  
-inkey myhostname.companyname.com.key  
-in myhostname.companyname.com.cer
```

NOTE: The above is displayed on multiple lines for readability.
To execute, it must be all on one line.

Import to Cert Store

Once the PFX is generated, you can import it to the Windows Certificate Store.

For SSRS, import the pfx file to the Local Machine/Personal folder of the certificate store.



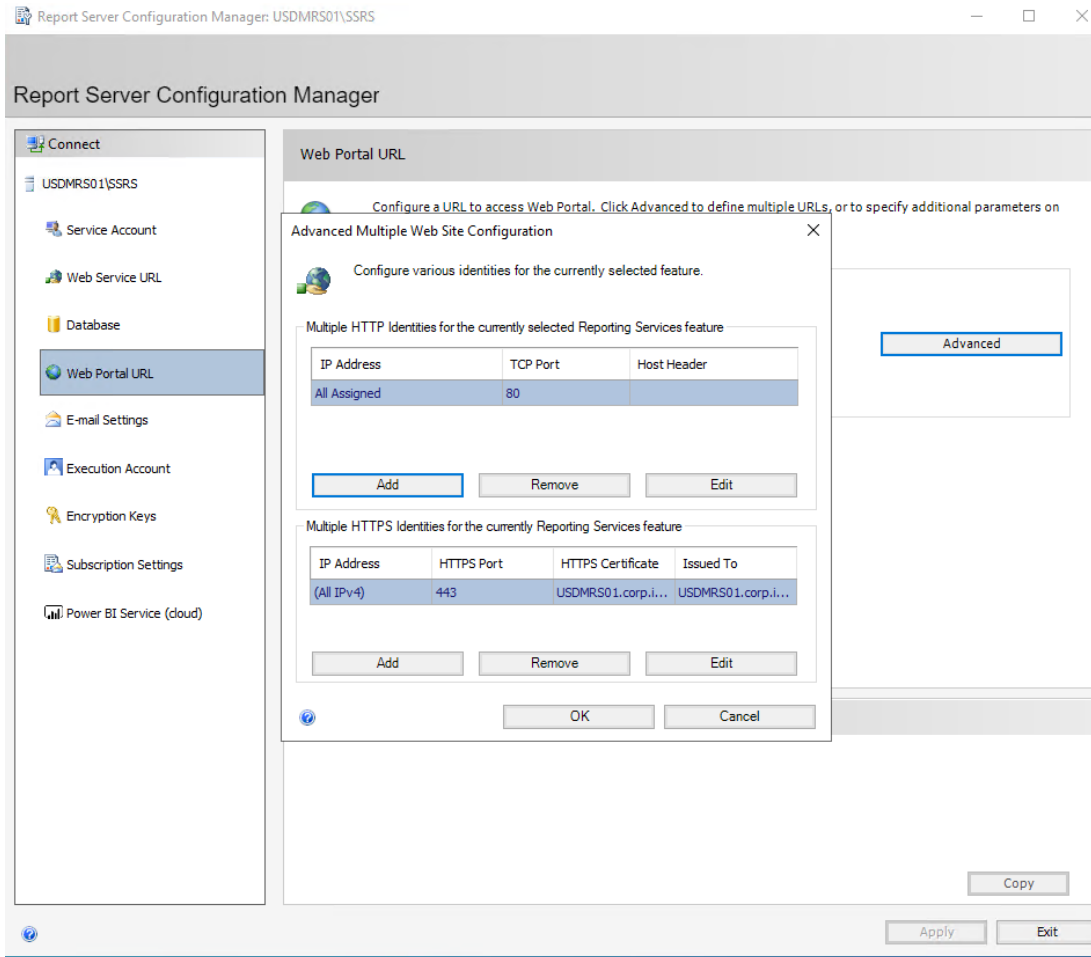
Certificate Binding

Once the certificate is in the store, you can bind it in SSRS.

Open the Report Server Configuration Manager from the Start menu.

From the left tree, click on Web Portal URL.

Click the Advanced button, and locate the lower pane for https bindings.



Remove any existing SSL binding, by selecting it and hitting Remove.

The old binding may not successfully delete.

So you can open an elevated command line, and delete the old cert binding, with this:

```
netsh http delete sslcert ipport=0.0.0.0:443
```

Return to the Report Server Configuration Manager, and add the new cert binding.

Select the imported certificate (from earlier), and bind it to port 443.

NOTE: Both Web Service URL and Web Portal URL must have the same origin.
So, be sure to do the same addition to the Web Service URL tab.

You should be able to access the report server URL.