2018

Backing Up Cornerstone SQL Server Data

Steps for configuring a Cornerstone server to back up configuration data stored on a SQL Server to a local disk.

QuickStart Guide

Backing Up Cornerstone Configuration Data from SQL

Cornerstone creates a database in Microsoft SQL Server to store configuration information for each server and its users and groups. This is ideal for scaled environments where multiple Cornerstone servers on different computers are connected on the same network, as each server can access the same data from a central data repository.

Configure Cornerstone Server

You will need a running Cornerstone server configured to use information stored on an SQL Server database in order to back up SQL data. If you want to configure a backup for a server already storing information in SQL, skip to the next section on Backup Templates.

To set up a server to use SQL, you will need to know the SQL Server instance you wish to store information in.

1. Launch the New Server Wizard and follow the prompts until the wizard asks for database storage information.

2. Enter your SQL Server instance name.

3. Enter a new name for the database that will be created in the SQL Server to hold this Cornerstone server's configuration data.

NOTE: The SQL Server instance name and database name are important variables throughout this process. Take note of them for later reference.

Finish creating your server to your specifications.

Tailor Backup Templates

Two script files are packaged in the Cornerstone installer and can be found in the Templates folder in the Program Files on your C: drive. The first batch file is the command used to launch the backup. The second SQL script tells the SQL Server to place the copied information in the appropriate backup folder.

Create a backup folder on your local disk or UNC with the same name as the database you created for your Cornerstone server (step 3 above), then alter the script templates according to these instructions.

The first file is a .bat batch file with the following code:

```
sqlcmd -S <SQLSERVER_INSTANCE> -d <DATABASE_NAME>-E -i "C:\Program Files\
South River Technologies\Cornerstone MFT\templates\sqlbackup.sql" -o "C:\
backups\<DATABASE_NAME>\sqlbackup.log"
```

 southrivertech.com

Replace the bracketed variables with your SQL Server and the database name selected during server creation. Do the same for the .sql file, which has the following code:

```
BACKUP DATABASE [<DATABASE_NAME>] TO  DISK = N'c:\backups\<DATABASE_NAME>' WITH
NOFORMAT, INIT,  NAME = N'<DATABASE_NAME>', SKIP, NOREWIND, NOUNLOAD,  STATS = 10

GO
```
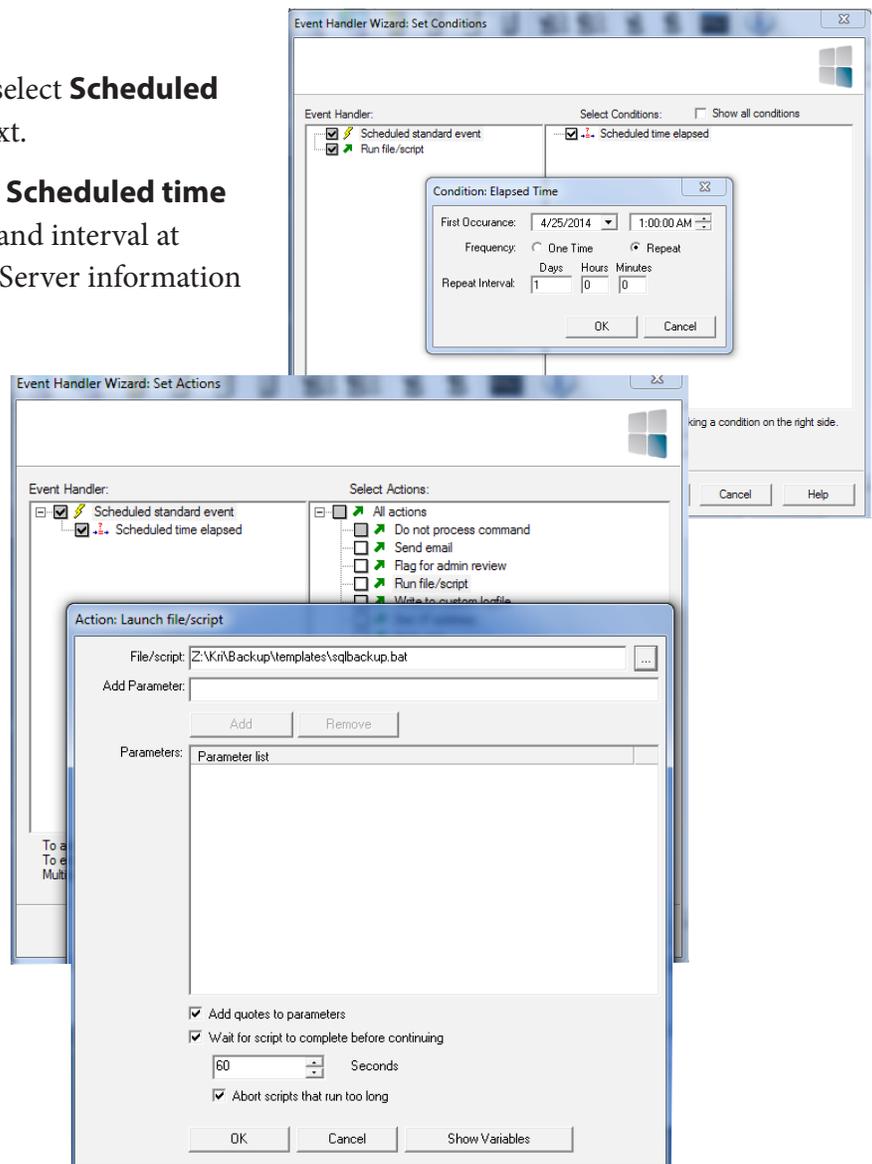
In the script above, replace "c:/backups" with the name of your backup folder.

Create a SQL Backup Event

The backup process is automated through Events Management. By creating one event, you will instruct Cornerstone to copy all SQL files to your designated backup folder at a regular interval of your choosing.

1. Expand the appropriate server, select Events, and click Add.

2. Exapand **Scheduled events**, select **Scheduled standard event**, and click Next.

3. On the Conditions page, select **Scheduled time elapsed**. Here, enter the time and interval at which you would like the SQL Server information to be backed up. Click Next.

4. On the Actions page, select **Run file/script**, which will open a window. In the first box, you can use the "..." button to browse for the .bat file template you altered earlier. Click OK.

Finish the event handler. The new event should appear in the Events list, and your Cornerstone data will back up to the local backup file at the time you scheduled.

© South River Technologies

System Requirements

Supported Operating Systems

- Windows Server 2012-R2 editions, 32-bit and 64-bit

- Windows Server 2012, all editions, 32-bit and 64-bit

- Windows Server 2008-R2, all editions, 32-bit and 64-bit

- Windows Server 2008, all editions, 32-bit and 64-bit

- Windows Server 2016, all editions, 32-bit and 64-bit

Minimum Hardware Requirements

- 2 GHz Pentium® class processor

- 4GB of RAM is required; 8GB of RAM is recommended

- Minimum 100MB of free disk space for the application

- Minimum SVGA (800x600) resolution display is required to run the Administration console program.

Minimum Software Requirements

- Microsoft .NET Framework v4.0 is required

- Microsoft SQL Server 2005 or later is required

- Microsoft SQL Server Management Studio Express (only if you plan to enable Statistics Tracking and reporting)

Limitations

- Titan FTP Server is a multi-threaded, dynamic server solutions for the Microsoft Windows operating system. While designed to handle an unlimited number of user connections and servers, like all software, Titan is limited by the resources of the computer; most notably, those limitations imposed by the Windows Sockets (WINSOCK) Library.

About South River Technologies

South River Technologies (SRT) is an innovator in secure file management software. SRT software allows users to securely access, manage, and collaborate on files over the Internet, streamlining business processes to improve productivity. SRT's products enhance customers' existing applications by instantly enabling secure access and collaboration within those applications. More than 90,000 customers in 140 countries use SRT's software to make remote file access and collaboration more efficient for their customers, partners, and distributed workforce.

For more information, please visit www.southrivertech.com. South River Technologies, Cornerstone MFT, Titan FTP Server, WebDrive, and DMZedge Server are registered trademarks of South River Technologies, Inc. in the U.S. and other countries. Any information in this document about compatible products or services should not be construed in any way to suggest SRT endorsement of that product or service.

Contact Information

South River Technologies, Inc.
1910 Towne Centre Blvd
Suite 250
Annapolis, Maryland 21401
USA

Toll Free: 1-866-861-9483
Main: 443-603-0290
Fax: 410-266-1191
Corporate Web site: www.southrivertech.com
Online Support: www.srthelpdesk.com