Microsoft SQL Server Backup to Windows Azure Tool

Overview:

SQL Server 2012 SP1 CU2 and SQL Server 2014 have built in capability to back up to Windows Azure storage. The SQL Server Backup to Windows Azure tool provides the same functionality for previous versions of SQL Server. It can also be used to provide encryption and compression for your backups.

Using the 3-step wizard, you can specify a rule or set of rules that are applied to any SQL Server backup. One example of a rule could be to redirect all local backups to the specified Windows Azure storage. Another example of a rule would be to use compression or encryption for backups stored in a specific location.

Once you configure the rules, these rules are applied to SQL Server Backup files. If the rule is set to use a Windows Azure storage account, the tool redirects the backups to the specified Windows Azure storage account, but leaves a stub file in the local storage with metadata information to be used during restore.

Benefits:

- Support for backups to Windows Azure Storage for SQL Server versions that do not have the built-in capability. Using Windows Azure storage for your backups has several benefits, such as providing off-site storage for disaster recovery, accessibility regardless of location, etc. For more information, see SQL Server Backup and Restore with Windows Azure.
- Encryption support for SQL Server versions that do not have the built in capability. Currently only SQL Server 2014 has encryption support.
- Compression support for SQL Server versions that do not have the built in capability. Currently, SQL Server 2008 supports compression in Enterprise edition only, but SQL Server 2008 R2 and later, encryption is supported on Enterprise and Standard editions.

SQL Server and Operating Systems Support:

This tool is supported on SQL Server 2005 or later, and Operating System versions: Windows Server 2008 or later for Servers, and Windows 7 or later for Client Operating Systems.

Prerequisites:

- Windows Azure subscription and a Windows Azure Storage Account.
    - You can log in to the Windows Azure Management Portal using your Microsoft account. If you do not have a Microsoft account, visit Windows Azure 3-Month free trial.
    - To create a Windows Azure storage account, see How to Create a Windows Azure Storage Account.

- A Windows Azure Blob Storage Container: SQL Server uses the Windows Azure Blob storage service and stores the backups as blobs. A container is a grouping of blobs an all blobs must live in a container.

Installing the Tool:

The setup is simple and involves the following steps:

1. From the download page, download the MSI (x86/x64) to your local machine that has the SQL Server Instances installed. If your production machines do not have access outside of your organization, download to a local share and use the MSI to install the tool on your production machines.
2. Double click the MSI file to start the installation.
3. Read and accept the terms of the license agreement, and click **Install** to start the installation process.

When the installation completes, a service named Microsoft SQL Server Backup to Windows Azure Tool Service is created on the machine. This service runs the SQLBackup2Azure.exe to apply the configured rules such as backing up to Windows Azure Blob storage, compression, or encryption of the backup files. The installation also requires and attempts to create a low privilege account which is used to run the service. When installed, Microsoft SQL Server Backup to Windows Azure Tool adds two objects to the users and groups on the local machine:

- A user group called “TempGroup”
- A user in the group called “SQLBackup2Azure”

These names are hard coded. The password for the user SQLBackup2Azure account is generated automatically and not accessible by anyone including Microsoft.

When a rule is created, Read/Write permission for the specified folder is automatically granted to the group “TempGroup”. This includes the SQLBackup2Azure account. If the account does not have permission to read or write to the specified folder, it will not function correctly.

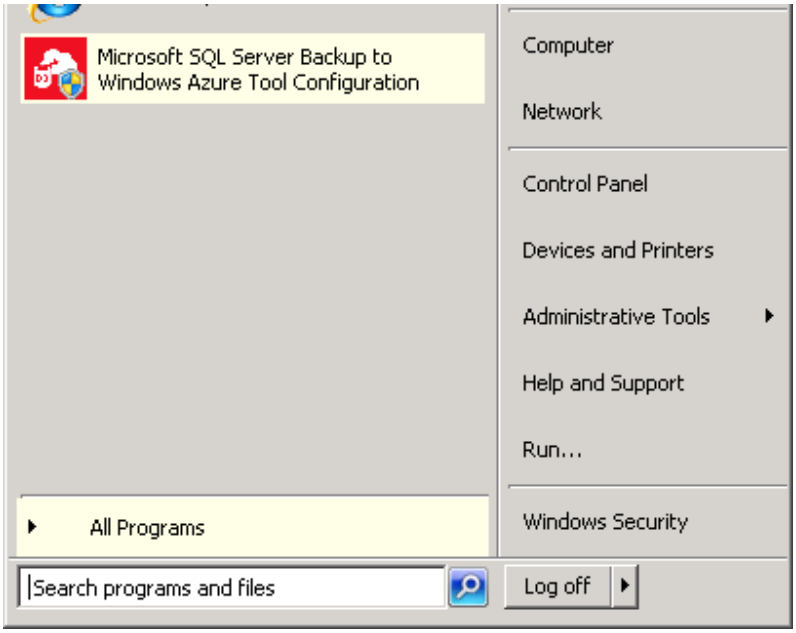
How to Use the Tool to Create Rules:

NOTES:

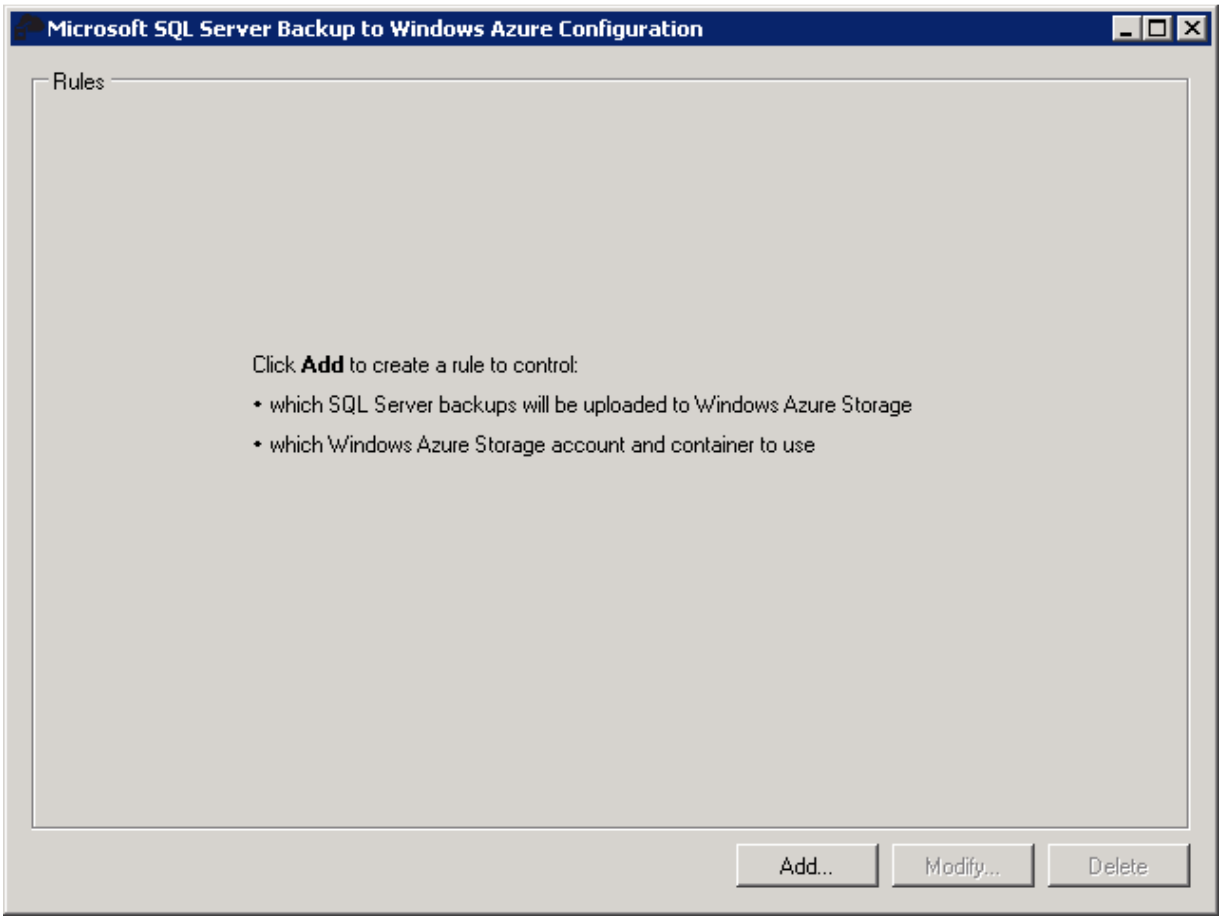
Going through the wizard to setup the rules allows the program to process the backup files that should be encrypted, compressed or uploaded to Azure storage. This tool does not provide for scheduling, error tracking or logging of the backups. All backup job scheduling, maintenance and error tracking should be done by using SQL Server Management Studio or other applications which can provide this functionality.

In addition, turn off SQL Backup compression on the databases that you want to back up with the Tool. SQL Backup Compression does not give additional benefit over the tool’s compression features. If you wish to compress your backup, please use the Tool for compression.

1. Once installed, the tool should be appear on the Programs list.

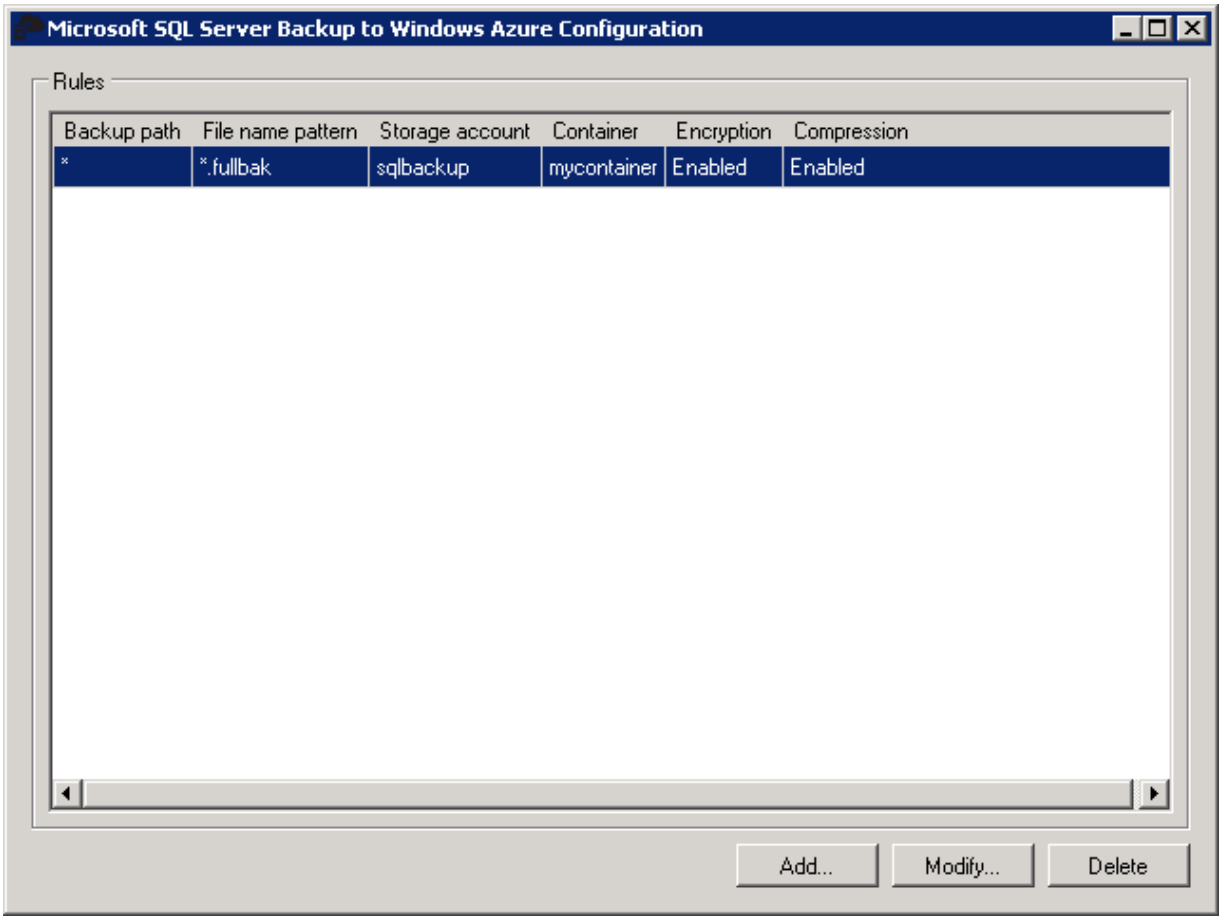


2. Click on the "**Microsoft SQL Server Backup to Windows Azure**…" to start the Wizard.
3. On the **Rules** page, click **Add** to create a new rule. NOTE: Only the **Add** button is enabled if you are using this tool for the first time.



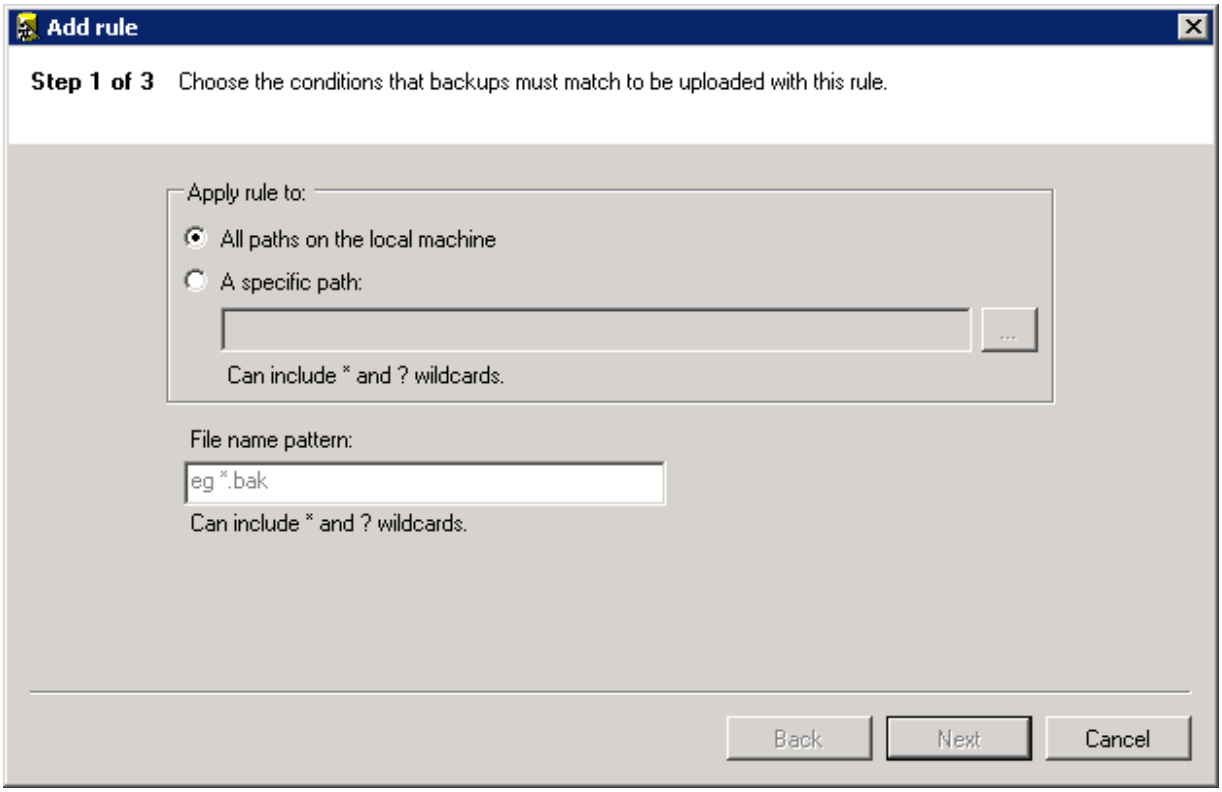
Once you have created one or more rules, you will see the existing rules and the option to

Modify or **Delete** the rule as shown below.



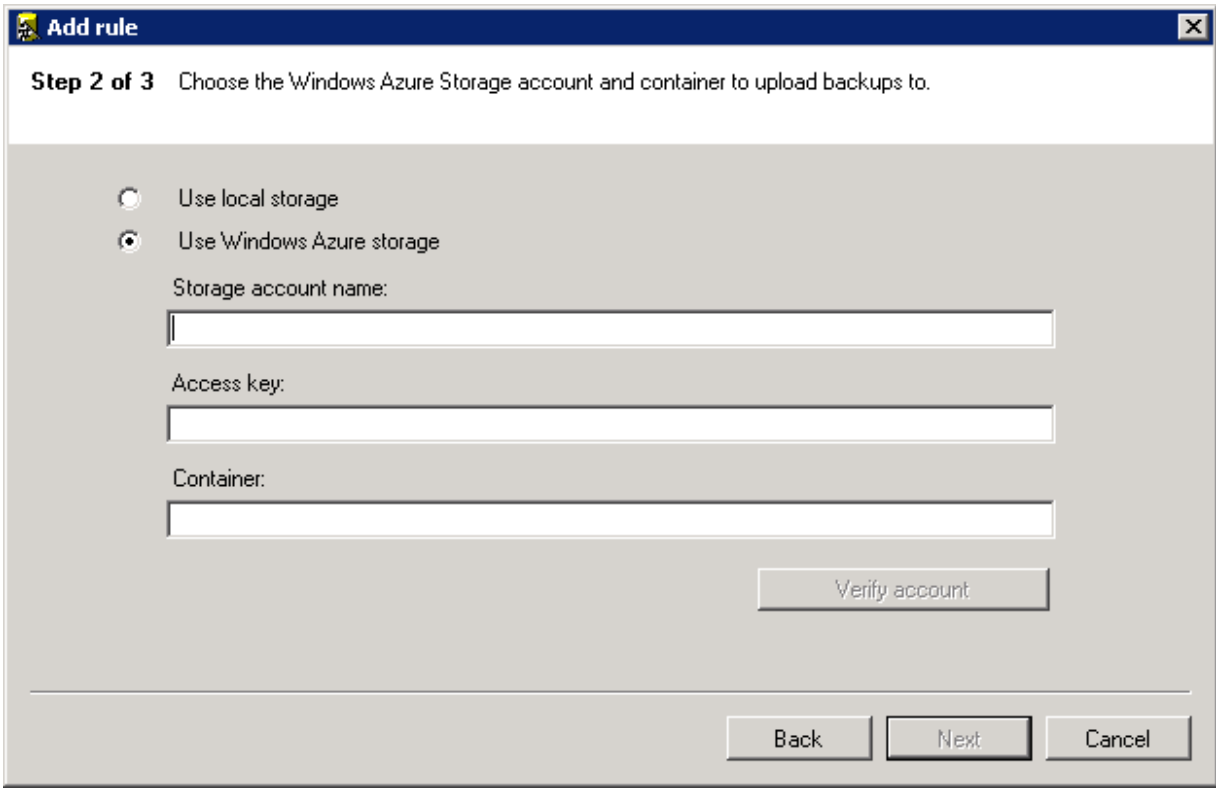
4. On the **Add Rule (Step 1 or 3)** page, you can either choose to apply the rule to all paths to the local machine or to one specific location. You must also specify the file name pattern that this rule should apply to. For example, if you want to apply this rule to all files with the extension

.bak, you would specify *.bak in the File name pattern field.



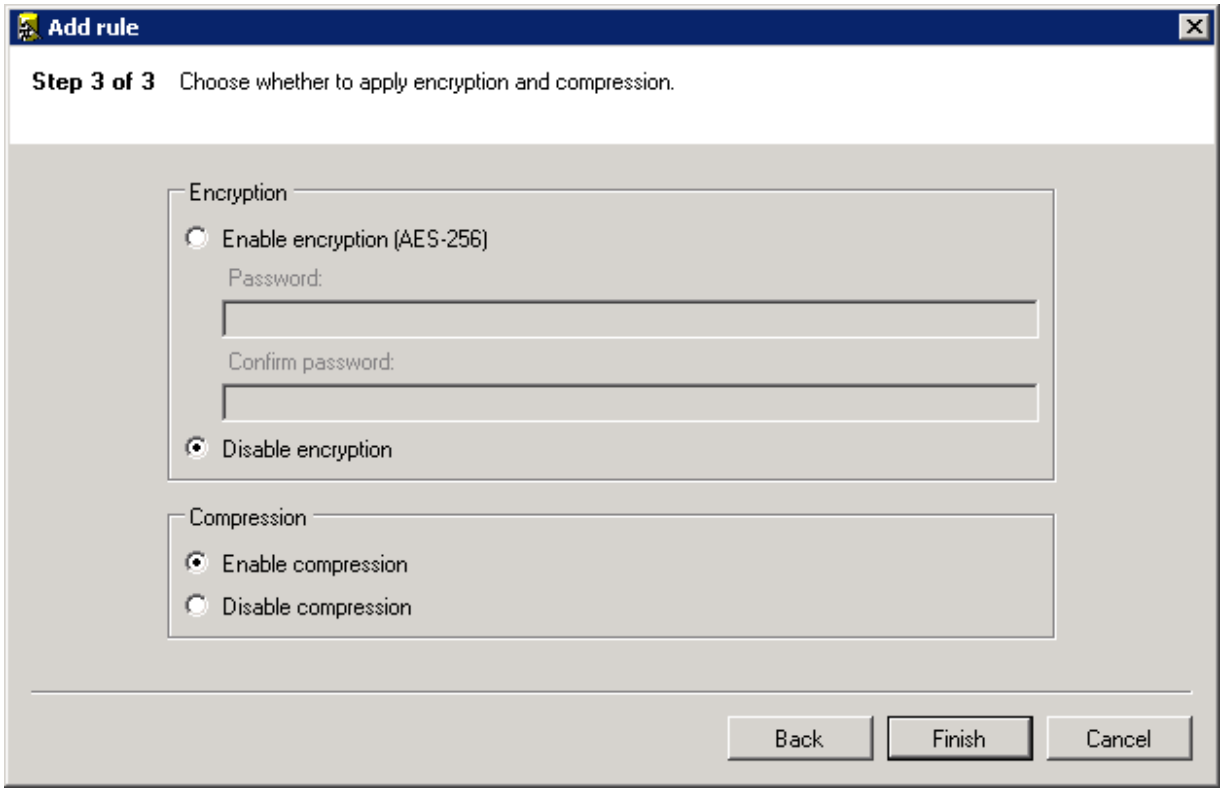
5. **On the Add Rule (Step 2 of 3)** page, you can specify the Windows Azure storage account information, so the backups you specified in Step 1 can be redirected to use the Windows Azure storage as the backup destination. Alternatively, you can choose to keep the local storage as the backup destination.
    a. For Windows Azure Storage, you must specify the name of the account, the storage access key, and the name of the container. You can retrieve the name of the storage account and the access key information by logging into the Windows Azure management portal. For more information on where to find this information, see http://go.microsoft.com/fwlink/?LinkId=392743. The storage name and access key are used to authenticate to the storage account, the container. Click **Verify account** to ensure that the information specified is valid and the tools is able to connect to the

storage account.

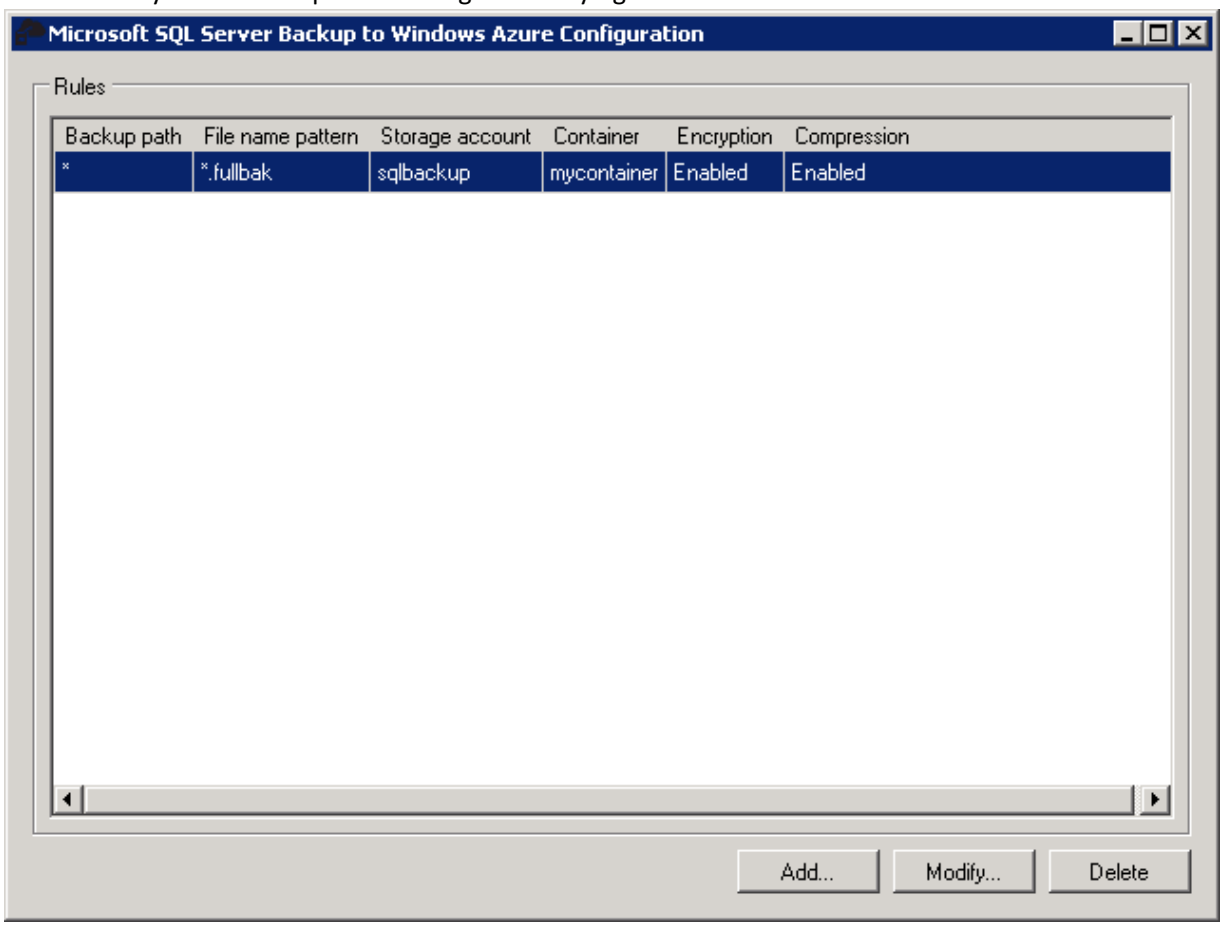


6. **On the Add Rule (Step 3 of 3)** page, you can enable or disable encryption or compression. If you enable encryption, you must specify a password. The password is used for decryption purposes. For more information, see Backup Encryption. Once you specify the options, click **Finish** to

create the rule.



7. On clicking Finish you will see the following page with the rule configuration. You can close the wizard once you have completed adding or modifying the rules.



Restoring a Database from a Backup Taken with SQL Server Backup to Windows Azure Tool in Place:

The SQL Server Backup to Windows Azure Tool creates a 'stub' file with some metadata to use during restore. Use this file like your regular backup file when you wish to restore a database. SQL Server uses the metadata from this file and the backup on Windows Azure storage to complete the restore. For example:

RESTORE DATABASE AdventureWorks2012

FROM DISK = '<metadata file location>'

Restoring a Database from a Backup Taken with SQL Server Backup to Windows Azure Tool in Place If You have Lost the Stub File

If you have lost the stub file (e.g. through loss of the storage media that contained the stub file) and you have chosen the option of backing up to a Windows Azure Storage account, you may recover the stub file through Windows Azure Storage by downloading it from the storage container in which it was placed. You should then place the stub file into a folder on the local machine where the Tool is configured to detect and upload to the same container with the same encryption password if encryption was used with the original rule.

Frequently Asked Questions:

Q: The tool doesn't seem to detect and select my backup files correctly.

A: Try the following:

1. Confirm the "Microsoft SQL Server Backup to Windows Azure" service is started
2. Check if your rule matches the file name you try to back up. A rule will look for files named ".bak", for example, if you placed ".bak" in the "File Name Pattern" field. To properly match files, you may need to use wild cards.

Q: I can't start the "Microsoft SQL Server Backup to Windows Azure" service.

A: Check Windows System Event logs for errors when the service fails to start.

Q: Is there any retry logic built into the tool?

A: No. In the event of a loss of network connectivity, SQL Server will surface the error as not being able to write to the device. You will need to clean up the files relating to this backup operation (if any) and retry the backup.

Q: Can I back up to an existing backup file on disk?

A: No. This is a known limitation of the Tool. If you want to create a backup using the Tool, you must create a new backup file instead of overwriting/appending to an existing backup file.

Q: Should I use SQL Backup Compression with the Tool?

A: No. SQL Backup Compression provides no additional benefit over the Tool's compression algorithms. We strongly suggest you turn off SQL Backup Compression for any backups taken to a folder that the tool is monitoring. If you wish to compress these backups, please turn on the Compression option in the Tool instead of using SQL Backup Compression.

Additional Resources:

SQL Server Backup and Restore with Windows Azure.