

PS SIMPLE BACKUPER TOOL -MANUAL

WHAT IS IT?

This tool will allow you to make a copy of the local folders or network shares available to the public (along the UNC path). Works recursively, but allows you to define folders to be excluded.

The tool is also able to back up MSSQL databases. In principle, it copies all databases on the server (except tempdb), but it also allows you to define your own exclusions. To use this feature, the script must run on the same machine as the MSSQL server installed.

This tool is configured using a simple xml configuration file.

REQUIREMENTS

- Powershell 4.0 or later.
- SQLPS Powershell Module
- 7-zip archive tool.

DEPLOYMENT

INSTALATION ON THE SERVER

Simply copy the downloaded package to the server and extract it to any folder on your hard drive. Remember that logs are stored in this folder (in the .\logs subfolder), so you will need from several dozen, up to several hundred megabytes per year for log files, depending on the frequency of copying and the number of copied resources.

For example:

Create a folder c:\PSSimpleBackupTool\ and extract the files there.

CONFIGURATION OF TASKS - WORKING WITH XML CONFIGURATION FILE

All task parameters can be found in the .\config file. Open the file using a good editor (e.g. Notepad++).

Example configuration:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <backups>
3   <!--File backup Job parameters - duplicate this node if you want to more than one file type job-->
4   <backup name="NetStorage" type="file">
5     <source path="\\networkstorage.domain.local\userdata\"/>
6     <target path="C:\BACKUPS\Output" remove="true"/>
7     <exclude items="Folder1,Folder2,Folder3"/> #Comma separated names of folders(!) to exclude
8     <smb sendbackup="true" credentialVaultName="backup" unc="//backupstorage.domain.local/user_backups\" />
9   </backup>
10  <backup name="LocalData" type="file">
11    <source path="c:\users\bob\documents\"/>
12    <target path="C:\BACKUPS\Output" remove="false"/>
13    <exclude items="" /> #Comma separated names of folders(!) to exclude
14    <smb sendbackup="false" user="" pass="" unc="" />
15  </backup>
16  <!--MSSQL backup Job parameters -->
17  <backup name="database" type="mssql">
18    <source path="SRV01\OPTIMA\"/>
19    <target path="C:\BACKUPS\Output" remove="true"/>
20    <exclude items="master"/> # Comma separated names of databases
21    <smb sendbackup="true" credentialVaultName="backup" unc="//backupstorage.domain.local/db_backups\" />
22  </backup>
23  <!-- Common parameters -->
24
25  <sevenzip path="c:\program files\7-zip\7z.exe" />
26  <temp path="C:\BACKUPS\Temp" />
27  <archive compressionlevel="5" />
28
29 </backups>
```

The configuration can be divided into two parts:

- 1) Task configurations - node backup
- 2) Common parameters:
 - a. sevenzip path
 - b. temp path
 - c. archive compression level

TASK CONFIGURATION

The tool will perform as many backup tasks as you place <backup> nodes in the configuration file during operation.

A backup node has several following attributes:

Attribute name	[Data type: possible values]	Description
name	[string]	label, task name (used in the backup file name)
type	[switch: file,mssql]	specifies the type of task. The [file] value for file and folder backup tasks. The [mssql] value for the database backup task

Subnode	Attribute	[Data type: possible values]	file type task	mssql type task
source	path	[string]	Absolute file path or unc patch to source folder (which will be backed up). e.g. c:\user\bob\documents \\share\users\bob	Fully qualified sql instance name, e.g. DatabaseSrv\mssqlexpress or Databaseserver
target	path	[string]	Absolute path to the folder where the 7z file, containing the backup, will be stored.	Absolute path to the folder where the 7z file, containing the backup, will be stored.
	remove	[switch: true, false]	Determines whether the copy will be deleted from the local folder after it has been uploaded to the server (see below).	Determines whether the copy will be deleted from the local folder after it has been uploaded to the server (see below).
Exclude	items	[string: comma separated list of values]	Defines the names of the folders to be excluded from the backup.	Defines the names of the databases to be excluded from backup.
smb	sendbackup	[switch: true, false]	Specify whether the copy from the local folder should be copied to the remote server (smb).	Specify whether the copy from the local folder should be copied to the remote server (smb).
	credentialVaultName	[string]	Name of vault name in Windows Credential Manager where you stored credentials for smb session of this job.	Name of vault name in Windows Credential Manager where you stored credentials for smb session of this job.
	unc	[string]	Defines the unc path where the backup file will be saved.	Defines the unc path where the backup file will be saved.

COMMON PARAMETERS

It is also necessary to define three parameters common to the whole process.

Subnode	Attribute	[Data type: possible values]	Description
sevenzip	path	[string]	Path to the 7z.exe executable file. Will be used to compress backup files.
temp	path	[string]	Specifies a temporary folder where the tool will save files during backup.
archive	compressionlevel	[numbers: 0 1 3 5 7 9]	Sets level of backup compression. x=0 means copy mode (no compression archive).

CONFIGURATION OF CREDENTIALS MANAGER

"The "PS Simple Backuper" is able to send the copies to another smb location. Usually, backups are stored in a location that requires authentication. If you are using a public location (without logging in) leave credentialsVaultName attribute empty. If you want to log in to a specific user, please refer to the following description.

As you can see, in the configuration file, it does not store the login and password needed to authenticate the smb session. I consider it a bad practice to keep such data in a plain text. Fortunately, in Windows, we have a special place where we can safely store credentials in a classified form. This is what Credential Manager is all about. To use it, we need to include appropriate credentials in it.

Note the smb node and credentialVaultName attribute in the configuration file. There we will place the name of the container in which we will store credentials in Windows Credential Manager.

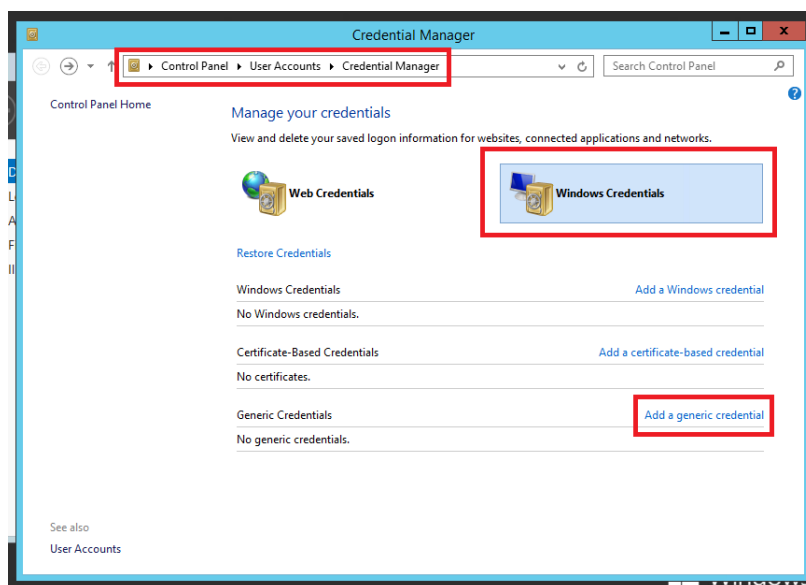
You can use separate credentials for each <backup> node, so you can send each backup to a different location using different credentials.

Steps:

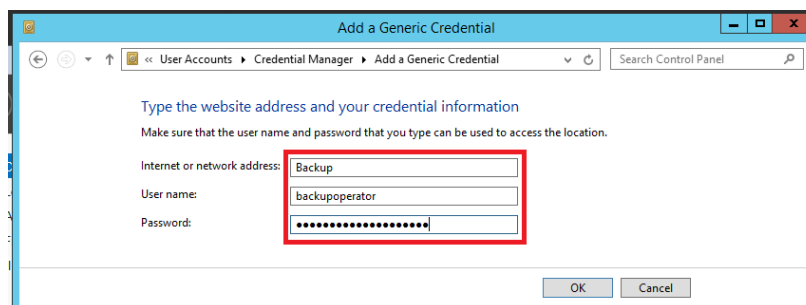
1. Open config.xml insert the name of the container with credentials in the credentialsVaultName attribute – e.g. "backup.

```
<backup name="NetStorage" type="file">
  <source path="\\networkstorage.domain.local\userdata\"/>
  <target path="C:\BACKUPS\Output" remove="true"/>
  <exclude items="Folder1,Folder2,Folder3"/> #Comma separated names of folders(!) to exclude
  <smb sendbackup="true" credentialsVaultName="backup" unc="//backupstorage.domain.local/user_backups\" />
</backup>
```

2. Open Credential Manager. Open Windows Credentials and in select New generic credentials



3. In the **Add a Generic Credentials** fill the all fields. Field Internet or network address should contain your credentialsVaultName value (e.g. backup).



If you want to use more than one set of credentials, repeat the configuration for each of them.

FIRST TEST

Before you will configure the scheduled task you need to check if you configured tool correctly or not. To do this, just run Powershell (as a different user – use the credentials of this user, who have privileges to perform the backup of database and is able to connect to source of network share which you want to backup). In the powershell console, navigate to the home folder of tool (there, where you extracted downloaded package) and run the tool using command:

```
.\RunBackup.ps1
```

and observe if everything is going smoothly. Check if the log file was created correctly. If something is configured wrong, you will observe errors in the script output (in Powershell console).

If everything is ok, go to the next part - Scheduled task configuration.

SCHEDULED TASK CONFIGURATION

Of course, the backup should be done automatically at a convenient moment - e.g. at night, when nobody uses the data that we protect. That is why you need to configure the scheduled task in the environment. You can use CobianBackup or other tools for this purpose. I prefer the mechanisms of the operating system. So..

Steps:

1. Open the Task Scheduler,
2. On the **left panel**, create the Backup folder and select it.

3. On the right panel, select **Create Task...**
4. In the Create Task window, type the name of the task (any - e.g. "data backup") if you want to, fill in the description.
5. Select **Run whether user is logged or not** and **Run with highest privileges**. Go to the **Triggers** tab,
6. On the Triggers tab, click **New...** and select a startup scheme that is convenient for you,
7. go to the Actions tab and click **New...**
8. type in the Program/script
field: **%SystemRoot%\system32\WindowsPowerShell\v1.0\powershell.exe**
9. Enter in the box **Add Arguments: -ExecutionPolicy Bypass -File .\RunBackup.ps1**
10. In the Start in field, type the path where the RunBackup.ps1 script is located.
11. Confirm all Task window.

When you confirm the task, system ask you for credentials. **This is important** - remember that you need to run the task in the context of a user who has the right to login to the database (if you configured mssql job type) and back up the database. Also, this user need to have READ privileges in the source folder which you want to back up. Enter these credentials.

Now, you are ready to the test. On the main window of Task scheduler, please right click on the task name and select RUN. The status of job will should change to "running". In the folder where you extracted the tool, should be freshly created logs folder. There you should find log of current job. If you find it, everything should be ok. Just check if the tool could connect to the network share and database.

TROUBLESHOOTING

1. Cannot connect to the database:
 - a) Check if the Instance name is correct
 - b) Check if your user has privileges to connect to the database and backup database

Tip:

The easiest way to specify the name of the instance is to run **Sql Server Menagement Studio** and copy the name of the server from the **Connect to Server** window.

Try logging in to the user you are using to create a copy of the database. Check if you can make a copy from the SSMS.

