



Technical Report

SnapCreator 3.2 Installation and Administration Guide

Keith Tenzer, Alfredo Quiroga, Dietmar Ebert,
Antonio Jose Rodrigues Neto, NetApp

May 2010 | TR-3841

TABLE OF CONTENTS

1	INTRODUCTION	4
2	SNAPCREATOR OVERVIEW	4
2.1	SNAPCREATOR ARCHITECTURE OVERVIEW	4
2.2	SNAPCREATOR FEATURES	5
3	SNAPCREATOR AGENT	6
4	BACKUP PROCESS	6
4.1	APPLICATION MODULES	7
4.2	APPLICATION QUIESCE/UNQUIESCE COMMANDS	7
4.3	ARCHIVE COMMANDS	7
4.4	PRE COMMANDS	7
4.5	POST COMMANDS	7
4.6	CONNECT TO NETAPP STORAGE	8
4.7	NETAPP STORAGE ZAPI CALLS	8
5	SECURITY FEATURES	8
5.1	STORAGE CONTROLLER SECURITY	8
5.2	SNAPCREATOR AGENT SECURITY	8
6	SNAPCREATOR POLICIES	9
7	SNAPCREATOR SERVER INSTALLATION	9
7.1	EXTRACTING THE ZIP FILE	9
7.2	COPYING THE BINARY TO THE SNAPCREATOR ROOT DIRECTORY	9
7.3	RUNNING SNAPCREATOR SETUP	9
7.4	CREATING A ZAPI USER	10
7.5	INSTALLING SNAPCREATOR GUI (NOT REQUIRED)	10
8	SNAPCREATOR AGENT INSTALLATION	11
8.1	EXTRACTING THE ZIP FILE	11
8.2	COPYING THE BINARY TO SNAPCREATOR ROOT DIRECTORY	11
8.3	INSTALLING AGENT AS A SERVICE (WINDOWS)	11
8.4	INSTALLING THE AGENT AS A DAEMON (UNIX)	11
9	SNAPCREATOR SERVER CONFIGURATION	12
9.1	CREATING A CONFIGURATION FILE (CLI)	12
9.2	CREATING A CONFIGURATION FILE (GUI)	12
9.3	REFERENCING	13
9.4	BASIC CONFIGURATION	13
9.5	NETAPP OPTIONS	15
9.6	OPERATIONS MANAGER SETTINGS	17

9.7	OTHER OPTIONS.....	18
9.8	CLIENT/SERVER CONFIGURATION.....	20
9.9	ADDITIONAL MODULES.....	20
9.10	POSTGRESQL MODULE.....	23
10	SNAPCREATOR AGENT CONFIGURATION.....	26
11	RUNNING SNAPCREATOR SERVER.....	27
12	RUNNING SNAPCREATOR AGENT.....	30
13	SNAPCREATOR EXAMPLES.....	31
13.1	NETAPP SNAPSHOT (ALWAYS REQUIRED).....	31
13.2	NETAPP APPLICATION-CONSISTENT BACKUP.....	31
13.3	NETAPP SNAPSHOT AND BACKUP TO TAPE.....	32
13.4	NETAPP BACKUP TO DISK (OSSV).....	32
13.5	NETAPP SNAPSHOT AND BACKUP TO DISK (SNAPVAULT).....	32
13.6	NETAPP SNAPSHOT AND BACKUP TO DISK (SNAPMIRROR).....	33
13.7	NETAPP SNAPSHOT AND DATABASE REFRESH (PRIMARY).....	33
13.8	NETAPP SNAPSHOT AND DATABASE REFRESH (SECONDARY).....	34
13.9	NETAPP SNAPSHOT AND PROTECTION MANAGER UPDATE.....	34
14	TROUBLESHOOTING.....	35
14.1	NETWORKING.....	35
14.2	UPGRADING FROM SNAPCREATOR V2.6.....	36
14.3	UPGRADING FROM SNAPCREATOR V3.0.....	36
14.4	UPGRADING FROM SNAPCREATOR V3.1.....	37
15	FURTHER QUESTIONS/HELP.....	38

1 INTRODUCTION

This technical report provides an overview of NetApp® SnapCreator and describes the procedures to install, configure, and run SnapCreator on NetApp storage systems.

2 SNAPCREATOR OVERVIEW

SnapCreator provides a central framework that integrates NetApp Snapshot™ technology with applications that do not have SnapManager products in a way that is seamless to our customers. Normally, this requires a customized script that would then interface with the application and the NetApp storage system. These customized scripts are written over and over every day. SnapCreator saves time and provides our customers with the best most reliable solution possible.

As integration for most applications is unique and challenging, most products support only a few select applications. In contrast, SnapCreator provides application integration through modules or plug-ins that enables it to support any application anywhere. SnapCreator offers a framework in which you can integrate application consistency (backup) scripts or use the built-in SnapCreator modules.

Currently there are seven application modules available for SnapCreator, with more on the way. The seven modules are Oracle®, MaxDB®, MySQL®, Lotus Domino, DB2, PostgreSQL, and Sybase®. SnapCreator handles communicating with NetApp storage and performs various tasks which includes policy-based Snapshot management (via API or SnapDrive®), an optional LUN or volume clone, seamless integration with SnapMirror® or SnapVault®, and integration with Operations Manager or Protection Manager. SnapCreator is by no means a replacement for our SnapManager® and SnapDrive Products; in fact, it integrates with both products.

2.1 SNAPCREATOR ARCHITECTURE OVERVIEW

Figure 1 depicts the SnapCreator architecture.

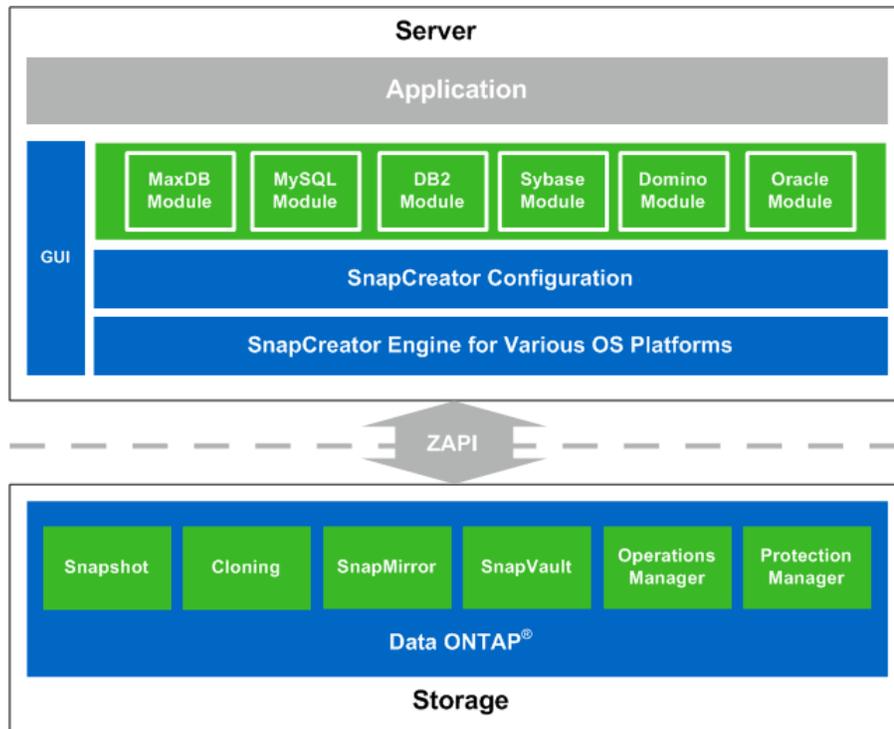


Figure 1) SnapCreator architecture.

2.2 SNAPCREATOR FEATURES

SnapCreator contains the following features:

- Browser-based GUI on NetApp Web Framework (NWF)
- Integration with SnapDrive
- Integration with the following NetApp technologies through ZAPI: Snapshot, SnapVault, SnapMirror, LUN cloning, volume cloning, and IGROUP mapping
- Integration modules for Oracle, MaxDB, MySQL, Lotus Notes, DB2, PostgreSQL and Sybase
- Integration with any application or database that runs in an open systems environment (you can write the application backup script or module if one does not exist)
- Integration with NetBackup™, CommVault, or any backup software with CLI commands
- Ability to configure multiple Snapshot or SnapVault policies with different retentions
- Both Snapshot and SnapVault policies are managed from SnapCreator
- Ability to create and manage Snapshot copies and SnapVault retentions across multiple volumes and storage controllers
- Support for volume and qtree SnapMirror
- Support for FAS and vFiler™ units
- Support for consistency groups (consistent Snapshot copies across multiple volumes or appliances)
- Support for 1-to-1, fan-in, and fan-out SnapMirror or SnapVault relationships
- Support for cascading SnapMirror Snapshot->SnapVault->SnapMirror
- A dynamic configuration file in which users can pass their own user-defined values
- Granular error logging and the ability to send error messages via e-mail or other tools
- Integration with Operations Manager (the ability to create events in Operations Manager)
- Integration with Protection Manager (SnapCreator backups can be registered in Protection Manager)
- Integration with Open Systems SnapVault (OSSV)
- Support for password encryption so clear text passwords are not saved in the configuration file
- Support for both volume, file, and SnapVault restore but without direct application integration
- Agent for central backup management
- Global configuration files

SnapCreator does not do the following:

- SnapCreator is not a replacement for SnapManager products.
- SnapCreator does not directly handle the mounting of cloned LUNs; mount commands or SnapDrive is required for that.
- SnapCreator does not create or manage SnapVault and SnapMirror relationships.

3 SNAPCREATOR AGENT

The SnapCreator agent provides the ability to run backups centrally, meaning all SnapCreator configurations can be stored on the same system and all backup jobs can be scheduled from the same host.

Figure 2 depicts the SnapCreator architecture.

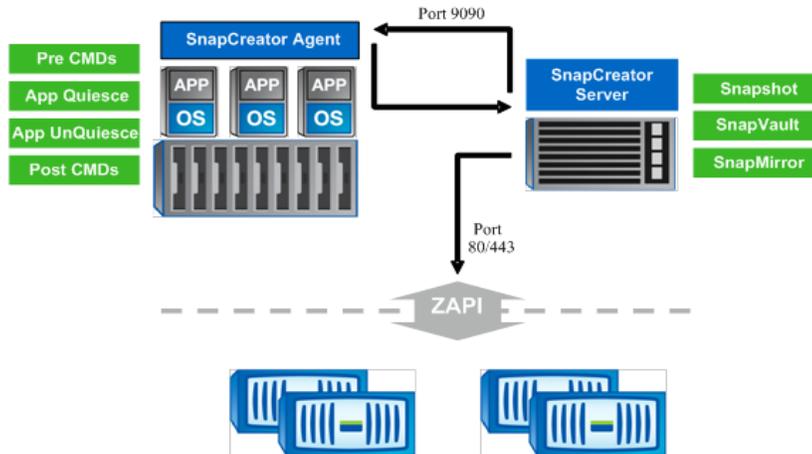


Figure 2) SnapCreator agent.

To quiesce the application, SnapCreator uses its agent, which runs as a daemon. The default port is 9090, but any port can be chosen. SOAP over the HTTP protocol is used for communication. All application modules are already built into the agent, so all applications supported by SnapCreator are also supported by its agent. In addition to the application modules, all PRE commands, POST commands, and APP commands can be executed remotely through the agent. This provides the ability to mount file systems or do extra application processing. The agent has an access file under `/path/so/scAgent_v<#>/config/agent.conf` where certain commands can be allowed. By default, everything is denied, meaning only the built-in modules can execute commands through the agent. PRE or POST scripting commands or scripts need to be allowed in the `agent.conf`.

4 BACKUP PROCESS

The backup process is same regardless of the application and consists of the following steps:

1. Achieve application consistency.
2. Achieve file system consistency (optional).
3. Create a NetApp Snapshot copy.
4. Return the application to normal mode.
5. Clone the LUN or volume (optional).
6. Perform various SnapMirror or SnapVault functions (optional).
7. Delete old NetApp Snapshot copies, thus managing retention policies on both primary and secondary storage devices in the case of SnapVault or SnapMirror.

In SnapCreator, each of the above steps presents the opportunity for running optional scripts or commands. These scripts or commands are triggered using PRE and POST command parameters in the config file and are the heart of the integration framework. All of the PRE and POST commands can be run remotely using the SnapCreator agent. For the application quiesce/unquiesce, you can either use the SnapCreator application modules or issue commands and scripts.

4.1 APPLICATION MODULES

SnapCreator offers built-in application support for Oracle, DB2, Sybase, MySQL, MaxDB, PostgreSQL, and Lotus Domino. All of these modules provide application quiesce and unquiesce functionality. Some modules provide additional functionality for cloning and backup validation.

4.2 APPLICATION QUIESCE/UNQUIESCE COMMANDS

The quiesce and unquiesce commands are used to achieve application consistency when not using SnapCreator application modules. NetApp Snapshot copies of volumes without application consistency are crash consistent only.

4.3 ARCHIVE COMMANDS

Archive commands are user-defined commands or scripts that run before the NetApp POST commands (last command that runs). The purpose of the archive commands is to provide the opportunity to interface with archiving tools after a backup occurs.

4.4 PRE COMMANDS

PRE commands are user-defined commands or scripts that run before the application quiesce commands or application integration modules, the NetApp ZAPI or SnapDrive calls, the application unquiesce command or application integration modules, the volume or LUN clone delete commands, and the script exits in the case of a failure. The types of PRE commands are as follows:

- PRE app quiesce
- PRE NetApp
- PRE app unquiesce
- PRE clone create
- PRE app clone create
- PRE NetApp clone delete
- PRE exit
- PRE restore (in the case of interactive restore)

A PRE command is used for a file system sync or consistency check before NetApp calls using ZAPI or SnapDrive are made. In this example, run the file system command or script as a PRE NetApp command.

4.5 POST COMMANDS

POST commands are user-defined commands or scripts that run after the application quiesce command, the NetApp ZAPI calls, and the application unquiesce command. The types of POST commands are as follows:

- POST application quiesce
- POST NetApp
- POST application unquiesce
- POST data transfer
- POST clone create
- POST app clone create
- POST restore (in the case of interactive restore)

A POST command is used for a database check after the application unquiesce command has run in order to make sure that the database is running in normal mode. In this example, you should run the database check command or script as a POST application unquiesce command.

4.6 CONNECT TO NETAPP STORAGE

For every primary or secondary NetApp storage controller specified, a connection is made using ZAPI over HTTP or HTTPS. This connection is made with a user account that exists on the NetApp storage controller, and this user is only granted HTTP login and API roles, meaning that the user has no permission to do anything outside of ZAPI. The root user account also works but NetApp does not recommend using it.

4.7 NETAPP STORAGE ZAPI CALLS

SnapCreator has the ability to create a NetApp Snapshot copy of the following:

- One or more volumes (primary)
- LUN or volume clone of a Snapshot copy
- Map IGROUPS to cloned LUNs
- SnapMirror update of one or more volumes (secondary)
- SnapVault update of one or more volumes (secondary)
- Snapshot rename of the recent Snapshot copy (SnapVault secondary only)
- Snapshot delete of one or more Snapshot copies based on retention policy (SnapVault secondary only)
- Snapshot rename of the recent Snapshot copy (primary)
- Snapshot delete of one or more Snapshot copies based on retention policy (primary)
- Deleting volume or LUN clones

For example, you could have three volumes: vol1, vol2, and vol3. SnapCreator can handle creating regular Snapshot copies of vol1, vol2, and vol3 (primary) and do a SnapVault update on vol2 as well as a SnapMirror update on vol3. Since SnapVault retentions (secondary) are different than normal Snapshot retentions (primary), SnapCreator can handle a different retention policy for primary and secondary storage as well as various policies that will be covered later. This is a huge advantage because it prevents backup scripts from being out of sync with SnapVault schedules on the NetApp storage controller. It is not necessary to configure SnapVault schedules with SnapCreator. This provides an end-to-end process dependent on a single schedule for any application.

5 SECURITY FEATURES

5.1 STORAGE CONTROLLER SECURITY

A user name and password are needed to communicate with NetApp storage controllers. Passwords can be protected so they are not saved in clear text. SnapCreator does use and support NetApp RBAC. NetApp recommends creating a backup user that only has the necessary API permissions and does not recommend using the root user. Network communications are done through HTTP (80) or HTTPS (443), so you need to have one or both of those ports open from the host where SnapCreator runs to the NetApp storage controllers. The user that was created for authentication purposes also needs to be created on the NetApp storage controllers. In the case of HTTPS, make sure it is enabled and configured on the NetApp storage controllers.

5.2 SNAPCREATOR AGENT SECURITY

SnapCreator can also run centrally and use an agent to communicate with database servers. The agent is a daemon that will run by default on port 9090, but it is possible to choose any port. This port must be open from the SnapCreator server to the server running the agent. The communication between the agent and the SnapCreator server is done through SOAP over the HTTP protocol.

6 SNAPCREATOR POLICIES

Policies are user-defined Snapshot retentions that can apply to Snapshot copies on primary storage and also to SnapVault Snapshot copies on secondary storage. You must define at least one policy in the `NTAP_SNAPSHOT_RETENTIONS` configuration parameter. A policy is the number of Snapshot copies that you want to keep. In the case of SnapVault, you have the ability to associate the same policy with different SnapVault retentions. For example, if you want to create daily Snapshot copies and retain them for seven days on primary storage and one month on secondary storage, do the following:

```
NTAP_SNAPSHOT_RETENTIONS=daily:7
```

```
NTAP_SNAPVAULT_RETENTIONS=daily:28
```

You can also specify the minimum age of a Snapshot copy (in days) to make sure that Snapshot copies will only be deleted if they are older than the minimum age, regardless of how many Snapshot copies exist in a given volume. Using the above example of 7 days for Snapshot copies and 30 days for SnapVault Snapshot copies, do the following:

```
NTAP_SNAPSHOT_RETENTION_AGE=7
```

```
NTAP_SNAPVAULT_RETENTION_AGE=28
```

You can define as many policies as desired, but each time SnapCreator is run, it can only use one policy. The maximum age value is global, so it applies to all policies. If you want an additional weekly policy, configure it and then call SnapCreator once a week via cron or task manager with the Snap Type set to "Weekly".

7 SNAPCREATOR SERVER INSTALLATION

The SnapCreator server (scServer) is designed to run on any open systems platform.

7.1 EXTRACTING THE ZIP FILE

Extract the ZIP file to `/usr/local` for UNIX[®] or `C:\` for Windows[®].

Note: On Windows, do not use spaces in the install path (i.e., "Program Files").

7.2 COPYING THE BINARY TO THE SNAPCREATOR ROOT DIRECTORY

The SnapCreator server distribution contains binaries for Windows, Solaris[™], Linux[®], and AIX. The binaries are located in the `/path/to/scServer_v<#>/bin/<os>` directory. Copy the correct binary from `/path/to/scServer_v<#>/bin/<os>` to `/path/to/scServer_v<#>`.

Note: If the OS is not available with the distribution, request a compiled version by either contacting a NetApp sales representative or posting a message to the SnapCreator community at:

http://communities.netapp.com/community/products_and_solutions/databases_and_enterprise_apps/snapcreator

7.3 RUNNING SNAPCREATOR SETUP

SnapCreator setup will configure an admin user for the GUI and set up the GUI parameters. SnapCreator setup is only required for the server, not the agent. It is also only required for the GUI.

```
./snapcreator --profile setup
```

7.4 CREATING A ZAPI USER

Since SnapCreator uses ZAPI, you must create a user on the NetApp storage controller that has the necessary access. It is important that this user only have the access that it requires. NetApp recommends that you create a new role, group, and user so you can control access and lock the SnapCreator user down. A user needs to be created on all NetApp storage systems involved (primary and secondary, in the case of SnapMirror or SnapVault), but depending on the storage controller's function, not all roles are needed. Below are the possible roles that SnapCreator may require:

- 1 login-* (for all login access) or login-http-admin (to only allow API login)
- 2 api-snapshot-* (always required)
- 3 api-system-* (always required)
- 4 api-ems-* (always required)
- 5 api-snapvault-* (required only for SnapVault management)
- 6 api-snapmirror-* (required only for SnapMirror management)
- 7 api-volume-* (required for volume clones)
- 8 api-lun-* (required for volume and LUN clones)
- 9 api-cg-* (required for Consistency Group Snapshot copies)

7.5 INSTALLING SNAPCREATOR GUI (NOT REQUIRED)

The SnapCreator GUI is based on Java™ and uses the NetApp Web Framework (NWF). NetApp recommends using IE or Mozilla browsers. The GUI uses Jetty, which is an imbedded Java application server. The GUI is not required but rather a complementary tool for helping build, manage, and run SnapCreator configuration profiles.

GUI REQUIREMENTS

Java Runtime Environment (JRE) 1.6 or higher must be installed on the system running SnapCreator.

NetApp recommends downloading the Java JRE from <http://www.java.com/en/download/manual.jsp>

On UNIX, the bash shell needs to be installed under `/bin/bash`. If it is not, then make a soft symlink pointing `/bin/bash` to wherever bash is installed.

GUI SETUP

On UNIX, set the following parameters in the `snapcreatorgui.conf` located in `/path/to/scServer_v<#>/gui/jetty.x.x.x/etc/snapcreatorgui.conf`:

```
SNAPCREATOR_PATH=/path/to/scServer_v<#>
```

```
SNAPCREATOR_EXEC=/path/to/scServer_v<#>/snapcreator
```

```
SNAPCREATOR_USER=<username>
```

```
SNAPCREATOR_PASS=<encrypted password>
```

Note: The `snapcreatorgui.conf` settings should automatically be set after running `./snapcreator -profile setup`.

After this is configured, you can start the GUI with the following commands:

```
cd /path/to/scServer_v<#>/gui
```

```
/path/to/java -jar gui.jar --port 8080
```

Note: You must be in the `/path/to/scServer_v<#>/gui` directory to start the GUI.

After Jetty is running, you can access the SnapCreator GUI over http as follows:

<http://myserver.mydomain.com:8080>

8 SNAPCREATOR AGENT INSTALLATION

The SnapCreator agent (scAgent) is designed to run on any open systems platform. If the agent is not required; you can choose to run the SnapCreator server on the application server locally.

8.1 EXTRACTING THE ZIP FILE

Extract the ZIP file to `/usr/local` for UNIX or `C:\` for Windows. On Windows, NetApp recommends not having spaces in the install path (i.e., "Program Files").

8.2 COPYING THE BINARY TO SNAPCREATOR ROOT DIRECTORY

The SnapCreator agent distribution contains binaries for Windows, Solaris, Linux, and AIX. The binaries are located under the `/path/to/scServer_v<#>/bin/<os>` directory. Copy the correct binary from `/path/to/scAgent_v<#>/bin/<os>` to `/path/to/scAgent_v<#>`.

Note: If the OS is not available with the distribution, request a compiled version by either contacting a NetApp sales representative or posting a message to the SnapCreator community at:

http://communities.netapp.com/community/products_and_solutions/databases_and_enterprise_apps/snapcreator

8.3 INSTALLING AGENT AS A SERVICE (WINDOWS)

The SnapCreator agent has the ability to run as a daemon under Windows. The agent runs under either the default 9090 port or a user-specified port. To set a non-default port number, configure the following environment variable: `SC_AGENT_PORT`. If the `SC_AGENT_PORT` environment variable is not set, then the agent will default to port 9090.

The agent does not run as a service. If you want the agent to be started across reboots use scheduled-tasks under Start->Program Files->Accessories->System Tools. To run a task, create a `.bat` with `/path/to/scAgent_v<#>/snapcreator -start-agent <port>` in it and use the `.bat` file as your task.

8.4 INSTALLING THE AGENT AS A DAEMON (UNIX)

The SnapCreator agent has the ability to run as a daemon under UNIX. The port that the agent runs under is either the default 9090 or a user-specified port. To set a non-default port number, you must configure the following environment variable: `SC_AGENT_PORT`. If the `SC_AGENT_PORT` environment variable is not set, then the agent will default to port 9090.

To make sure the agent starts when rebooting, a start script must be set up under `/path/to/rc2.d`. Start scripts must begin with `S9x` (`S99scagent`). The start script should contain the following:

To start the agent manually, run the following command:

```
/path/to/scAgent_v<#>/bin/scAgent start
```

To stop the agent manually, run the following command:

```
/path/to/scAgent_v<#>/bin/scAgent stop
```

To query the agent status, run the following command:

```
/path/to/scAgent_v<#>/bin/scAgent status
```

9 SNAPCREATOR SERVER CONFIGURATION

The configuration of SnapCreator is intended to be simple. The configuration file is located in `/path/to/scServer_v<#>/configs/<profile>/<config>.conf`. The configuration file is dynamic, which means that you can create and set variables within the configuration itself using `%` to reference a value. You can have as many configurations as you want, but each time SnapCreator is run, it uses only one configuration. SnapCreator should be scheduled through cron (UNIX) or task manager (Windows). You can edit this file using VI (UNIX) or a text editor of your choice on Windows. In addition, you can use the SnapCreator GUI to edit and manage configuration files. The configuration file is divided into four sections: Basic Configuration, NetApp Options, Other Options, and Additional Modules.

9.1 CREATING A CONFIGURATION FILE (CLI)

First create a new directory or profile for your configuration under `/path/to/scServer_v<#>/configs`. The best practice is to name it after the host or application that you are backing up. The example uses the name `oraprod01`.

The default template is located in `/path/to/scServer_v<#>/configs/default/default.conf`.

1. Create the directory:

```
mkdir /path/to/scServer_v<#>/configs/oraprod01
```

2. Copy or rename the default template to your new configuration directory:

```
cp /path/to/scServer_v<#>/configs/default/default.conf  
/path/to/scServer_v<#>/configs/oraprod01/oraprod01.conf
```

3. Edit your configuration file by using VI (UNIX) or whatever text editor you have in Windows.

If you haven't tried using the GUI, you may want to try it before manually editing the configuration file.

9.2 CREATING A CONFIGURATION FILE (GUI)

First open a Web browser to the following URL: <http://myserver.mydomain.com:8080>

1. Create profile:

You will be asked for the login information that you set up when you ran `./snapcreator --profile setup`. After you are logged in under "Management->Configurations" you will see an "Add Profile" button. Create a new profile. The profile name should relate to the application being backed up. Adding a profile simply creates a directory under the `/path/to/scServer_v<#>/configs` directory.

2. Run the GUI configuration wizard:

After creating the profile, right-click the profile and select New Configuration. This starts the Configuration Wizard, which guides you through creating a SnapCreator configuration for the given profile. You can also rename and delete a configuration by right-clicking and selecting Delete or Rename.

The GUI simply creates the same configuration files you would create using the CLI. You can update and edit configuration files created by the GUI from the CLI or visa versa.

9.3 REFERENCING

Referencing is the ability to define variables within the configuration file. This is done by using the “%” character. A perfect example of this is when you want to use SnapDrive instead of ZAPI to create the Snapshot copies. Because the Snapshot copy names need to be unique, being able to set a dynamic variable could be required. Following is an example from a Windows SnapDrive config. Note the “/.”

```
NTAP_SNAPSHOT_CREATE_CMD1="c:/Program Files/NetApp/SnapDrive/sdcli" snap create
-m fx1b4 -s %SNAME-%SNAP_TYPE-%SNAP_TIME_recent -D E:
```

Table 1 lists the built-in variables that can be referenced.

Table 1) Referencing variables.

Variables	Description
%SNAP_TYPE	This is given when you run SnapCreator and it is your retention policy, that is, daily, weekly, monthly.
%SNAP_TIME	This is a timestamp (YYYYMMDDhhmmss) used in the naming of Snapshot copies to create a guaranteed unique name for every Snapshot copy. It is also used to name the backup reports and in the case of Sybase can be used to name the Sybase transaction logs.
%ACTION	This is the given action when you run SnapCreator: (snap clone_vol clone_lun arch restore delete snaplist clonelist pmset up ossv)
%MSG	This is used to send an error message to another program such as e-mail or Tivoli. It can only be used with the SENDTRAP function.
%USER_DEFINED	This passes a user-defined argument to the SnapCreator config file. A good example would be when you want to integrate with a backup application like NetBackup. You may need to pass the desired NetBackup schedule into the configuration file when you have multiple schedules and want to call NetBackup with a certain schedule. You could also achieve this by having multiple configuration files for the different scenarios.

Note: When combining variables in the SnapCreator configuration file (%SNAME-%SNAP_TYPE-%SNAP_TIME_recent), you should always use the “-“as a separator between the variables, just like in the above SnapDrive example.

9.4 BASIC CONFIGURATION

Table 2) Basic configuration.

Variables	Setting	Description
SNAME (REQUIRED)		Your Snapshot copy naming convention should be unique. Snapshot copies on NetApp are deleted according to the naming convention and retention policy used.
SNAP_TIMESTAMP_ONLY (REQUIRED)	Y/N	If set to Y, Snapshot copies end with YYYYMMDDHHMMSS. Otherwise, new Snapshot copies will be recent and they are renamed to end with YYYYMMDDHHMMSS.
VOLUMES (REQUIRED)		This is the list of source appliances and volumes you want to create a Snapshot copy of, that is, filer1:vol1,vol2,vol3; filer2:vol1;filer3:vol2,vol3.
VFILERS		This is the list of vFiler™ units and their hosting appliances/volumes, that is, vfiler1@filer1:vol1,vol2,vol3;vfiler2@filer2:vol1;vfiler3@filer3:vol2,vol3.
SNAPMIRROR_VOLUMES		This is the list of source appliances and volumes on which you want to perform a SnapMirror update, that is, filer1:vol1,vol2,vol3;filer2:vol1;filer3:vol2,vol3.

Variables	Setting	Description
SNAPMIRROR_CASCADING_VOLUMES		This is the list of SnapVault destination appliances and volumes where, after a SnapVault update, you want to perform a SnapMirror update, that is, <code>sec-filer1:vol1-sec,vol2-sec</code> .
SNAPVAULT_VOLUMES		This is the list of source appliances and volumes on which you want to perform a SnapVault update, that is, <code>filer1:vol1,vol2,vol3;filer2:vol1;filer3:vol2,vol3</code> . Note: For SnapVault and SnapMirror updates to work, the relationships need to exist. SnapCreator does not create the relationships. Note: The hostname(s) in the SnapMirror or SnapVault relationship must be the same as specified in the VOLUMES, SNAPMIRROR_VOLUMES, and SNAPVAULT_VOLUMES options. Also, the host where SnapCreator runs must be able to resolve the hostname(s). Note: Hostnames should be the short hostname (name that appears on storage controller command prompt), not the FQDN.
SNAPVAULT_QTREE_INCLUDE		This is the list of source appliances and qtree paths that should be included in the SnapVault update. Without this option, all qtrees under a volume are vaulted by SnapVault if a relationship exists. Qtrees listed here will be vaulted by SnapVault and the rest will be ignored by SnapVault, for example, <code>filer1:/vol/qtree/qtree1,/vol/volume/qtree2;filer2:/vol/volume/qtree1</code> .
NTAP_USERS (REQUIRED)		This is the list of appliances and their corresponding user names/passwords, that is, <code>filer1:joe/password1;filer2:bob/password2;filer3:ken/password3</code> . Note: If you want to use protected passwords, first run <code>./snapcreator --cryptpasswd</code> and then save the scrambled password in the config file.
NTAP_PWD_PROTECTION (REQUIRED)	Y/N	This is the setting that enables password protection. You must encrypt passwords and save encrypted passwords in config when this option is set to "Y."
TRANSPORT (REQUIRED)	HTTP/HTTPS	This allows you to choose which protocol to use for API communications. Note: HTTPS may require <code>openssl-devel</code> libraries
PORT (REQUIRED)		This is the port you want to use to communicate with the NetApp storage controller(s), normally (80/443).
MANAGEMENT_INTERFACES		This is an interface that you can specify for every physical appliance. All ZAPI communications between SnapCreator and the controller go over this interface. For example: <code>MANAGEMENT_INTERFACES=filer1:filer1-mgmt;filer2:filer2-mgmt</code>
SECONDARY_INTERFACES		For every appliance or vFiler unit, you can specify a secondary interface source/destination pair (name or IP). It is used for SnapMirror and SnapVault updates. For example, <code>filer1:filer1-backup/filer2-backup</code> . Where: filer1 – Storage controller or vFiler unit name filer1-backup – Secondary interface for filer1/vfiler1 filer2-backup – Secondary interface for filer1/vfiler1's SnapMirror or

Variables	Setting	Description
		SnapVault destination Note: The SnapVault/SnapMirror relationships need to be configured to use this secondary interface. SnapCreator does not manage SnapMirror/SnapVault relationships.
LOG_NUM (REQUIRED)		This is the number of .debug and .out reports that SnapCreator has to retain.

9.5 NETAPP OPTIONS

Table 3) NetApp options.

Variables	Setting	Description
SNAPDRIVE	Y/N	This setting allows you to use SnapDrive instead of ZAPI for creating a Snapshot copy.
NTAP_SNAPSHOT_DISABLE	Y/N	This setting disables the SnapCreator from creating a Snapshot copy. The idea of this option is that SnapCreator can handle SnapVault or SnapMirror for SnapManager. For this setting to work, the SnapManager Snapshot copies need to follow this naming convention: <snapshot copy name>-<policy>_recent.
NTAP_SNAPSHOT_CREATE_CMD<#>		This SnapDrive command creates a Snapshot copy and flushes the file system buffers. <#> is a number from 01-99. Note: This is required if you enable the SNAPDRIVE option. ZAPI is still used to do everything else, but the SNAPDRIVE option creates Snapshot copies.
NTAP_SNAPSHOT_RETENTIONS (REQUIRED)		This setting determines the number of NetApp Snapshot copies you want to retain for a given policy, that is, daily:7,weekly:4,monthly:1.
NTAP_SNAPVAULT_RETENTIONS		This setting determines the number of NetApp Snapshot copies on the SnapVault secondary that you want to retain for a given policy, that is, daily:21,weekly:12,monthly:3.
NTAP_SNAPSHOT_RETENTION_AGE		This setting (in days) allows you to define a retention age for Snapshot copies. If configured, Snapshot copies are deleted only if there are more than the number defined in NTAP_SNAPSHOT_RETENTIONS and if they are older than the retention age (in days).
NTAP_SNAPVAULT_RETENTION_AGE		This setting (in days) allows you to define a retention age for SnapVault Snapshot copies. If configured, SnapVault Snapshot copies are deleted only if there are more than the number defined in NTAP_SNAPVAULT_RETENTIONS and if they are older than the retention age (in days).
NTAP_SNAPSHOT_NODELETE	Y/N	This setting overrides NTAP_SNAPSHOT_RETENTIONS and prevents Snapshot copies from being deleted. Leaving this on can cause your NetApp volume to fill up.
NTAP_SNAPVAULT_NODELETE	Y/N	This setting overrides NTAP_SNAPVAULT_RETENTIONS and prevents Snapshot copies from being deleted. Leaving this on can cause your NetApp volume to fill up.
NTAP_SNAPMIRROR_UPDATE	Y/N	This setting allows you to turn off and on the SnapMirror update function.
NTAP_SNAPMIRROR_	Y/N	This setting allows you to turn off and on the cascading SnapMirror update

Variables		Description
CASCADING_UPDATE		function. This is a SnapMirror update using a SnapVault destination volume.
NTAP_SNAPVAULT_UPDATE	Y/N	This setting allows you to turn off and on the SnapVault update function.
NTAP_PM_UPDATE	Y/N	This setting allows you to turn off and on the Protection Manager update that registers SnapCreator Snapshot copies in Protection Manager Note: If NTAP_PM_UPDATE is enabled, you must configure NTAP_DFM_DATA_SET.
NTAP_SNAPVAULT_WAIT		This is the wait time (in minutes) for the SnapVault update process to complete before taking a Snapshot copy on the SnapVault secondary.
NTAP_SNAPMIRROR_WAIT		This is the wait time (in minutes) for the SnapMirror update process to complete before creating a clone on the SnapMirror destination. If NTAP_CLONE_SECONDARY=Y, SnapCreator waits until the SnapMirror update is complete before proceeding. Note: This can only be used in conjunction with NTAP_CLONE_SECONDARY and ACTION clone_vol (only volume clones are currently supported).
NTAP_SNAPMIRROR_USE_SNAPSHOT	Y/N	If enabled, the SnapMirror update uses the newly created Snapshot copy, thus creating a Snapshot copy on the SnapMirror destination. Note: This is required for NTAP_CLONE_SECONDARY because a Snapshot copy is needed in order to create a clone on the SnapMirror destination.
NTAP_SNAPVAULT_MAX_TRANSFER		This is the maximum bandwidth SnapVault is allowed to use in kbps. If it is not set, SnapVault uses the maximum available bandwidth.
NTAP_SNAPMIRROR_MAX_TRANSFER		This is the maximum bandwidth SnapMirror is allowed to use in kbps. If it is not set, SnapMirror uses the maximum available bandwidth.
NTAP_VOL_CLONE_RESERVE	none file volume	This is the space guarantee for a cloned volume.
NTAP_LUN_CLONE_RESERVATION	true false	If set to true, space is reserved for the cloned LUNs if the clone_lun Action is selected. Otherwise, space is not reserved.
NTAP_CLONE_IGROUP_MAP		Specify the appliance, source volume, and an IGROUP. The IGROUP is then mapped to cloned LUNs that reside in the source volume or cloned LUNs that reside in the volume clone, that is, filer1:src_volume1/igroup1;filer2:src_volume2/igroup2. Note: LUN clones assume the same name as their parent volume or LUN and end with _CLONE, that is, if the volume is called myvol, its clone would be myvol_CLONE. Note: Volume clones start with "cl_" and end with "-YYYYMMDDHHMMSS."
NTAP_CLONE_FOR_BACKUP	Y/N	If enabled, clones (volume and LUN) are created and then deleted after the other NetApp operations are complete. Otherwise, clones are deleted before NetApp operations complete. Note: If you are backing up clones to tape, this should be set to Y. If you are doing database refreshes, then you will most likely want to set it to N.
NTAP_CLONE_SECONDARY	Y/N	If enabled, clones are created on the SnapMirror destination after the SnapMirror update finishes.

Variables		Description
		Note: This setting should be used with NTAP_SNAPMIRROR_USE_SNAPSHOT, NTAP_SNAPMIRROR_WAIT, NTAP_CLONE_SECONDARY_VOLUMES , and ACTION clone_vol.
NTAP_CLONE_SECONDARY_VOLUMES		This is a mapping of primary/secondary appliances and the secondary volumes. This is required so that SnapCreator can find the secondary volumes, that is, filer1:filer1-sec/vol1,vol2,vol3;filer2:filer2-sec/vol1;filer3:filer3-sec/vol2,vol3.
NTAP_NUM_VOL_CLONES		This is the number of volume clones you want to keep. This works in a similar way to the snapshot retention policy. Note: This only works for volume clones that require a FlexClone [®] license on the NetApp storage controller.
NTAP_DFM_DATA_SET		This is a list of appliances and Protection Manager data sets to volume correlations, that is, filer1:dataset1/vol1,vol2;filer1:dataset2/vol3.
NTAP_CONSISTENCY_GROUP_SNAPSHOT	Y/N	This setting enables use of consistency groups. By using this setting, you can take a consistent (I/O fencing) Snapshot copy across more than one volume and across multiple storage controllers. Note: Enabling this option requires NTAP_CONSISTENCY_GROUP_TIMEOUT
NTAP_CONSISTENCY_GROUP_TIMEOUT	urgent medium relaxed	This setting controls the wait time for I/O fencing between volumes to finish. Before we can create a consistency group snapshot, all volumes must be quiesced (I/O fencing). The default setting and recommendation is medium (7 seconds); however, depending on how many volumes you have and how many appliances are involved, that may not be long enough. <ul style="list-style-type: none">• Urgent is 2 seconds• Medium is 7 seconds• Relaxed is 20 seconds
NTAP_OSSV_ENABLE	Y/N	This setting enables the Open Systems SnapVault (OSSV) integration. This option must be used in combination with the NTAP_OSSV_HOMEDIR parameter. OSSV is also required on the host running SnapCreator.
NTAP_OSSV_HOMEDIR	/usr/snapvault	The path to the OSSV home directory, that is, /usr/snapvault

9.6 OPERATIONS MANAGER SETTINGS

Table 4) Operations Manager settings.

Variables	Setting	Description
OM_HOST		The name or IP address of the Operations Manager host.
OM_USER		The user name of an Operations Manager user who has permission to create events.
OM_PWD		The password for the above Operations Manager user.
OM_PORT		The port to use for communications with Operations Manager; 8080 is the default http port that the Operations Manager uses.
OM_EVENT_GENERATE	Y/N	This setting enables or disables event creation in Operations Manager

9.7 OTHER OPTIONS

Table 5) Other options.

Variables	Setting	Description
APP_CLONE_FOLLOW_UP_CMD<##>		<p>These are scripts or commands to be executed after the database is cloned, where "##" is a number from 01–99.</p> <p>This can be used to perform application-specific follow-up activities on SAP systems, such as, installing a SAP license, adjusting database tables, like deleting or updating content, and starting up the application.</p>
APP_QUIESCE_CMD<##>		<p>These are scripts or commands that put your application into backup mode where, "##" is a number 01–99.</p> <p>Note: This is ignored if you use APP_NAME because it is handled internally in SnapCreator.</p>
APP_UNQUIESCE_CMD<##>		<p>These are scripts or commands that take your application out of backup mode, where "##" is a number 01–99.</p> <p>Note: This is ignored if you use APP_NAME because it is handled internally in SnapCreator.</p>
ARCHIVE_CMD<##>		<p>This setting handles database archiving or to be used as a wrapper to run other scripts.</p> <p>The archive command, where "##" is a number 01–99.</p>
PRE_APP_QUIESCE_CMD<##>		<p>This is the pre application backup start command, where "##" is a number 01–99.</p>
PRE_NTAP_CMD<##>		<p>This is the pre NetApp Snapshot command, where "##" is a number 01–99; it runs before all NetApp operations.</p>
PRE_APP_QUIESCE_CMD<##>		<p>This is the pre application backup stop command, where "##" is a number 01–99.</p>
PRE_NTAP_CLONE_DELETE_CMD<##>		<p>This is the pre NetApp clone delete command, where "##" is a number 01–99.</p> <p>Note: The purpose of NetApp clone delete command(s) is to call a mount script or commands so that cloned LUNs can be mounted for the purpose of backing up (probably to tape).</p>
PRE_EXIT_CMD<##>		<p>This is an optional command that execute after a fatal error occurs but before SnapCreator exits. This is useful for returning everything to the state it was before SnapCreator ran.</p> <p>Note: The purpose is to return an application into normal operation mode before SnapCreator exists due to an error.</p> <p>Note: This is ignored if you use APP_NAME because it is handled internally in SnapCreator.</p>
PRE_RESTORE_CMD<##>		<p>This is an optional command that can be run before you enter an interactive restore. This allows you to interact with the application being restored. Before you do a restore, for example, you might want to shut down the application.</p>
PRE_CLONE_CREATE_CMD<##>		<p>This is an optional command that can be run before ZAPI cloning operations occur, where "##" is a number 01–99.</p>
PRE_APP_CLONE_CREATE_CMD<##>		<p>This is an optional command that can be run before any app cloning operations are run, where "##" is a number 01–99.</p>

Variables	Setting	Description
POST_APP_QUIESCECMD<##>		This is a post application backup start command, where “##” is a number 01–99
POST_NTAP_CMD<##>		This is a post NetApp command, where “#” is a number 01–99. This runs after all NetApp operations are complete.
POST_APP_UNQUIESCE_CMD<##>		This is a post application backup stop command, where “##” is a number 01–99.
POST_NTAP_DATA_TRANSFER_CMD<##>		This is a post data transfer command runs after SnapVault or SnapMirror transfer, where “##” is a number 01–99.
POST_RESTORE_CMD<##>		This is an optional command that can be run after you complete an interactive restore. It allows you to interact with the application being restored. After your restore is complete, you might want to start the application.
POST_CLONE_CREATE_CMD<##>		This is an optional command commands that can be run after ZAPI cloning operations occur, where “##” is a number 01–99. These would be things like mounting cloned file systems.
POST_APP_CLONE_CREATE_CMD<##>		This is an optional command that can be run after any app cloning operations are run, where “##” is a number 01–99. These would be things like changing DB ID of your cloned DB.
NTAP_ASUP_ERROR_ENABLE	Y/N	This setting enables SnapCreator error messages to also log an auto support message on the NetApp storage controller. SnapCreator always creates an info auto support message when the backup has started and completed.
SENDTRAP		This command interfaces with your monitoring software or e-mail allowing you to pass alerts generated from SnapCreator into your own monitoring infrastructure. The %MSG variable is the message sent from SnapCreator. Following is an example of how to send email on a UNIX system: SENDTRAP=/usr/bin/mailx -s %MSG myaddress@mydomain.com </dev/null
SUCCESS_TRAP		This command interfaces that interfaces with your monitoring software or e-mail allowing you to pass the success message generated from SnapCreator into your own monitoring infrastructure. The %SUCCESS_MSG variable is the success message for SnapCreator. Following is an example of how to send email on UNIX system: SUCCESS_TRAP=/usr/bin/mailx -s %SUCCESS_MSG myaddress@mydomain.com </dev/null
SUCCESS_MSG		Upon a successful SnapCreator backup, this setting logs the message defined and also sends it to SUCCESS_TRAP, if defined, or to SENDTRAP, if SENDTRAP is defined.

9.8 CLIENT/SERVER CONFIGURATION

Table 6) Client/server configuration.

Variables	Setting	Description
SC_AGENT	<hostname or IP >:<port>	SnapCreator has the capability to perform tasks on remote hosts. A task is either a defined module (parameter APP_NAME) or a command specified with the parameters *_CMD*, for example, NTAP_SNAPSHOT_CREATE_CMD01. To specify a remote host, enter its name or IP address followed by a colon and the port the SnapCreator Agent is listening on. On the remote host, start the SnapCreator Agent: <path to scAgent_v<#>>/snapcreator --start-agent <port>.
SC_CLONE_TARGET	<hostname or IP of the clone target>:<port>	SnapCreator has the capability to perform clone operations. Using the action clone_vol in combination with {PRE POST}_CLONE_CREATE_CMDxx to handle the storage objects on the remote side (for example, mounting/unmounting file systems), to specify a clone target, enter its name or IP address followed by a colon and the port, the SnapCreator Agent listening on.
SC_AGENT_TIMEOUT	Time in seconds	This setting specifies the timeout in seconds. The implemented client/server architecture uses a timeout mechanism. This means that if the client does not respond in the specified interval, the server fails with a timeout message. However, the task on the client is left untouched (not aborted) and needs further investigation. On a server with high load or known long-running tasks like own scripts or complex SnapDrive operations, it might be necessary to extend the timeout and adapt this value to your own needs. By default, a timeout of 300 seconds is used.

9.9 ADDITIONAL MODULES

Currently, SnapCreator supports the following applications: Oracle, DB2, MaxDB, MySQL, Lotus Notes, PostgreSQL, and Sybase.

Table 7) Additional modules.

Variables	Setting	Description
APP_NAME	oracle db2 maxdb mysql notes sybase	This setting determines which application is being backed up. SnapCreator has built-in support for the listed applications. You can either use APP_NAME or configure APP_QUIESCE_CMDXX, APP_UNQUIESCE_CMDXX, and PRE_EXIT_CMDXX.. If the application is not directly supported in SnapCreator, you can run your own application quiesce/unquiesce commands or scripts. APP_QUIESCE_CMD01=/path/to/quiesceCMD APP_UNQUIESCE_CMD01=/path/to/unquiesceCMD PRE_EXIT_CMD01=/path/to/unquiesceCMD
VALIDATE_VOLUMES	DATA	SnapCreator validates that all volumes where the database resides are in fact part of the backup. Currently, there are some limitations. Only NFS is supported and only for db2, maxdb, and Oracle. Currently, this option only checks data files only for the above databases. Going forward, support for more data types like LOG will be added.

MAXDB MODULE

MaxDB is the preferred database used by SAP because SAP owns MaxDB. This module supports both Windows and UNIX systems for MaxDB 7.x. The MaxDB module uses the `dbmccli` command to communicate with the database.

Table 8) MaxDB options.

Parameter	Setting	Description
APP_NAME	maxdb	This setting determines which application is being backed up.
XUSER_ENABLE	Y/N	This setting enables the use of an xuser for maxdb. Using xuser does not require the password of the database user.
DBMCLICMD		The path to the MaxDB dbmccli command; if not set, dbmccli on the search path is used.
SQLCLICMD		The path to the MaxDB sqlcli command; if not set, sqlcli on the search path is used.
MAXDB_UPDATE_HIST_LOG	Y/N	This setting updates the MaxDB history log.
MAXDB_DATABASES		This is a comma-separated list of the maxdb databases that you wish to backup.
HANDLE_LOGWRITER		This setting forces a suspend/resume logwriter; suspending the logwriter is only necessary in deployments in which the data files reside on more than one volume.

OPTIONAL CLONING OPTIONS FOR MAXDB

MAXDB_CLONE_META

Parameter	Description
source_sid	Database SID used on the source side.
target_sid	Database SID used on the target side.
target_db_path	Top-level directory of the MaxDB installation.
dbm_user	Target database management user.
dbm_passwd	Password or authentication key of dbm_user.
dbadmin_user	Target SQL user with DBA privileges.
dbadmin_passwd	Password or authentication key of dbadmin_user.
os_user	Target operating system user.
os_group	Primary group of os_user.

For example:

```
MAXDB_CLONE_META=PRD:QAS, /sapdb/QAS, dbm/secret, dbadmin/secret, sdb/sdba
```

MAXDB_CLONE_ADAPT_FS

Parameter	Description
target_sid	Database SID used on the target side.
path	This is a full qualified path. File access permissions will be changed recursively. Wildcards are allowed; multiple paths can be specified, separated by a comma.

For example:

```
MAXDB_CLONE_ADAPT_FS=QAS: /sapdb/QAS/sapdata*
```

MAXDB_CLONE_RENAME_USER

Parameter	Description
target_sid	Database SID used on the target side.
source_username	Schema/Table owner on the source side.
target_username	Schema/Table owner on the target side; multiple sources/target users can be specified, separated by a comma.

For example:

```
MAXDB_CLONE_RENAME_USER=QAS:sapprd/sapqas ,saprddb/sapqasdb
```

MAXDB_CLONE_RESIZE_LOG

This option is only available on MaxDB >= 7.8.

Parameter	Description
target_sid	Database SID used on the target side.
new log size	Log size in pages.

For example:

```
MAXDB_CLONE_RESIZE_LOG=QAS:2000
```

MAXDB_SOURCE_PRESERVE_PARAM

These are parameters from the source set on the target.

Parameter	Description
source_sid	Database SID used on the source side.
parameter	Parameter on the source that should be enabled on the target; multiple parameters can be specified, separated by a comma.

For example:

```
MAXDB_SOURCE_PRESERVE_PARAM=PRD:MaxLogVolumes ,MaxDataVolumes
```

ORACLE MODULE

Oracle Database is an enterprise database. The Oracle Module supports both Windows and UNIX systems. The Oracle Module only supports Oracle Database10g™ or higher. The entire database is put into backup mode and that is only supported starting with Oracle10g. The Oracle Module uses sqlplus to communicate with the database.

Table 9) Oracle options.

Parameter	Settings	Description
APP_NAME	oracle	The application name.
ORACLE_DATABASES		A list of Oracle Databases and the user name, i.e., db1:user1;db2:user2.
SQLPLUS_CMD		The path to the sqlplus command.
CNTL_FILE_BACKUP_DIR		The path to the directory where we should store backup control files (Oracle user must have permissions).
ORA_TEMP		The path to a directory for storing temp file, that is, /tmp (Oracle user must have permissions).
ARCHIVE_LOG_ONLY	Y/N	Informs Oracle Module to only do a switch log. This setting is useful if you are handling archive logs separate from data backup.

9.10 POSTGRESQL MODULE

PostgreSQL is an open source database and an alternative to Oracle and SQL Server. The PostgreSQL module supports both Windows and UNIX systems running PSQL 8.3.x or 8.4.x. The PostgreSQL module uses the psql CLI interface to communicate with the database.

POSTGRESQL OPTIONS

Table 10) PostgreSQL options.

Parameter	Settings	Description
APP_NAME	postgres	The application name.
POSTGRES_DATABASES		A list of PostgreSQL database(s) and the username i.e., db1:user1;db2:user2
PSQL_CMD		The path to the to the psql cli command
NO_PASSWORD_OPTION	Y/N	Option supported by PostgreSQL 8.4 and higher which doesn't require PGPASSWORD to be set.

MYSQL MODULE

MySQL is a popular database alternative to Oracle and SQL Server®. The MySQL module supports both Windows and UNIX systems running MySQL 5.x. The MySQL module uses Net-MySQL to communicate with the database.

Table 11) MySQL options.

Parameter	Settings	Description
APP_NAME	mysql	The application name.
MYSQL_DATABASES		A list of MySQL database(s) and the username/password i.e., db1:user1/pwd1;db2:user2/pwd2
HOST		Name of the host where the databases are located, i.e., localhost
PORTS	Y/N	A list of database(s) and the ports they are listening on, i.e., db1:3307;db2:3308
MASTER_SLAVE		If the database(s) is (are) part of the MASTER7SLAVE environment.

DB2 MODULE (UNIX AND WINDOWS)

DB2 is a high-end database alternative to Oracle and SQL Server. The DB2 module at this time will only run on UNIX systems. The DB2 module uses the db2 command to communicate with the database.

Table 12) DB2 options.

Parameter	Settings	Description
APP_NAME	db2	The application name.
DB2_DATABASES		A list of db2 database(s) and the username, i.e., db1:user1;db2:user2
DB2_CMD		The path used by the db2 command to interact with the database. Note: For Windows, you need to first give a path to db2cmd.exe below is an example. UNIX: DB2_CMD=/opt/ibm/sqllib/bin/db2 Windows: DB2_CMD=X:\IBM\SQLLIB\BIN\db2cmd.exe /c /w /i X:\IBM\SQLLIB\BIN\db2.exe

Optional Cloning for DB2 (UNIX)

Table 13) Optional cloning for DB2 (UNIX).

Parameter	Description
DB2INIDB_CMD	Path to the db2inidb command. If not specified, sqllib/adm/db2inidb will be used.
DB2RELOCATEDB_CMD	Path to the db2relocatedb command. If not specified, sqllib/bin/db2relocatedb will be used.
DB2_CLONE_META	source_sid: Database SID used on the source side target_sid: Database SID used on the target side target_db_path: Database path on the target side target_os_user: DB2 instance owner on the target side target_os_passwd: Group of DB2 instance owner For example: DB2_CLONE_META=PRD:QAS, /db2/QAS, db2qas/dbqasadm
DB2_CLONE_RELOCxx	Relocation information for the db2relocatedb command, e. g. cloning PRD to QAS target_sid: Name of the target database entry: Entry of the file containing the relocate information For example: DB2_CLONE_RELOC01=QAS:DB_NAME=PRD:QAS DB2_CLONE_RELOC02=QAS:DB_PATH=/db2/PRD:/db2/QAS DB2_CLONE_RELOC03=QAS:DB_INSTANCE=db2prd:db2qas DB2_CLONE_RELOC04=QAS:LOG_DIR=/db2/PRD/log_dir,/db2/QAS/log_dir DB2_CLONE_RELOC05=QAS:CONT_PATH=/db2/PRD/sap*,/db2/QAS/sap*
DB2_CLONE_RELOC_FILE	Relocation information for the db2relocatedb command in an external file on the clone target target_sid: Name of the target database fqdn: Fully qualified filename containing the relocation information Multiple target databases (and their relocation information) can be either specified with the DB2_CLONE_RELOC_FILE parameter, separated by a semicolon or by extending

	<p>this parameter with a number (e. g. DB2_CLONE_RELOC_FILE01). A combination of both is allowed.</p> <p>Note: If DB2_CLONE_RELOC_FILE and DB2_CLONE_RELOC## is specified for the same database, DB2_CLONE_RELOC_FILE will be used and DB2_CLONE_RELOC## will be ignored.</p>
DB2_CLONE_PARAMxx	<p>Target database specific database configuration parameters</p> <p>target_sid: Name of the target database</p> <p>parameter=value: Pair of parameter and value to be set on target side multiple pairs can be specified, comma separated</p> <p>For example:</p> <pre>DB2_CLONE_PARAM01=QAS:LOGARCHMETH1=DISK:/db2/QAS/logarch1,LOGARCHMETH2=DISK:/db2/QAS/logarch2</pre>

SYBASE MODULE

Sybase ASE is a high-end database alternative to Oracle and SQL Server. The Sybase module uses the isql command to communicate with the database.

Table 14) Sybase options.

Parameter	Description
APP_NAME	sybase
SERVER	Host name of the server running Sybase
SYBASE_DATABASES	A list of Sybase database(s) and their username/password i.e., db1:user1/pwd1;db2:user2/pwd2
ISQL_CMD	Path to the isql command i.e.: /path/to/isql
SYBASE	Path to the Sybase home directory i.e.: /path/to/sybase_home
LOGDIR	Path to the location where logs should be written i.e.: /path/to/logdir

LOTUS DOMINO MODULE

Lotus Domino is a collaboration solution that provides reliable and security-rich e-mail and calendaring capabilities as well as a powerful platform for collaborative business application. The Lotus Domino module will run on both Windows and UNIX systems. The module uses CLI commands as opposed to API calls, meaning that the transaction logs are not integrated and this is a point in time backup only.

For consistency purposes it is suggested that you make regular offline backups while the Domino server is stopped, perhaps during your nightly or weekly maintenance window. Online backups work very well, but consistency is not guaranteed. This module only handles online backups only at this time.

Note: SnapCreator must be run as the domino_admin user.

Table 15) Lotus Domino options.

Parameter	Description
APP_NAME	notes
DOM_ADMIN	The Domino administration user (UNIX only)
SERVER_CMD	Path to the Domino "server" (UNIX) or nserver (Windows) command
LOTUS	Path to the Lotus Domino bin directory where the "server" or nserver exists

DOMINO_DATA_PATH	Path to the Domino data directory (UNIX only)
DOMINO_SERVICE_NAME	The Domino service name (Windows only)
RC_DOMINO_CMD	Path to the domino rc script <code>rc_domino_script</code> (Unix only)

ARCHIVE LOG MODULE

The Archive Log Module works for any database, and is database independent. This module does not communicate with the database. Its purpose is to delete archive logs that are older than the configured threshold value (in days).

Table 16) Archive log options.

Parameter	Setting	Description
ARCHIVE_LOG_ENABLE	Y/N	Setting that enables archive log management (deletion of old archive logs).
ARCHIVE_LOG_RETENTION		Retention in days of how old the archive logs should be; epoch time is used and is precise to the second.
ARCHIVE_LOG_DIR		Path to the directory that contains the archive logs.
ARCHIVE_LOG_EXT		File extension of the archive logs, i.e., if the archive logs are 10192091019.log you would set this to "log"; the search pattern we use is <something>.<extension>

10 SNAPCREATOR AGENT CONFIGURATION

SnapCreator server access is the only parameter to be configured on the agent. By default, the SnapCreator server is not allowed to execute any commands or scripts on the agent. It only has access to the built-in application modules. If pre/post commands or app commands are required for mounting or any other operations then those commands or scripts must be added to the `/path/to/scAgent_v<#>/config/agent.conf` file. Regular expressions can be used to be more restrictive. Each command or script should be added to the file as follows:

```
echo
/bin/mount
/path/to/myscript
```

In this example the agent will allow the SnapCreator server to run `echo`, `/bin/mount`, or `/path/to/myscript` in addition to of course the application modules. If the application modules are not used and the agent is desired, allow the application quiesce and unquiesce commands for application consistency.

```
^/bin/mount.*filer:/vol/cl_.* /mnt/
```

Additionally the wildcard "*" can be used to allow all commands but NetApp does not recommend it for security reasons.

11 RUNNING SNAPCREATOR SERVER

NetApp recommends scheduling the SnapCreator server from the UNIX cron or the Windows task manager. The binary for the SnapCreator server and agent are the same; only the packaging is different.

Note: When scheduling through the Windows task manager, create a .bat file, since task manager doesn't like programs that require input.

```
./snapcreator --profile <Config> --action <Action> --policy <Policy> <Optional Arguments>
```

Table 10 lists the parameters.

Table 17) SnapCreator server parameters.

Parameter	Description
--profile <Profile>	The name of the profile should be the name you gave to the directory/file without the .conf suffix. This is your profile; you can create other config files under the same profile directory but then you need to use the --config option explained below to use that alternate config instead of the default config associated with the profile.
list	Displays all configs known to SnapCreator. Do not use "list" as a profile name as it will only display configs. For example: <pre>./snapcreator --profile list</pre>
setup	Runs the SnapCreator setup that configures both the GUI and SnapCreator itself. For example: <pre>./snapcreator --profile setup</pre>
--action <Action> (snap clone_vol clone_lun arch restore delete snaplist clonelists pmsetup ossv)	There are currently 10 possible actions: create a Snapshot copy (snap), perform an OSSV backup (ossv), take a Snapshot copy as well as a volume clone (clone_vol), take a Snapshot copy as well as a LUN clone (clone_lun), handle only archiving for an application (arch), do an interactive restore (restore), do an interactive delete (delete), list snapshots for a given policy (snaplist), list volume clones (clonelists), configure a Protection Manager data set (pmsetup).
snap	Takes a Snapshot copy and whatever else you have configured. For example: <pre>./snapcreator --profile <Profile> --action snap --policy <Policy> <Optional Arguments></pre>
ossv	Uses OSSV to perform the backup; no Snapshot copy is created. For example: <pre>./snapcreator --profile <Profile> --action ossv --policy <Policy> <Optional Arguments></pre>
clone_vol	Takes a Snapshot copy plus whatever else is configured and then uses that Snapshot copy to make a volume clone. This does require a FlexClone license. Once the volume is cloned, an IGROUP is mapped to all the LUNs that reside in the cloned volume using the NTAP_CLONE_IGROUP_MAP option in the configuration file. Note: The format of the volume clone names are as follows: <pre>cl_<source volume>_YYYYMMDDhhmmss</pre> <pre>./snapcreator --profile <Profile> clone_vol --policy <Policy> <Optional Arguments></pre> If the SC_CLONE_TARGET parameter is set as well as the options necessary for the database clone handling, after cloning the volume, the database cloning will be done.

Parameter	Description
clone_lun	<p>Creates a Snapshot copy plus whatever else is configured and then uses that Snapshot copy to make a LUN clone of all LUNs that reside in the source volume. Once the LUNs are cloned, an IGROUP is mapped to all the cloned LUNs that reside in the source volume using the NTAP_CLONE_IGROUP_MAP option in the configuration file.</p> <p>Note: Due to the potential for Snapshot copies to be locked by cloned LUNs, this is not the recommended method, but it comes down to whether or not the customer owns a FlexClone license. If he/she does, use clone_vol, which uses FlexClone. LUN clone names are always as follows: <source LUN>_CLONE</p> <pre>./snapcreator --profile <Profile> --action clone_lun --policy <Policy> <Optional Arguments></pre>
arch	<p>Provides archiving functionality for an application. No NetApp commands or functions will be preformed. The idea is to allow one to do archiving outside a backup. At this time you need to provide the archiving script or command so this is just a wrapper to provide your scripts or commands with SnapCreator error handling.</p> <p>For example:</p> <pre>./snapcreator --profile <Profile> --action arch <Optional Arguments></pre>
restore	<p>Provides an interactive restore. You can perform a volume, file, or SnapVault restore. File restore is for LUNs. A volume restore is only recommended in case of a disaster otherwise you should always recover files. For NAS you can do this by simply copying files out of the ~snapshot dir so there is no need to handle NAS file restore with SnapCreator. A SnapVault restore will restore a qtree on secondary storage back to primary storage either original or alternate location.</p> <p>For example:</p> <pre>./snapcreator --profile <Profile> --action restore --policy <Policy> <Optional Arguments></pre>
delete	<p>Guides you through an interactive Snapshot delete menu for a given SnapCreator policy.</p> <p>For example:</p> <pre>./snapcreator --profile <Profile> --action delete --policy <Policy> <Optional Arguments></pre>
snaplist	<p>Lists all Snapshot copies that are being managed by SnapCreator. It displays Snapshot copies on primary as well as secondary storage.</p> <p>For example:</p> <pre>./snapcreator --profile <Profile> --action snaplist <Optional Arguments></pre>
clonelist	<p>Lists all volume clones that are being managed by SnapCreator. It show volume clones on primary as well as secondary storage.</p> <p>For example:</p> <pre>./snapcreator --profile <Profile> --action clonelist <Optional Arguments></pre>
dpstatus	<p>Shows detailed information about SnapVault or SnapMirror relationships.</p> <p>For example:</p> <pre>./snapcreator --profile <profile> --action dpstatus</pre>

Parameter	Description
pmsetup	<p>Creates a Protection Manager Dataset for given config.</p> <p>For example:</p> <pre>./snapcreator --profile <Profile> --action pmsetup <Optional Arguments></pre> <p>Note: After create the Protection Manager dataset you will need to go into PM and configure it which involves adding relationships and defining the protection policy.</p>
--policy <Policy>	<p>The name of the Snapshot policy defined in NTAP_SNAPSHOT_RETENTIONS and possibly NTAP_SNAPVAULT_RETENTIONS. You can name the Snapshot policy whatever you want but it is important that the name you pass SnapCreator as --policy be the exactly same and defined in NTAP_SNAPSHOT_RETENTIONS as well as possibly NTAP_SNAPVAULT_RETENTIONS if you are using SnapVault. You can also have as many of these as you want all with different retentions.</p> <p>Note: If you use SnapDrive to create the Snapshot copies, use lower case. SnapDrive (at least Windows) cannot differentiate between uppercase and lowercase.</p>

OPTIONAL ARGUMENTS

Table 18) Optional arguments.

Parameter	Description
--config	<p>Allows you to specify an alternate config file located under the /path/to/scServer_v<#>/<Config> directory.</p> <p>For example:</p> <pre>./snapcreator --profile <Profile> --action <Action> --policy <Policy> --config <Alternate Config></pre> <p>Note: This option is required when the profile /path/to/scServer_v<#>/configs/<profile> and the config does not match /path/to/scServer_v<#>/configs/<profile>/<config>.conf</p>
--global	<p>Allows you to use a global config file. There are two possible global config files you can have.</p> <p>Environment Global Config: configs/global.conf</p> <p>Profile Global Config: configs/<profile>/global.conf</p> <p>Parameters are loaded first from environment global, then the profile global, and finally your actual config file. Any duplicate parameters will simply be written over by your actual config file.</p> <p>For example:</p> <pre>./snapcreator --profile <Profile> --action <Action> --policy <Policy> --config <Alternate Config> --global</pre>
--verbose	<p>Displays all logging information to STDOUT. This is an optional setting and is mostly for testing and running SnapCreator manually.</p> <p>For example:</p> <pre>./snapcreator --profile <Profile> --action <Action> --policy <Policy> --config <Alternate Config> --verbose</pre>

Parameter	Description
--debug	Displays all logging information to STDOUT and the log file. This is an optional setting and is for debugging problems. For example: <pre>./snapcreator --profile <Profile> --action <Action> --policy <Policy> --config <Alternate Config> --verbose --debug</pre>
--version	Prints the SnapCreator version. For example: <pre>./snapcreator --version</pre>
--cryptpasswd	Encrypts a password for storing in a config file. For example: <pre>./snapcreator --cryptpasswd</pre>

12 RUNNING SNAPCREATOR AGENT

The SnapCreator agent runs as a daemon on UNIX and Windows. The binary for the SnapCreator server and agent are the same; it is only the packaging that is different.

The arguments are as follows:

```
./snapcreator -start-agent <port> <Optional Arguments>
```

Where:

- `start-agent <port>`: Starts the SnapCreator agent daemon on the port specified. If no port is specified port 9090 is used.
- `<Optional arguments>` are:
 - `--debug`: Displays all logging information to `STDOUT` and the log file. This is an optional setting and is for debugging problems. In agent mode we will not fork a process so that all agent information gets displayed to `STDOUT`.

Following is an example of the argument:

```
./snapcreator -start-agent <9090> --debug
```

Note: On UNIX, to start the agent automatically, use the `/path/to/scAgent_v<#>/bin/scAgent` script. Windows does not have a start script.

13 SNAPCREATOR EXAMPLES

SnapCreator offers the flexibility for performing simple procedures such as creating a Snapshot copy as well as complex procedures such as SnapMirror destination volume. This section lists the examples of supported SnapCreator configurations.

13.1 NETAPP SNAPSHOT (ALWAYS REQUIRED)

This solution uses SnapCreator to create a consistent Snapshot copy. As a Snapshot copy is the basis for all operations, all examples require the below settings.

Table 19) Snapshot configuration.

Configuration Option	Setting
SNAME	Snapshot copy name
VOLUMES	filer1:vol1,vol2,vol3
NTAP_USERS	filer1:snapadm/mypassword
TRANSPORT	HTTP
PORT	80
LOG_NUM	10
NTAP_PWD_PROTECTION	N
NTAP_SNAPSHOT_RETENTIONS	daily:7

Call SnapCreator with the following options:

```
./snapcreator --profile <Profile Name> --action snap --policy daily <Optional Arguments>
```

Note: These settings are always required. In this example, we have only created one retention policy set to daily, which will retain seven Snapshot copies.

13.2 NETAPP APPLICATION-CONSISTENT BACKUP

This solution combines an application quiesce and unquiesce with a NetApp Snapshot copy.

Table 20) Application-consistent backup configuration.

Configuration Option	Setting
APP_NAME	oracle db2 maxdb mysql notes Sybase

Call SnapCreator with the following options:

```
./snapcreator --profile <Profile Name> --action snap --policy <Policy Name> <Optional Arguments>
```

13.3 NETAPP SNAPSHOT AND BACKUP TO TAPE

This solution uses SnapCreator to create and clone a Snapshot copy, mount this clone, and then back up the mountpoint to tape using third-party backup software.

Table 21) Snapshot and backup to tape configuration.

Configuration Option	Setting
NTAP_CLONE_FOR_BACKUP	Y
NTAP_CLONE_SECONDARY	N
NTAP_CLONE_IGROUP_MAP	filer1:src_volume1/igroup1
NTAP_VOL_CLONE_RESERVE	none
NTAP_NUM_VOL_CLONES	1
NTAP_LUN_CLONE_RESERVATION	false
PRE_NTAP_CLONE_DELETE_CMD1	Mount script or SnapDrive command
PRE_NTAP_CLONE_DELETE_CMD2	CLI command to start backup of mountpoint

To clone a volume, call SnapCreator with the following options:

```
./snapcreator --profile <Profile Name> --action clone_vol --policy <Policy Name> <Optional Arguments>
```

To clone a LUN, call SnapCreator with the following options:

```
./snapcreator --profile <Profile Name> --action clone_lun --policy <Policy Name> <Optional Arguments>
```

13.4 NETAPP BACKUP TO DISK (OSSV)

This solution uses the OSSV client to perform a SnapVault update or create a Snapshot copy on the secondary storage.

Table 22) OSSV configuration.

Configuration Option	Setting
NTAP_OSSV_ENABLE	Y
NTAP_OSSV_HOMEDIR	/usr/snapvault

Call SnapCreator with the following options:

```
./snapcreator -profile <Profile Name> --action ossv --policy <Policy Name> <Optional Arguments>
```

13.5 NETAPP SNAPSHOT AND BACKUP TO DISK (SNAPVAULT)

This solution uses SnapCreator to create a Snapshot copy, perform a SnapVault update, and create a Snapshot copy of the SnapVault secondary.

Table 23) Snapshot and backup to disk configuration.

Configuration Option	Setting
SNAPVAULT_VOLUMES	filer1:vol1,vol2,vol3
NTAP_SNAPVAULT_RETENTIONS	Daily:30
NTAP_SNAPVAULT_RETENTION_AGE	30
NTAP_SNAPVAULT_MAX_TRANSFER	kbps or blank

NTAP_SNAPVAULT_UPDATE	Y
NTAP_SNAPVAULT_WAIT	minutes
NTAP_SNAPVAULT_NODELETE	N

Call SnapCreator with the following options:

```
./snapcreator --profile <Profile Name> --action snap --policy <Policy Name>
<Optional Arguments>
```

13.6 NETAPP SNAPSHOT AND BACKUP TO DISK (SNAPMIRROR)

This solution uses SnapCreator to create a Snapshot copy and perform a SnapMirror update.

Table 24) Snapshot and backup to disk configuration.

Configuration Option	Setting
SNAPMIRROR_VOLUMES	filer1:vol1,vol2,vol3
NTAP_SNAPMIRROR_USE_SNAPSHOT	Y
NTAP_SNAPMIRROR_MAX_TRANSFER	kpbs or blank
NTAP_SNAPMIRROR_UPDATE	Y

Call SnapCreator with the following options:

```
./snapcreator --profile <Profile Name> --action snap --policy <Policy Name>
<Optional Arguments>
```

13.7 NETAPP SNAPSHOT AND DATABASE REFRESH (PRIMARY)

This solution uses SnapCreator to create and clone a Snapshot copy.

Table 25) Snapshot and database refresh configuration.

Configuration Option	Setting
NTAP_CLONE_FOR_BACKUP	Y
NTAP_CLONE_IGROUP_MAP	filer1:src_volume1/igroup1
NTAP_VOL_CLONE_RESERVE	none
NTAP_LUN_CLONE_RESERVATION	false
POST_NTAP_CMD1	Mount Script or SnapDrive command to mount clone
POST_NTAP_CMD2	CLI command or script to start cloned database

To clone a volume, call SnapCreator with the following options:

```
./snapcreator --profile <Profile Name> --action clone_vol --policy <Policy Name>
<Optional Arguments>
```

To clone a LUN, call SnapCreator with the following options:

```
./snapcreator --profile <Profile Name> --action clone_lun --policy <Policy Name>
<Optional Arguments>
```

13.8 NETAPP SNAPSHOT AND DATABASE REFRESH (SECONDARY)

This solution uses SnapCreator to create a Snapshot copy, perform a SnapMirror update, and clone the SnapMirror destination volume.

Table 26) Snapshot and database refresh configuration.

Configuration Option	Setting
SNAPMIRROR_VOLUMES	filer1:vol1,vol2,vol3
NTAP_SNAPMIRROR_USE_SNAPSHOT	Y
NTAP_SNAPMIRROR_MAX_TRANSFER	kpbs or blank
NTAP_SNAPMIRROR_UPDATE	Y
NTAP_CLONE_SECONDARY_VOLUMES	filer1:filer1-sec/vol1,vol2,vol3
NTAP_CLONE_SECONDARY	Y
NTAP_CLONE_FOR_BACKUP	Y
NTAP_VOL_CLONE_RESERVE	none
NTAP_NUM_VOL_CLONES	1
POST_NTAP_CMD1	Mount script or SnapDrive command to mount cloned
POST_NTAP_CMD2	CLI command or script to start cloned database

For volume clone call SnapCreator with the following options:

```
./snapcreator --profile <Profile Name> --action clone_vol --policy <Policy Name> <Optional Arguments>
```

13.9 NETAPP SNAPSHOT AND PROTECTION MANAGER UPDATE

When using SnapCreator to create a Snapshot copy, register the Snapshot copy with Protection Manager and then use Protection Manager to manage SnapVault or SnapMirror.

Table 27) Snapshot and Protection Manager configuration.

Configuration Option	Setting
NTAP_PM_UPDATE	Y
NTAP_DFM_DATA_SET	filer1:dataset1/vol1,vol2;filer1:dataset2/vol3
NTAP_SNAPSHOT_NODELETE	Y

Call SnapCreator with the following options:

```
./snapcreator -profile <Profile Name> --action snap --policy <Policy Name> <Optional Arguments>
```

Note: Both Operations Manager and Protection Manager are required. In addition, you need to configure a dataset created by SnapCreator within Protection Manager and create a schedule for that dataset (secondary). SnapCreator or PM can delete primary Snapshot copies. Protection Manager will manage all secondary Snapshot copies and their retentions.

14 TROUBLESHOOTING

SnapCreator writes to several logs, and these are the first places to begin troubleshooting. There are three logs for every profile and configuration: out, error, and debug. The out and debug logs are retained as defined by the `LUG_NUM` value in the configuration file while the error log is always appended to. All logs are written to the `/path/to/scServer_v<#>/logs/<profile>` directory.

OUT LOG: The out or output log contains information about everything SnapCreator does. Messages can be INFO, CMD, OUTPUT, DEBUG, WARN, or ERROR. Anything with INFO is considered to be a standard, normally occurring operation. Anything with CMD is an external command or script that SnapCreator ran (according to configuration) and we do log the return code from the command or script. Typically these are PRE, POST, or APP Quiesce/Unquiesce commands. Anything with OUTPUT is a ZAPI call. Anything with DEBUG is debug information which is only displayed when using option `--debug`. Anything with WARN is considered to draw your attention, but, again under normal circumstances, it is considered to be normal activity and should require no action (an example is when we delete Snapshot copies). Anything with ERROR is a problem and will most likely require manual action to fix. SnapCreator will exit on any ERROR, so it is important to fix whatever problem occurred before it runs again. SnapCreator will not automatically fix problems although you can be proactive by telling SnapCreator what to do before one exists via the `PRE_EXIT_CMD` defined in the configuration file.

DEBUG LOG: The debug log contains the return from any external commands or scripts run by SnapCreator. It is very important if you call other scripts through SnapCreator that you properly set up exit codes and output for those scripts. You should never exit with status 0 if a problem indeed occurred.

ERROR LOG: The error log contains a history of all the ERROR events for a given configuration. This is a good place to see what kinds of errors happened in the past so you can correlate things better and get a historical perspective. You can also monitor this log file and use it as a means to integrate SnapCreator with your monitoring application.

SnapCreator Agent

The SnapCreator agent does not create any log files. If problems occur while running in the client/server mode, start the agent in the debug mode to print debug messages on `stdout`.

In debug mode, the agent does not run in a separate process. The debug can be enabled by specifying the additional `--debug` option:

```
/path/to/scAgent_v<#>/bin/scAgent --debug
```

14.1 NETWORKING

When authorization failures occur with SnapCreator, verify the configuration, firewall permissions, and network address translation (NAT).

IP/HOSTNAME

Unless you use `host equiv` the storage system name returned from the `hostname` command on the controller should be the same as the one entered in the SnapCreator config file.

Do not use an FQDN when the hostname of a storage system is abbreviated.

Make sure that the IP resolution matches the name you specified. If there is a mismatch, correct it by using `host equiv` on the storage system.

To enable `host equiv` do as follows:

```
options httpd.admin.hostsequiv.enable on
```

Edit the `/etc/hostsequiv` file and add the following:

```
<IP/Name of host in SnapCreator config> <SnapCreator user>
```

FIREWALL

If there is a firewall between the host running SnapCreator and your NetApp storage system, make sure you have ACLs (bidirectional) open for 80, 443, or both. Make sure the return traffic from the storage system is allowed to go to the system running SnapCreator on at least non-privileged ports.

NAT

If you use NAT, make sure the source/destination IP addresses are not changed in the TCP packet. The host and storage system need to know who they are talking to, presenting a firewall IP instead of the actual host or controller IP may cause problems.

HTTPS

To use HTTPS (443) for Linux or AIX, install the `openssl-devel` RPM. This RPM contains the `openssl` libraries, which are required to use SSL.

14.2 UPGRADING FROM SNAPCREATOR V2.6

For SnapCreator 3.0 to use a configuration file from version 2.6, add the following parameters:

```
TRANSPORT=HTTP
PORT=80
NTAP_NUM_VOL_CLONES=1
NTAP_CONSISTENCY_GROUP_TIMEOUT=medium
```

14.3 UPGRADING FROM SNAPCREATOR V3.0

For SnapCreator 3.1 to use a configuration file from version 3.0, add the following parameters:

```
NTAP_SNAPSHOT_DISABLE=N
NTAP_OSSV_ENABLE=N
NTAP_OSSV_HOMEDIR=/usr/snapvault
POST_NTAP_DATA_TRANSFER_CMD1=
NTAP_ASUP_ERROR_ENABLE=N
APP_NAME=oracle|db2|maxdb|mysql|notes|sybase
```

ORACLE

Replace all parameters with:

```
DATABASES=db1:user1;db2:user2
SQLPLUS_CMD="/path/to/sqlplus"
CNTRL_FILE_BACKUP_DIR=/path/to/cntrl_files
ORA_TEMP=/path/to/oratemp
OS_WINDOWS=N
```

LOTUS DOMINO

Add the following parameters:

```
DOMINO_SERVICE_NAME="Lotus Domino Server (FDominodata)"
```

14.4 UPGRADING FROM SNAPCREATOR V3.1

For SnapCreator 3.2 to use the configuration file from version 3.1, do the following:

1. Add the following parameters:

```
APP_QUIESCE_CMD<##>  
APP_UNQUIESCE_CMD<##>  
PRE_APP_QUIESCE_CMD<##>  
PRE_APP_UNQUIESCE_CMD<##>  
POST_APP_QUIESCE_CMD<##>  
POST_APP_UNQUIESCE_CMD<##>
```

2. Delete the following parameters:

```
APP_BACKUP_START_CMD<##>  
APP_BACKUP_STOP_CMD<##>  
PRE_APP_BACKUP_START_CMD<##>  
PRE_APP_BACKUP_STOP_CMD<##>  
POST_APP_BACKUP_START_CMD<##>  
POST_APP_BACKUP_STOP_CMD<##>
```

MAXDB

1. Add the following parameters:

```
MAXDB_DATABASES
```

2. Remove the following parameters:

```
OS_WINDOWS
```

ORACLE

1. Add the following parameters:

```
ORACLE_DATABASES  
ARCHIVE_LOG_ONLY
```

2. Remove the following parameters:

```
DATABASES  
OS_WINDOWS
```

MYSQL

1. Add the following parameters:

```
MYSQL_DATABASES
```

2. Remove the following parameters:

```
DATABASES
```

DB2

1. Add the following parameters:

```
DB2_DATABASES
```

2. Remove the following parameters:

```
DATABASES
```

SYBASE

- 1 Add the following parameters:
SYBASE_DATABASES
- 2 Remove the following parameters:
DATABASES

15 FURTHER QUESTIONS/HELP

If you have questions, comments, or suggestions, contact the SnapCreator community:

http://communities.netapp.com/community/products_and_solutions/databases_and_enterprise_apps/snapcreator

NetApp provides no representations or warranties regarding the accuracy, reliability or serviceability of any information or recommendations provided in this publication, or with respect to any results that may be obtained by the use of the information or observance of any recommendations provided herein. The information in this document is distributed AS IS, and the use of this information or the implementation of any recommendations or techniques herein is a customer's responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. This document and the information contained herein may be used solely in connection with the NetApp products discussed in this document.



www.netapp.com

© Copyright 2010 NetApp, Inc. All rights reserved. No portions of this document may be reproduced without prior written consent of NetApp, Inc. Specifications are subject to change without notice. NetApp, the NetApp logo, Go further, faster, AutoSupport, Data ONTAP, FlexClone, SnapDrive, SnapManager, SnapMirror, Snapshot, SnapVault, and vFiler are trademarks or registered trademarks of NetApp, Inc. in the United States and/or other countries. Oracle is a registered trademark and Oracle10g is a trademark of Oracle Corporation. NetBackup is a trademark of Symantec Corporation. UNIX is a registered trademark of The Open Group. Windows and SQL Server are registered trademarks of Microsoft Corporation. Solaris and Java are trademarks of Sun Microsystems, Inc. Linux is a registered trademark of Linus Torvalds. SAP is a registered trademark of SAP AG. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such. TR-3841