Technical Report

NetApp Snap Creator Framework—Advanced Deployment Scenarios

Greg Loughmiller, Matt Robinson, Keith Tenzer, Sebastian Hegmanns, Dr. Adolf Hohl, NetApp
June 2011 | TR-3933

TABLE OF CONTENTS

LIST OF FIGURES

1 ABSTRACT

This document outlines data protection use cases for which NetApp® Snap Creator Framework can provide a solution. We refer to these use cases as advanced use cases because they extend the applicability of Snap Creator beyond its broad support of applications and databases. This document highlights how to solve data protection issues using Snap Creator in the following scenarios:

- Backup of advanced virtual machine environments
- Data protection in highly consolidated hosting environments
- Data protection for secure hosting scenarios
- End-to-end storage provisioning and protection

2 INTRODUCTION

Data protection is key for productive and mission-critical applications. As data centers change and move toward service factories, their operators have to manage new challenges for protecting data of heterogeneous applications at reasonable cost. Leveraging shared infrastructures achieves cost efficiency. There is no question that being able to share resources at individual layers of the IT stack enables flexibility and efficient operation.

What is applicable for service delivery is applicable for data protection. To meet different customer and application requirements with regard to data protection you need to be able to leverage the shared storage infrastructure for efficient and flexible operation.

This document shows how to implement data protection for advanced applications and infrastructure deployments to leverage NetApp shared storage throughout the data lifecycle. We show how to use the Snap Creator Framework to implement these deployments.

2.1 ADVANCED USE CASES VERSUS SUPPORTED APPLICATIONS

Snap Creator Framework supports a variety of data protection applications. It does this by using a plug-in framework that pays attention to application specifics, for example, needing to bring the application into a consistent/recoverable state. For applications that don't have a plug-in available, one can easily be created. The Snap Creator Framework takes care of performing all necessary actions to perform backup and restore as a basic data protection use case.

This document addresses advanced Snap Creator use cases in which, for example, multiple-application consistency has to be coordinated for data protection purposes.

Figure 1) Snap Creator allows implementing deployment scenarios with multiple applications.
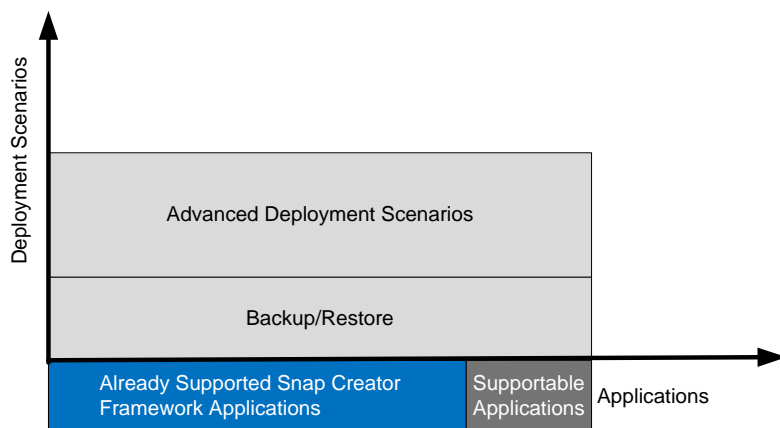
2.2 STRUCTURE OF THIS DOCUMENT

This document is structured with a focus on use cases and problems that arise in operating complex IT and software landscapes. Each use case is addressed in its own chapter. This document answers the following questions:

- Do you run virtual machine landscapes and need application consistency within the virtual machines?
- Do you host like environments with high consolidation and need data protection?
- Do you need to orchestrate and automate your software deployments including data protection?

3 BACKUP OF ADVANCED VIRTUAL MACHINE ENVIRONMENTS

This use case is about an application-consistent backup of applications running in virtual machines. The architecture is outlined based on an Oracle® Database application running inside a VMware® virtual machine. It describes how to master the challenges of creating consistent backups from a database, operating system, and storage perspective. It leverages the fact that the Snap Creator Framework can enforce consistency among more than just one layer to perform application-consistent backups.

3.1 ARCHITECTURE

To address the requirements of the use case, a consistent state at several layers has to be coordinated. Technically this is addressed through the layering of consistency points throughout the backup process to make sure that Oracle, VMware, and NetApp are in appropriate backup states as each snapshot is taken. Using the NetApp Snap Creator Framework, backup configurations are created for both the Oracle and VM layer so that the database and VM are quiesced before creating a NetApp Snapshot™ copy.

The process for taking a backup coordinates the application inside the VM, the VM, and the storage controller. It is as follows:

1. The Oracle Database inside the VM is placed in hot backup mode.
2. VMware creates a snapshot of the VM so that the state is consistent.
3. A NetApp Snapshot copy of the volume in which the VM and all of its database contents are stored is created.
4. The VMware snapshot is removed.
5. The Oracle Database is taken out of hot backup mode.

Once steps 1 and 2 are completed, a Snapshot copy can be created and all the database content can be recovered to the point at which the database went into hot backup mode. In addition, the VMware Snapshot copy will allow any new logs contained within the VM between steps 1 and 2 to be played into the database when it is recovered. Figure 2 illustrates the setup of the components. The outlined sequence of actions is controlled by the Snap Creator server (SC server). Several Snap Creator components are involved:

- A Snap Creator agent with the Oracle plug-in takes care of putting the Oracle Database into hot backup mode.
- The Snap Creator agent with the VIBE plug-in takes care of creating a Snapshot copy of the virtual machine.
- The Snap Creator server triggers the creation of a Snapshot copy at the NetApp storage controller.

Figure 2) Architecture of the Snap Creator components in this use case.
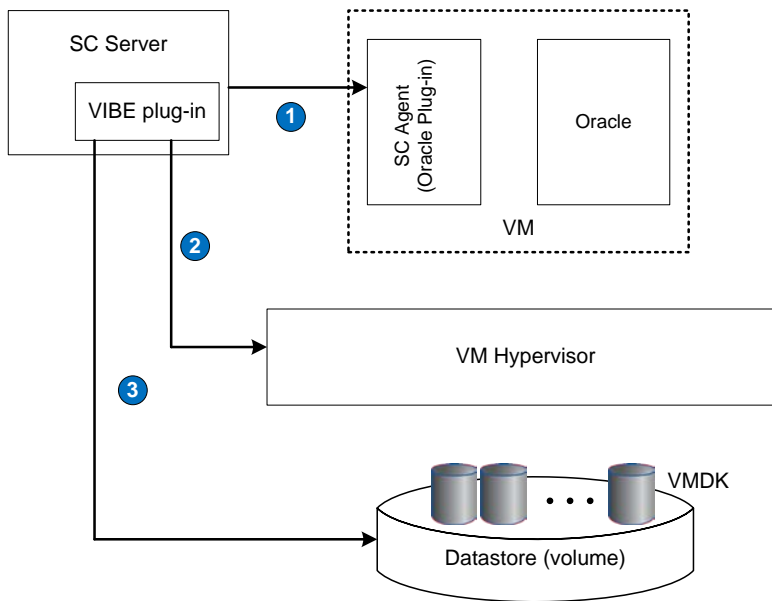
3.2   CONFIGURATION DETAILS

This section illustrates the interplay of the components.

SC CONFIGURATION FOR ORACLE IN VM

The purpose of the configuration for the Oracle Database running in the virtual machine environment is to put the database into hot backup mode while creating the Snapshot copy. This task is delegated to the Snap Creator agent residing right next to the database. The configuration resides centrally on the Snap Creator server and the delegation is triggered through an Internet connection. The config file is named oracle_in_vm.conf.

```
SNAME=agent
VOLUMES=172.17.1.110:vol_Datastore1
NTAP_USERS=172.17.1.110:root/passwd
APP_NAME=ORACLE              ## use the Oracle plugin
APP_DEFINED_RESTORE=Y

SC_AGENT=myoraclevmhost:9090            ## delegates the execution to this agent
SC_AGENT_TIMEOUT=60             ## set a robust timeout

## Oracle configuration parameters go here
ORACLE_DATABASES=mydatabase:dbadmin
SQLPLUS_CMD=c:\oraclexe\app\oracle\product\10.2.0\server\BIN\sqlplus.exe
CNTL_FILE_BACKUP_DIR=c:\oraclexe\backup
ORA_TEMP=c:\temp
ARCHIVE_LOG_ONLY=N
ORACLE_HOME=c:\oraclexe\app\oracle\product\10.2.0\server
```

In this example, `myoraclevmhost` is the hostname of the VM that is running Oracle.

SNAP CREATOR CONFIGURATION FOR VMWARE

The purpose of the Snap Creator configuration for VMware is to create a VMware Snapshot copy of the virtual machine. To do this, the VIBE plug-in needs credentials to perform the required actions via the VMware APIs.

Change the following parameters in the `configs\oracle_vm\vm.conf` file:

```
SNAME=agent
VOLUMES=172.17.1.110:vol_Datastore1
NTAP_USERS=172.17.1.110:root/passwd
APP_NAME=VIBE
VCENTER_USER=administrator
VCENTER_PASSWD=netapp1
VIBE_VSPHERE_NAMES=172.17.44.11::rhel54bcvm2
```

The `VCENTER_USER` and `VCENTER_PASSWD` parameters identify the vCenter™ authentication that allows the VIBE plug-in to perform VMware API calls. The vCenter IP address and Oracle VM being backed up are specified in the `VIBE_VSPHERE_NAMES` parameter. Other parameters can be changed in the configuration file to suit the user's environment.

MASTER SNAP CREATOR CONFIGURATION

Once the VM and Oracle configuration files are created, a new master configuration file is created. The new configuration file will call both the VM and Oracle Snap Creator configuration files in the appropriate order:

```
SNAME=OracleBackup
VOLUMES=172.17.1.110:vol_Datastore1
NTAP_SNAPSHOT_RETENTIONS=daily:7
NTAP_USERS=172.17.1.110:root/passwd
APP_QUIESCE_CMD01=C:\Snap Creator\bin\Snap Creator.exe --profile oracle_vm --config
oracle --action quiesce --verbose --debug
APP_QUIESCE_CMD02=C:\Snap Creator\bin\Snap Creator.exe --profile oracle_vm --config vm
--action quiesce --verbose --debug
APP_UNQUIESCE_CMD01=C:\Snap Creator\bin\Snap Creator.exe --profile oracle_vm --config
oracle --action unquiesce --verbose --debug
APP_UNQUIESCE_CMD02=C:\Snap Creator\bin\Snap Creator.exe --profile oracle_vm --config
vm --action unquiesce --verbose --debug
PRE_EXIT_CMD01=C:\Snap Creator\bin\Snap Creator.exe --profile oracle_vm --config
oracle --action unquiesce --verbose --debug
PRE_EXIT_CMD02=C:\Snap Creator\bin\Snap Creator.exe --profile oracle_vm --config vm --
action unquiesce --verbose --debug
```

The first four parameter changes are important, because the real snapshot names, schedules, and volumes are defined and used by Snap Creator. `NTAP_SNAPSHOT_RETENTIONS` specifies the number of Snapshot copies to retain when performing a daily backup. `NTAP_USERS` and `VOLUMES` are defined as the same as in the `vm.conf` and `oracle.conf` configuration files and point to the authentication user and password as well as the volume in which the Oracle VM is contained.

The `APP_QUIESCE_CMD`, `APP_UNQUIESCE_CMD` and `PRE_EXIT` parameters outline the order in which the Oracle and VM configuration files are called. For all three operations, the Oracle configuration file is called first, followed by the VM configuration file. This enables the database to stay in hot backup mode for a minimum amount of time.

3.3   CONCLUSION

This architecture exploits the Snap Creator Framework functionality of coordinating consistency points of different application layers to implement application-consistent data protection in virtual environments. The functionality was discussed using VMware and Oracle Databases. However the applicability is not limited to these components.

# 4   DATA PROTECTION IN HIGHLY CONSOLIDATED HOSTING ENVIRONMENTS

This use case is about a hosting provider who is providing a large amount of small to medium-sized database instances as a service. His customers are either internal or external ones. The major goal of the provider is to offer this service at a competitive price. Among other approaches he achieves this by optimizing the operation of the databases for very high consolidation.

The hosting provider does not sacrifice data protection service levels and uses Snap Creator to take care of data protection and backup.

4.1   ARCHITECTURE

This section highlights the architecture decisions in implementing this solution. Here, the ability of Snap Creator to coordinate database consistency points with NetApp Snapshot copies is exploited. This forms the basis for implementing a storage layout that allows achieving much higher consolidation.

Figure 3) Multiple database instances share set of volumes.
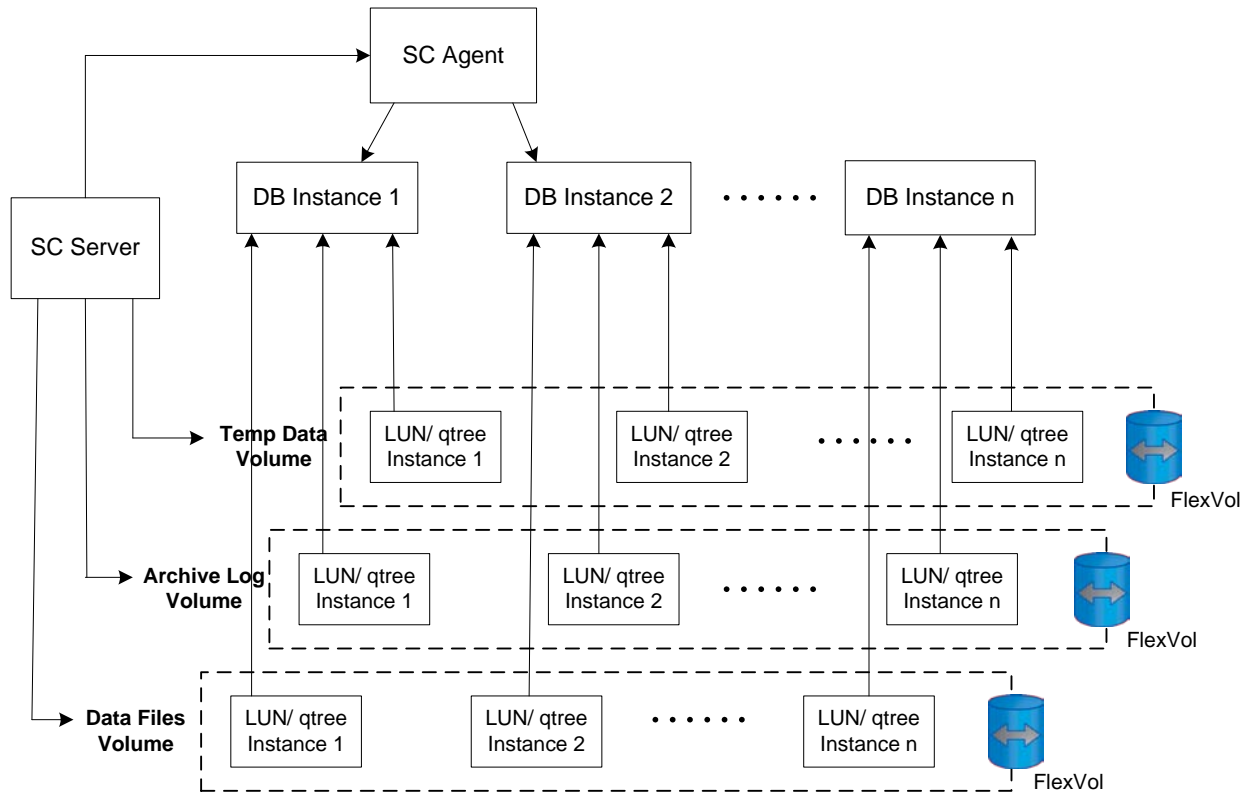
STORAGE LAYOUT

The storage layout plays a vital role in implementing this deployment scenario. It is optimized for achieving a high level of consolidation and an efficient use of available resources. The number of volumes (500) and Snapshot copies per volume (256) are two limitations that exist per storage controller. Implementing storage layouts according to the existing best-practice guides usually requires three or more separate volumes per database. This artificially limits the level of consolidating database instances to a single controller.

Here, a storage layout is used that is characterized by sharing volumes among multiple database instances. This allows us to shift the volume-per-instance trade-off and the Snapshot copy-per-instance trade-off. Since the database instances share volumes, they share the number of available Snapshot copies for creating recovery points. Providing similar data protection service levels and grouping the consistency points with the Snapshot copy preserve the highest flexibility in defining the data protection service-level agreement. Providing independent data protection for the instances comes at the price of a reduced number of individual recovery points per instance, assuming that 25 DB instances share Snapshot copies of 1 volume for an hourly backup occurring every 2 hours. Using Snapshot sharing in this example creates 12 Snapshot copies per day, allowing retention for 21 days (12 * 21 = 252). A per-instance Snapshot copy (not shared) would create 300 Snapshot copies per day and allow retention for only 1 day ((25 * 12) * 1 = 300.

When creating a shared volume layout for multiple database instances, similar data types are grouped into the same volumes. For databases, usually there are three types of data: data files, archive logs, and temp or transient data. NetApp recommends creating 3 volumes to bundle up to 25 database instances. Volumes should be created to separate the following data types: data files, archive logs, and temp data. Qtrees are created in each volume with the database name or SID to logically organize data within the shared volumes. Going with a 25:1 database consolidation ratio and using the shared volume layout creates an environment that can scale much higher and achieve greater efficiency compared to a classical storage layout with three volumes per database instance. Instead of 166 DBs per storage controller using a nonshared typical 3:1 volume layout we now have 4,150 DBs per storage controller (25 * (500 / 3)). This allows us to get the best efficiency and utilization from a storage perspective.

Note: Databases might also be consolidated at the host layer to have two consolidation efforts occur: one on the storage side and the other on the host side. The goal should be to consolidate as many databases as possible per host. NetApp recommends no more than 25 databases per host.

STORAGE EFFICIENCY CONSIDERATIONS

Since the layout is characterized by the grouping of similar types of data, it is possible to achieve significant storage efficiency returns by running NetApp deduplication. If you refer to RA-0007, you will see that this layout fulfills the preconditions for a dedupe-centric storage layout. Furthermore, instant storage efficiency returns can be achieved by provisioning new instances with a volume or file FlexClone® volume. Ideally the volumes are created as thin-provisioned volumes with autogrow = on and autodelete = off.

4.2   ARCHITECTURE DETAILS

In the shared volume layout it is important to be able to back up multiple databases at the same time on the shared host. Using Snap Creator we can achieve that for the same database type. In our example, 25 instances of the same database type are consolidated by utilizing just 3 volumes on the storage controller. All 25 databases are backed up at the same time, meaning the consistency point of the DB instances and NetApp Snapshot are coordinated and only 1 Snapshot copy is created. This gives great flexibility in creating a dense set of recovery points.

For database backups, NetApp recommends backing up the data files and archive logs daily. Snap Creator is configuration file driven; below are configuration best practices using Snap Creator for multiple-database backup. The configuration file on the server names all use volumes of the database instances on the server.

```
SNAME=ConsolidatedORABackup
VOLUMES=StorageControllerNAME:dblogvolumes,dbdatavolume1,dbdatavolume2
NTAP_SNAPSHOT_RETENTIONS=daily:7
NTAP_USERS=StorageControllerNAME:root/passwd


APP_NAME=oracle
SC_AGENT=HOSTNAME:port     ##(=dbserver)
SC_AGENT_TIMEOUT=300                        ## set a robust timeout
```

The application-specific plug-in (oracle.conf) lists the databases on this server that persist their data into the specified volumes. All database instances are referenced by their name and access credential.

```
## Oracle configuration parameters go here
ORACLE_DATABASES=mydatabase1:dbadmin;mydatabase2:dbadmin;… mydatabase25:dbadmin

#ORACLE specifics
SQLPLUS_CMD=c:\oraclexe\app\oracle\product\10.2.0\server\BIN\sqlplus.exe
CNTL_FILE_BACKUP_DIR=c:\oraclexe\backup
ORA_TEMP=c:\temp
ARCHIVE_LOG_ONLY=N
ORACLE_HOME=c:\oraclexe\app\oracle\product\10.2.0\server
```

4.3   CONCLUSION

This method for database consolidation enables greater efficiency and better database density on the storage controller. This setup, however, is not the most optimal layout from a performance perspective because shared volumes can create bottlenecks. NetApp recommends using this setup for smaller databases that don't have the highest performance requirements. Both a shared and a traditional nonshared volume approach can also work well together. At the end of the day, a healthy mix will enable the best performance for databases that need it and the best efficiency for databases that are less critical from a performance perspective.

5 DATA PROTECTION FOR SECURE HOSTING SCENARIOS

For service providers offering hosted IT applications, simplification of processes and offerings is a key to achieve operational excellence. This can be in conflict with the flexibility a potential customer expects. This use case allows a service provider's customer to select from a broader set of data protection options in a secure hosting environment. Thus customers can personalize their settings to best suit their business needs.

In this use cases, the service provider provides Microsoft® SQL®-based applications that are protected using NetApp SnapManager® for Microsoft SQL Server. The Snap Creator Framework extends the data protection provided by Microsoft SQL Server. While Snap Creator implements a single interface for the scheduling, an application-consistent backup is created by SnapManager.

5.1 ARCHITECTURE

The architecture is characterized by the ability to allocate and assign compute and storage resources in an isolated network on a per-customer basis. This virtual data center is completely isolated using VLAN fencing, and customers cannot see or learn anything about other customers. Management of the data protection is completely within the customer's security domain. There is no need to manage it from a separate management station requiring another communication path.

This architecture implements multiple security contexts for the service provider's customers and a trusted context for the service provider itself. Customers have a trust relationship with the service provider but they do not need to trust any other of the service provider's customers. Thus a customer cannot leave his or her security context. Figure 4 shows the security context and the data protection relationship.

Figure 4) Architecture of service platform.
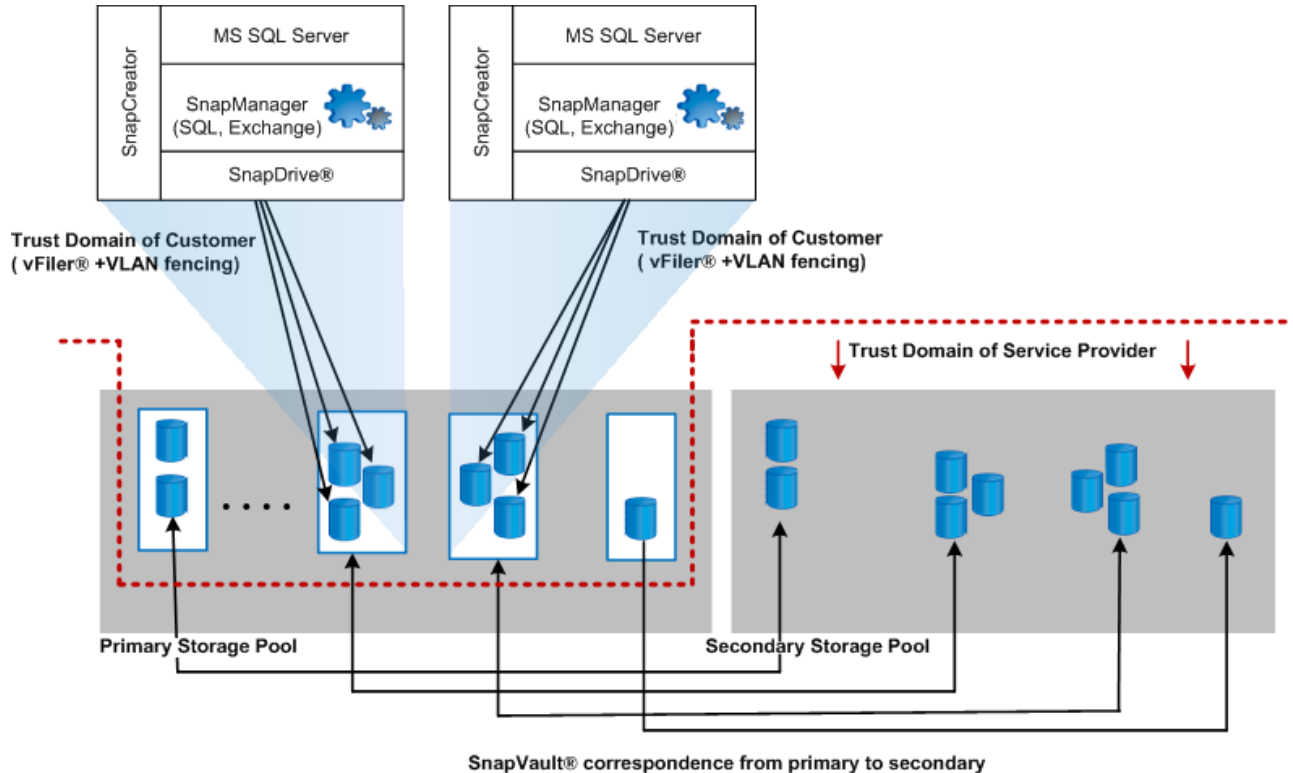
To summarize, a customer environment is implemented as follows:

- The security context on the storage layer is implemented by means of a vFiler unit in the primary storage controller that corresponds to the customer.
- The security context on the network layer is implemented by means of VLAN fencing. Each customer communicates within his/her own VLAN.
- The secondary storage controllers are in the trusted domain of the service provider. Data is transferred between primary and secondary storage controllers by means of a separate VLAN. Any content on this network cannot be accessed from customer networks.
- Secondary storage controllers maintain data protection relationships between volumes of the primary vFiler units and corresponding volumes on the secondary storage controllers. This allows scaling primary and secondary controllers independently, leading to lower incremental costs. There is no corresponding vFiler unit on the secondary controllers for vFiler units on the primary controllers.

The security context is set up by the service provider. In detail, the service provider is responsible for creating a customer vFiler unit and the SnapVault relationships for its object between the primary and secondary storage controller. In the following, only the storage-specific parts are covered:

- Creation of a customer vFiler unit on the primary storage controller. This vFiler unit is attached to the customer VLAN.
- Creation of volumes in the customer vFiler unit.
- Creation of SnapVault relationships between volumes on the primary storage controller and the secondary storage controllers.

5.2 ARCHITECTURE DETAILS

For service providers offering hosted IT applications it is important to be able to back up all customer applications using a single method. Using Snap Creator we can achieve that for almost all types of applications by leveraging the application plug-in framework. Supported applications can be consistently backed up with Snap Creator. In addition, the Snapshot copy is created and transports can be triggered.

In this specific use case, we want to outline the interplay between SnapManager for Microsoft SQL Server and Snap Creator (see Figure 5).

Figure 5) Interplay of Snap Creator with SnapManager.
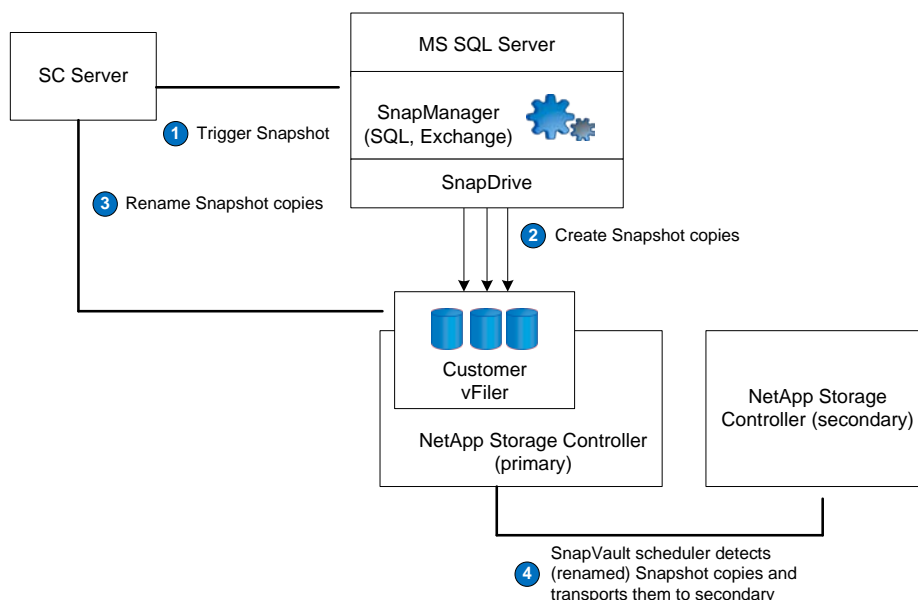
The Snap Creator server is executed on the same host as the Microsoft SQL Server and its SnapManager and SnapDrive software. Snap Creator triggers SnapManager to get the application in a consistent state and create a Snapshot copy of the data files (step 1). Once the Snapshot copies are created (step 2), Snap Creator renames them (to sv_nightly.0 and sv_snapinfo.0) in the customer's vFiler unit so that these Snapshot copies are considered by the SnapVault scheduler, which then transports them to the secondary controller.

Covering the complete configuration would be beyound the scope of this document. For more information, refer to the [Snap Creator Installation and Administration Guide](#).

5.3 CONCLUSION

This architecture extends the data protection options of the SnapManager product line to address different customer requirements. Using Snap Creator, data protection tasks can be automated to implement secure hosting environments, taking full advantage of the shared storage architecture.

6 END-TO-END STORAGE PROVISIONING AND PROTECTION

This use case focuses on data protection in an automated end-to-end application-provisioning scenario. It is relevant for service providers deploying applications on demand in a fully automated manner without needing manual intervention.

We assume the following items exist:

1. A storage provisioning engine provisioning the storage layout and its details for a complete application; for example, multiple volumes for a database-centric application with classification of log, data, and temporary data volumes
2. A definition of the data protection service level; within the scope of this document, as an example, we explain a simplified policy
3. The host is either virtual or physical, and attaches to the provisioned storage and serves the application; the host has a Snap Creator agent running
4. A Snap Creator server that controls and monitors its agents

A Snap Creator config file can be built to exactly fit this environment and perform the desired data protection for the provisioned instance.

Note that within this document we don't focus on how the software gets in place. It might be present because it was part of a golden postcustomized image or brought in place by software deployment tools used in orchestration/automation frameworks.

Within the scope of this document we highlight the config file creation of a specific application scenario and data protection service level. This can be easily automated via text processing tools.

6.1 ARCHITECTURE

The architecture consists of an orchestration engine that is able to talk to the individual layers of the IT stack. The provisioning of a new service instance consists of the following steps:

1. Provision the storage.
2. Provision the host.
3. Boot the host and provision the applications, including the Snap Creator agent.
4. Attach or mount the provisioned storage.
5. Create the configuration file for the specific host, application, data protection service level, and storage.
6. Deliver the configuration file to Snap Creator server, which centrally monitors and triggers data protection tasks.
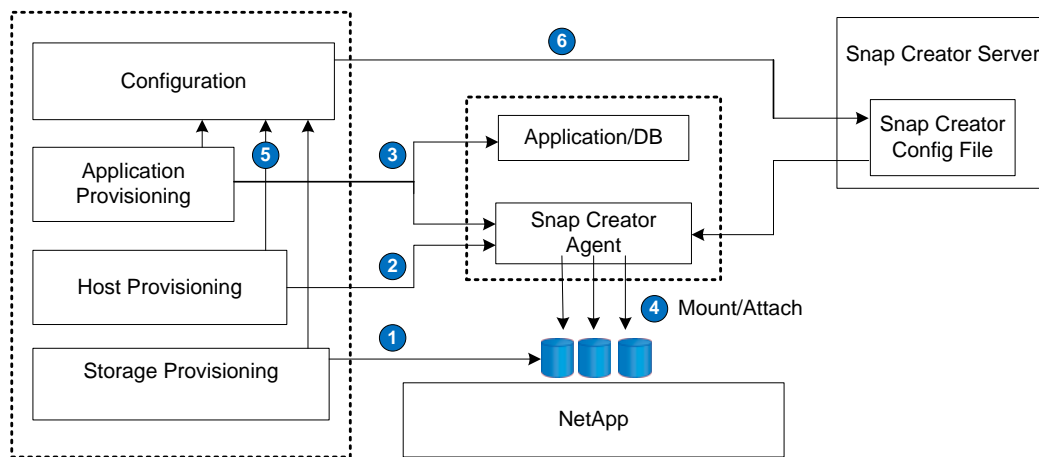
Using the given inputs, a config file that describes the environment has to be created. The basic steps are outlined below and are meant to be a starting point for further customizations (refer to the [Snap Creator Installation and Administration Guide](#)).

The configuration of a Snap Creator config file is performed in two steps. Step one focuses on the basic configuration, including details to create Snapshot copies on the storage controller. Step two focuses on configuring the desired data protection level.

6.2 STEP 1—BASIC SNAP CREATOR CONFIGURATION

This part of the configuration file sets up the basic information required for the target database.

```
#######################
### Required Options ###
#######################
```
(#1) `SNAME=fcpsi1`
(#2) `SNAP_TIMESTAMP_ONLY=N`
(#3) `VOLUMES=ben-5:fcpsi1_oradata,fcpsi1_oractl`
(#4) `NTAP_USERS=ben-5:snapc/53616c7465645f5f202b41b5f56560fac89642d24881b9200205b3cbea7e6251`
(#5) `NTAP_PWD_PROTECTION=Y`
(#6) `TRANSPORT=HTTP`
(#7) `PORT=80`
(#8) `NTAP_SNAPSHOT_RETENTIONS=daily:5`
`LOG_NUM=10`
`SC_TMP_DIR=`

The details of the parameters are as follows:

1. `SNAME`

Recommended Setting	fcpsi1
Description	Your Snapshot copy naming convention should be unique. Snapshot copies on NetApp are deleted according to the naming convention and retention policy used.
Notes	This will be part of the core name of the Snapshot copy when executing `snaplist`.

2. `SNAP_TIMESTAMP_ONLY`

Recommended Setting	N
Description	If set to Y, Snapshot copies will end with YYYYMMDDHHMMSS. Otherwise, new Snapshot copies will be suffixed with _**recent** and renamed to end with YYYYMMDDHHMMSS. If set to N, the Snapshot copies will have the _**recent** suffix to accommodate cloning in the future. Local and remote cloning requires this.
Notes	

3. `VOLUMES`

Recommended Setting	
Description	List of source appliances and volumes for which you want to create a Snapshot copy.
Notes	Enter the volumes that you want to back up.

4. `NTAP_USERS`

Recommended Setting	
Description	List of appliances and their corresponding user names/passwords.
Notes	The users and encrypted passwords are placed here for each storage system. Appliances are separated by ";". Users are separated by ",".

5. `NTAP_PWD_PROTECTION`

Recommended Setting	
Description	Enables password protection. When this option is set to "Y" it encrypts all passwords and saves the encrypted passwords in the configuration file.
Notes	If the password is encrypted, then this flag must be set. This is also applicable for all password fields in the configuration file. This option encrypts all passwords.

6. `TRANSPORT`

Recommended Setting	HTTP/HTTPS
Description	Protocol used between the storage system and the SC server process.
Notes	This example uses HTTP.

7. `PORT`

Recommended Setting	Port 80 or Port 443. This example uses Port 80.
Description	Port between the SC server and the storage system.
Notes	

8. `NTAP_SNAPSHOT_RETENTIONS`

Recommended Setting	
Description	The number of Snapshot copies that you want to retain for a given policy. For example, to keep seven daily, four weekly, and one monthly Snapshot copy, specify it as follows: `daily:7,weekly:4,monthly:1.`
Notes	

6.3 STEP 2—DATA PROTECTION AND LOCAL SNAPSHOT CONFIGURATION

To set data protection, configure the following parameters:

```
#######################
### Snapshot Options ###
#######################
(#1)   NTAP_SNAPSHOT_RETENTION_AGE=4
       SNAPDRIVE=N
       SNAPDRIVE_DISCOVERY=N
(#2)   NTAP_SNAPSHOT_DISABLE=N
(#3)   NTAP_SNAPSHOT_NODELETE=N
       NTAP_SNAPSHOT_CREATE_CMD01=
(#4)   NTAP_CONSISTENCY_GROUP_SNAPSHOT=Y
(#5)   NTAP_CONSISTENCY_GROUP_TIMEOUT=medium
(#6)   NTAP_SNAPSHOT_DELETE_BY_AGE_ONLY=N
       NTAP_SNAPSHOT_DEPENDENCY_IGNORE=Y
```

The details of the parameters are as follows:

1. `NTAP_SNAPSHOT_RETENTION_AGE`

Recommended Setting	4 NetApp recommends a setting of four days for most scenarios.
Description	Defines the retention age (in days) after which the Snapshot copies will be deleted. Snapshot copies are deleted if: (1) the number of Snapshot copies exceeds the one specified in the Snapshot retentions policy `NTAP_SNAPSHOT_RETENTIONS` and (2) the Snapshot is older than the specified retention age.
Notes	

2. `NTAP_SNAPSHOT_DISABLE`

Recommended Setting	N We want Snap Creator to create a Snapshot copy.
Description	Disables Snap Creator from creating a Snapshot copy. The idea of this option is that Snap Creator can handle SnapVault® or SnapMirror® without creating a new Snapshot copy.
Notes	This is useful when you want to run other features and NOT create a Snapshot copy.

3. `NTAP_SNAPSHOT_NODELETE`

Recommended Setting	N
Description	It this option is enabled, no Snapshot copies will be deleted, which will cause your NetApp volume to fill up. This setting overrides `NTAP_SNAPSHOT_RETENTIONS`.
Notes	

4. `NTAP_CONSISTENCY_GROUP_SNAPSHOT`

Recommended Setting	Y
Description	Enables the use of consistency groups. By enabling this setting, you can take a consistent (I/O fencing) Snapshot copy across more than one volume and across multiple storage controllers.
Notes	

5. `NTAP_CONSISTENCY_GROUP_TIMEOUT`

Recommended Setting	In busy environments, set this to "relaxed."
Description	Sets the timeout value for the I/O fencing between volumes. All volumes must be quiesced (I/O fencing) before creating a consistency group Snapshot copy.
Notes	This is the timeout value used for the CG Snapshot function.

6. `NTAP_SNAPSHOT_DELETE_BY_AGE_ONLY`

Recommended Setting	{PRIMARY|SECONDARY|BOTH|N}
Description	This option overrides `NTAP_SNAPSHOT_RETENTIONS` and deletes outdated Snapshot copies.
Notes	

6.4 CONCLUSION

The ability to configure the full feature set of Snap Creator through defined configuration file options makes the solution well suited for environments that are set up in an automated manner and have no need for manual configuration steps. To illustrate the ease of configuration, we detailed the creation of a sample configuration.

7 SUMMARY

This document outlines the applicability of the Snap Creator Framework in scenarios that require coordination of consistency points and automation. The applicability is showcased with selected software components, but is not limited to them. The Snap Creator Framework provides data protection for a broad range of applications in advanced deployment scenarios.

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.

Go further, faster®

NetApp®

www.netapp.com