Automated Oracle Data Protection

NetApp Solutions

NetApp
March 11, 2022

Table of Contents

Solution Overview

Automated Data Protection for Oracle Databases

Organizations are automating their environments to gain efficiencies, accelerate deployments, and reduce manual effort. Configuration management tools like Ansible are being used to streamline enterprise database operations. In this solution, we demonstrate how you can use Ansible to automate the data protection of Oracle with NetApp ONTAP. By enabling storage administrators, systems administrators, and DBAs to consistently and rapidly setup data replication to an offsite data center or to public cloud, you achieve the following benefits:

- Eliminate design complexities and human errors, and implement a repeatable consistent deployment and best practices
- Decrease time for configuration of Intercluster replication, CVO instantiation, and recovery of Oracle databases
- Increase database administrators, systems and storage administrators productivity
- Provides database recovery workflow for ease of testing a DR scenario.

NetApp provides customers with validated Ansible modules and roles to accelerate deployment, configuration, and lifecycle management of your Oracle database environment. This solution provides instruction and Ansible playbook code, to help you:

On Prem to on prem replication

- Create intercluster lifs on source and destination
- Establish cluster and vserver peering
- Create and initialize SnapMirror of Oracle volumes
- Create a replication schedule through AWX/Tower for Oracle binaries, databases, and logs
- Restore Oracle DB on the destination, and bring database online

On Prem to CVO in AWS

- Create AWS connector
- Create CVO instance in AWS
- Add On-Prem cluster to Cloud Manager
- Create intercluster lifs on source
- Establish cluster and vserver peering
- Create and initialize SnapMirror of Oracle volumes
- Create a replication schedule through AWX/Tower for Oracle binaries, databases, and logs
- Restore Oracle DB on the destination, and bring database online

For more details or to begin, please see the overview videos below.

AWX/Tower Deployments

- Part 1: TBD

video

- Part 2: TBD

video

After you are ready, click here for getting started with the solution.

Getting started

This solution has been designed to be run in an AWX/Tower environment.

AWX/Tower

For AWX/Tower environments, you are guided through creating an inventory of your ONTAP cluster management and Oracle server (IPs and hostnames), creating credentials, configuring a project that pulls the Ansible code from NetApp Automation Github, and the Job Template that launches the automation.

1. The solution has been designed to run in a private cloud scenario (on-premise to on-premise), and hybrid cloud (on-premise to public cloud Cloud Volumes ONTAP [CVO])
2. Fill out the variables specific to your environment, and copy and paste them into the Extra Vars fields in your job template.
3. After the extra vars have been added to your job template, you can launch the automation.
4. The automation is set to be ran three phases (Setup, Replication Schedule for Oracle Binaries, Database, Logs, and Replication Schedule just for Logs), and a forth phase to recovering the database at a DR site.
5. For detailed instructions for obtaining the keys and tokens necessary for the CVO Data Protection visit Gather Pre-requisites For CVO and Connector Deployments

Requirements

On-Prem |

Environment	Requirements
Ansible environment	AWX/Tower
	Ansible v.2.10 and higher
	Python 3
	Python libraries - netapp-lib - xmltodict - jmespath
ONTAP	ONTAP version 9.8 +
	Two data aggregates
	NFS vlan and ifgrp created
Oracle server(s)	RHEL 7/8
	Oracle Linux 7/8
	Network interfaces for NFS, public, and optional mgmt
	Existing Oracle environment on source, and the equivalent Linux operating system at the destination (DR Site or Public Cloud)

CVO

Environment	Requirements
Ansible environment	AWX/Tower
	Ansible v.2.10 and higher
	Python 3
	Python libraries - netapp-lib - xmltodict - jmespath
ONTAP	ONTAP version 9.8 +
	Two data aggregates
	NFS vlan and ifgrp created
Oracle server(s)	
	RHEL 7/8
	Oracle Linux 7/8
	Network interfaces for NFS, public, and optional mgmt
	Existing Oracle environment on source, and the equivalent Linux operating system at the destination (DR Site or Public Cloud)
	Set appropriate swap space on the Oracle EC2 instance, by default some EC2 instances are deployed with 0 swap

Environment	Requirements
Cloud Manager/AWS	AWS Access/Secret Key
	NetApp Cloud Manager Account
	NetApp Cloud Manager Refresh Token

Automation Details

On-Prem |

This automated deployment is designed with a single Ansible playbook that consists of three separate roles. The roles are for ONTAP, Linux, and Oracle configurations.
The following table describes which tasks are being automated.

Playbook	Tasks
ontap_setup	Pre-check of the ONTAP environment
	Creation of Intercluster LIFs on source cluster (OPTIONAL)
	Creation of Intercluster LIFs on destination cluster (OPTIONAL)
	Creation of Cluster and SVM Peering
	Creation of destination SnapMirror and Initialization of designated Oracle volumes
ora_replication_cg	Enable backup mode for each database in /etc/oratab
	Snapshot taken of Oracle Binary and Database volumes
	Snapmirror Updated
	Turn off backup mode for each database in /etc/oratab
ora_replication_log	Switch current log for each database in /etc/oratab
	Snapshot taken of Oracle Log volume
	Snapmirror Updated
ora_recovery	Break SnapMirror
	Enable NFS and create junction path for Oracle volumes on the destination
	Configure DR Oracle Host
	Mount and verify Oracle volumes
	Recover and start Oracle database

CVO

This automated deployment is designed with a single Ansible playbook that consists of three separate roles. The roles are for ONTAP, Linux, and Oracle configurations.
The following table describes which tasks are being automated.

Playbook	Tasks
cvo_setup	Pre-check of the environment
	AWS Configure/AWS Access Key ID/Secret Key/Default Region
	Creation of AWS Role
	Creation of NetApp Cloud Manager Connector instance in AWS
	Creation of Cloud Volumes ONTAP (CVO) instance in AWS
	Add On-Prem Source ONTAP Cluster to NetApp Cloud Manager
	Creation of destination SnapMirror and Initialization of designated Oracle volumes
ora_replication_cg	Enable backup mode for each database in /etc/oratab
	Snapshot taken of Oracle Binary and Database volumes
	Snapmirror Updated
	Turn off backup mode for each database in /etc/oratab
ora_replication_log	Switch current log for each database in /etc/oratab
	Snapshot taken of Oracle Log volume
	Snapmirror Updated
ora_recovery	Break SnapMirror
	Enable NFS and create junction path for Oracle volumes on the destination CVO
	Configure DR Oracle Host
	Mount and verify Oracle volumes
	Recover and start Oracle database

Default parameters

To simplify automation, we have preset many required Oracle parameters with default values. It is generally not necessary to change the default parameters for most deployments. A more advanced user can make changes to the default parameters with caution. The default parameters are located in each role folder under defaults directory.

License

You should read license information as stated in the Github repository. By accessing, downloading, installing, or using the content in this repository, you agree the terms of the license laid out here.

Note that there are certain restrictions around producing and/or sharing any derivative works with the content in this repository. Please make sure you read the terms of the License before using the content. If you do not agree to all of the terms, do not access, download, or use the content in this repository.

After you are ready, click here for detailed AWX/Tower procedures.

Step-by-step deployment procedure

AWX/Tower Oracle Data Protection

1. Create the inventory, group, hosts, and credentials for your environment

This section describes the setup of inventory, groups, hosts, and access credentials in AWX/Ansible Tower that prepare the environment for consuming NetApp automated solutions.

1. Configure the inventory.
    a. Navigate to Resources → Inventories → Add, and click Add Inventory.
    b. Provide the name and organization details, and click Save.
    c. On the Inventories page, click the inventory created.
    d. Navigate to the Groups sub-menu and click Add.
    e. Provide the name oracle for your first group and click Save.
    f. Repeat the process for a second group called dr_oracle.
    g. Select the oracle group created, go to the Hosts sub-menu and click Add New Host.
    h. Provide the IP address of the Source Oracle host's management IP, and click Save.
    i. This process must be repeated for the dr_oracle group and add the the DR/Destination Oracle host's management IP/hostname.

Below are instructions for creating the credential types and credentials for either On-Prem with ONTAP, or CVO on AWS.

On-Prem

Unresolved directive in ent-apps-db/db_protection_awx_automation.adoc - include::ent-apps-db/db_protection_onprem_creds.adoc[]

CVO

Unresolved directive in ent-apps-db/db_protection_awx_automation.adoc - include::ent-apps-db/db_protection_cvo_creds.adoc[]

2. Create a project

1. Go to Resources → Projects, and click Add.
    a. Enter the name and organization details.
    b. Select Git in the Source Control Credential Type field.
    c. enter https://github.com/NetApp-Automation/na_oracle19c_data_protection.git as the source control URL.
    d. Click Save.
    e. The project might need to sync occasionally when the source code changes.

3. Configure global variables

Variables defined in this section apply to all Oracle hosts, databases, and the ONTAP cluster.

1. Input your environment-specific parameters in following embedded global variables or vars form.

   The items in blue must be changed to match your environment.

On-Prem

Unresolved directive in ent-apps-db/db_protection_awx_automation.adoc - include::ent-apps-db/db_protection_onprem_vars.adoc[]

CVO

Unresolved directive in ent-apps-db/db_protection_awx_automation.adoc - include::ent-apps-db/db_protection_cvo_vars.adoc[]

4. Automation Playbooks

There are four separate playbooks that need to be ran.

1. Playbook for Setting up your environment, On-Prem or CVO.
2. Playbook for replicating Oracle Binaries and Databases on a schedule
3. Playbook for replicating Oracle Logs on a schedule
4. Playbook for Recovering your database on a destination host

ONTAP/CVO Setup

Unresolved directive in ent-apps-db/db_protection_awx_automation.adoc - include::ent-apps-db/db_protection_ontap_cvo_setup.adoc[]

Replication For Binary and Database Volumes

Unresolved directive in ent-apps-db/db_protection_awx_automation.adoc - include::ent-apps-db/db_protection_db_replication.adoc[]

Replication for Log Volumes

Unresolved directive in ent-apps-db/db_protection_awx_automation.adoc - include::ent-apps-db/db_protection_log_replication.adoc[]

Restore and Recover Database

Unresolved directive in ent-apps-db/db_protection_awx_automation.adoc - include::ent-apps-db/db_protection_restore_recovery.adoc[]

5. Recovering Oracle Database

1. On-premises production Oracle databases data volumes are protected via NetApp SnapMirror replication to either a redundant ONTAP cluster in secondary data center or Cloud Volume ONTAP in public cloud. In a fully configured disaster recovery environment, recovery compute instances in secondary data center or public cloud are standby and ready to recover the production database in the case of a disaster. The

standby compute instances are kept in sync with on-prem instances by running paraellel updates on OS kernel patch or upgrade in a lockstep.

2. In this solution demonstrated, Oracle binary volume is replicated to target and mounted at target instance to bring up Oracle software stack. This approach to recover Oracle has advantage over a fresh installation of Oracle at last minute when a disaster occurred. It guarantees Oracle installation is fully in sync with current on-prem production software installation and patch levels etc. However, this may or may not have additional sofware licensing implication for the replicated Oracle binary volume at recovery site depending on how the software licensing is structured with Oracle. User is recommended to check with its software licensing personnel to assess the potential Oracle licensing requirement before deciding to use the same approach.
3. The standby Oracle host at the destination is configured with the Oracle prerequisite configurations.
4. The SnapMirrors are broken and the volumes are made writable and mounted to the standby Oracle host.
5. The Oracle recovery module performs following tasks to recovery and startup Oracle at recovery site after all DB volumes are mounted at standby compute instance.
    a. Sync the control file: We deployed duplicate Oracle control files on different database volume to protect critical database control file. One is on the data volume and another is on log volume. Since data and log volumes are replicated at different frequency, they will be out of sync at the time of recovery.
    b. Relink Oracle binary: Since the Oracle binary is relocated to a new host, it needs a relink.
    c. Recover Oracle database: The recovery mechanism retrieves last System Change Number in last available archived log in Oracle log volume from control file and recovers Oracle database to recoup all business transactions that was able to be replicated to DR site at the time of failure. The database is then started up in a new incarnation to carry on user connections and business transaction at recovery site.

Before running the Recovering playbook make sure you have the following:
Make sure it copy over the /etc/oratab and /etc/oraInst.loc from the source Oracle host to the destination host