Automatisierte Oracle Datensicherung

NetApp Solutions

NetApp
October 31, 2024

Inhalt

Automatisierte Oracle Datensicherung

Lösungsüberblick

Auf dieser Seite wird die automatisierte Methode zur Implementierung von Oracle19c auf NetApp ONTAP Storage beschrieben.

Automatisierte Datensicherung für Oracle Datenbanken

Unternehmen automatisieren ihre Umgebungen, um die Effizienz zu steigern, Implementierungen zu beschleunigen und manuelle Aufgaben zu reduzieren. Konfigurationsmanagement-Tools wie Ansible werden zur Optimierung der Abläufe in Unternehmensdatenbanken eingesetzt. Diese Lösung zeigt, wie Sie mit Ansible die Datensicherung von Oracle mit NetApp ONTAP automatisieren können. Storage-Administratoren, Systemadministratoren und DBAs können die Datenreplizierung konsistent und schnell auf ein externes Datacenter oder in die Public Cloud einrichten. Sie profitieren von folgenden Vorteilen:

- Vermeiden Sie Designkomplexität und menschliche Fehler und implementieren Sie eine wiederholbare, konsistente Implementierung und Best Practices
- Verkürzung der Zeit für die Konfiguration von Intercluster-Replizierung, CVO Instanziation und Recovery von Oracle-Datenbanken
- Erhöhen Sie die Produktivität von Datenbank-Administratoren, -Systemen und -Storage-Administratoren
- Bietet Datenbank-Recovery-Workflow zum einfachen Testen eines DR-Szenarios.

NetApp bietet Kunden validierte Ansible-Module und -Rollen, um die Implementierung, Konfiguration und das Lifecycle-Management Ihrer Oracle-Datenbankumgebung zu beschleunigen. Diese Lösung bietet Anweisungen und Ansible-Playbook-Code, um Sie bei folgenden Aufgaben zu unterstützen:

Replizierung vor Ort und in On-Premises-Umgebungen

- Erstellen von Intercluster-Libs an Quelle und Ziel
- Cluster- und vServer-Peering einrichten
- SnapMirror von Oracle Volumes erstellen und initialisieren
- Erstellen Sie einen Replikationszeitplan über AWX/Tower für Oracle-Binärdateien, -Datenbanken und -Protokolle
- Stellen Sie Oracle DB auf dem Ziel wieder her und bringen Sie die Datenbank in den Online-Modus

Von On-Premises zu CVO in AWS

- AWS Connector erstellen
- CVO-Instanz in AWS erstellen
- Hinzufügen eines On-Premises-Clusters zu Cloud Manager
- Erstellen von Intercluster-Libs auf der Quelle
- Cluster- und vServer-Peering einrichten
- SnapMirror von Oracle Volumes erstellen und initialisieren
- Erstellen Sie einen Replikationszeitplan über AWX/Tower für Oracle-Binärdateien, -Datenbanken und -Protokolle

- Stellen Sie Oracle DB auf dem Ziel wieder her und bringen Sie die Datenbank in den Online-Modus

Klicken Sie anschließend auf "Hier sind erste Schritte mit der Lösung".

Erste Schritte

Diese Lösung wurde für den Betrieb in einer AWX/Tower-Umgebung entwickelt.

AWX/Tower

Für AWX-/Tower-Umgebungen werden Sie geleitet durch das Erstellen einer Bestandsaufnahme für das ONTAP Cluster-Management und den Oracle Server (IPs und Hostnamen), das Erstellen von Anmeldeinformationen, das Konfigurieren eines Projekts, das den Ansible-Code aus NetApp Automation Github zieht, und durch die Jobvorlage, die die Automatisierung startet.

1. Die Lösung wurde für die Ausführung in einem Private-Cloud-Szenario (vor Ort und lokal) und in einer Hybrid Cloud (On-Premises zu Public-Cloud-Cloud Volumes ONTAP [CVO]) entwickelt.
2. Füllen Sie die Variablen aus, die für Ihre Umgebung spezifisch sind, und kopieren Sie sie in die Felder Extra Vars in Ihrer Job-Vorlage.
3. Nachdem die zusätzlichen Vars zu Ihrer Job-Vorlage hinzugefügt wurden, können Sie die Automatisierung starten.
4. Die Automatisierung umfasst drei Phasen (Setup, Replizierungsplan für Oracle Binaries, Database, Logs und Replication Schedule nur für Logs) und einen vierten Schritt zur Wiederherstellung der Datenbank an einem DR-Standort.
5. Detaillierte Anweisungen zum Abrufen der für den CVO-Datenschutz erforderlichen Schlüssel und Token finden Sie unter "Sammeln von Voraussetzungen für CVO- und Connector-Implementierungen"

Anforderungen

-Eingeg.-

Umgebung	Anforderungen
Ansible-Umgebung	AWX/Tower
	Ansible v.2.10 und höher
	Python 3
	Python Libraries - netapp-lib - xmltodict - jmespath
ONTAP	ONTAP Version 9.8 +
	Zwei Datenaggregate
	NFS vlan und iffrp wurden erstellt
Oracle Server	RHEL 7/8
	Oracle Linux 7/8
	Netzwerkschnittstellen für das NFS-, öffentlichen und optionalen Management
	Vorhandene Oracle Umgebung auf Quelle und das entsprechende Linux Betriebssystem am Zielort (DR-Standort oder Public Cloud)

<Strong class=„big“>CVO

Umgebung	Anforderungen
Ansible-Umgebung	AWX/Tower
	Ansible v.2.10 und höher
	Python 3
	Python Libraries - netapp-lib - xmltodict - jmespath
ONTAP	ONTAP Version 9.8 +
	Zwei Datenaggregate
	NFS vlan und iffrp wurden erstellt
Oracle Server	RHEL 7/8
	Oracle Linux 7/8
	Netzwerkschnittstellen für das NFS-, öffentlichen und optionalen Management
	Vorhandene Oracle Umgebung auf Quelle und das entsprechende Linux Betriebssystem am Zielort (DR-Standort oder Public Cloud)
	Legen Sie auf der Oracle EC2-Instanz angemessenen Swap-Speicherplatz fest. Standardmäßig sind einige EC2-Instanzen mit 0-Swap bereitgestellt
Cloud Manager/AWS	AWS Zugriff/geheimer Schlüssel
	NetApp Cloud Manager Konto
	NetApp Cloud Manager – Token für die Aktualisierung
	Fügen Sie Cluster-Quell-LIFs zur AWS Security-Gruppe hinzu

Automatisierungsdetails

-Eingeg.-

Diese automatisierte Implementierung basiert auf einem einzigen Ansible-Playbook, das aus drei separaten Rollen besteht. Rollen sind Konfigurationen von ONTAP, Linux und Oracle. In der folgenden Tabelle werden die automatisierten Aufgaben beschrieben.

Playbook	Aufgaben
ontap_Setup	Vorabprüfung der ONTAP-Umgebung
	Erstellung von Intercluster LIFs am Quell-Cluster (OPTIONAL)
	Erstellung von Intercluster LIFs am Ziel-Cluster (OPTIONAL)
	Erstellung von Cluster- und SVM-Peering
	Erstellung des Ziel-SnapMirror und Initialisierung designierter Oracle Volumes
Ora_Replication_cg	Aktivieren Sie den Backup-Modus für jede Datenbank in /etc/oratab
	Snapshot von Oracle Binary und Datenbank-Volumes erstellt
	Snapmirror Aktualisiert
	Deaktivieren Sie den Backup-Modus für jede Datenbank in /etc/oratab
Ora_Replication_log	Schalten Sie das aktuelle Protokoll für jede Datenbank in /etc/oratab um
	Snapshot vom Oracle Log Volume erstellt
	Snapmirror Aktualisiert
Ora_Erholung	SnapMirror unterbrechen
	Aktivieren Sie NFS und erstellen Sie Verbindungspfad für Oracle Volumes auf dem Ziel
	Konfigurieren Sie den Oracle-Host für DR
	Mounten und überprüfen Sie Oracle Volumes
	Stellen Sie die Oracle Datenbank wieder her und starten Sie sie

<Strong class=„big“>CVO

Diese automatisierte Implementierung basiert auf einem einzigen Ansible-Playbook, das aus drei separaten Rollen besteht. Rollen sind Konfigurationen von ONTAP, Linux und Oracle. In der folgenden Tabelle werden die automatisierten Aufgaben beschrieben.

Playbook	Aufgaben
cvo_Setup	Vorabprüfung der Umgebung
	AWS konfigurieren/AWS Zugriffsschlüssel-ID/geheimer Schlüssel/Standardregion
	Erstellung der AWS Rolle
	Erstellung der NetApp Cloud Manager Connector-Instanz in AWS
	Erstellung der Cloud Volumes ONTAP-Instanz (CVO) in AWS
	Fügen Sie ein On-Premises-Quell-ONTAP-Cluster zu NetApp Cloud Manager hinzu
	Erstellung des Ziel-SnapMirror und Initialisierung designierter Oracle Volumes
Ora_Replication_cg	Aktivieren Sie den Backup-Modus für jede Datenbank in /etc/oratab
	Snapshot von Oracle Binary und Datenbank-Volumes erstellt
	Snapmirror Aktualisiert
	Deaktivieren Sie den Backup-Modus für jede Datenbank in /etc/oratab
Ora_Replication_log	Schalten Sie das aktuelle Protokoll für jede Datenbank in /etc/oratab um
	Snapshot vom Oracle Log Volume erstellt
	Snapmirror Aktualisiert
Ora_Erholung	SnapMirror unterbrechen
	Aktivieren Sie NFS und erstellen Sie den Verbindungspfad für Oracle Volumes auf dem Ziel-CVO
	Konfigurieren Sie den Oracle-Host für DR
	Mounten und überprüfen Sie Oracle Volumes
	Stellen Sie die Oracle Datenbank wieder her und starten Sie sie

Standardparameter

Um die Automatisierung zu vereinfachen, haben wir viele erforderliche Oracle Parameter mit Standardwerten voreingestellt. In der Regel ist es nicht erforderlich, die Standardparameter für die meisten Implementierungen zu ändern. Ein fortgeschrittener Benutzer kann mit Vorsicht Änderungen an den Standardparametern vornehmen. Die Standardparameter befinden sich in jedem Rollenordner unter dem Standardverzeichnis.

Lizenz

Sie sollten die Lizenzinformationen wie im Github-Repository angegeben lesen. Durch Zugriff, Herunterladen, Installation oder Nutzung des Inhalts in diesem Repository stimmen Sie den Bedingungen der Lizenz zu "Hier".

Beachten Sie, dass es bestimmte Beschränkungen bezüglich der Erstellung und/oder Freigabe abgeleiteter Werke mit dem Inhalt in diesem Repository gibt. Bitte lesen Sie die Bedingungen des "Lizenz" Vor der Verwendung des Inhalts. Wenn Sie nicht mit allen Bedingungen einverstanden sind, dürfen Sie den Inhalt in diesem Repository nicht aufrufen, herunterladen oder verwenden.

Klicken Sie anschließend auf "Hier finden Sie ausführliche AWX/Tower-Verfahren".

Schritt-für-Schritt-Anweisungen zur Implementierung

Auf dieser Seite wird die automatisierte Datensicherung von Oracle19c auf NetApp ONTAP Storage beschrieben.

AWX/Tower Oracle Data Protection

Erstellen Sie Inventar, Gruppe, Hosts und Anmeldedaten für Ihre Umgebung

In diesem Abschnitt wird die Einrichtung von Inventar, Gruppen, Hosts und Zugangsdaten im AWX/Ansible Tower beschrieben, die die Umgebung für den Einsatz automatisierter NetApp Lösungen vorbereiten.

1. Konfigurieren Sie den Bestand.
    a. Navigieren Sie zu Ressourcen → Inventar → Hinzufügen, und klicken Sie auf Inventar hinzufügen.
    b. Geben Sie den Namen und die Organisationsdetails ein, und klicken Sie auf Speichern.
    c. Klicken Sie auf der Seite Inventar auf den erstellten Bestand.
    d. Navigieren Sie zum Untermenü Gruppen, und klicken Sie auf Hinzufügen.
    e. Geben Sie den Namen oracle für Ihre erste Gruppe ein, und klicken Sie auf Speichern.
    f. Wiederholen Sie den Vorgang für eine zweite Gruppe namens dr_oracle.
    g. Wählen Sie die erstellte oracle-Gruppe aus, gehen Sie zum Untermenü Hosts und klicken Sie auf Neuen Host hinzufügen.
    h. Geben Sie die IP-Adresse der Management-IP des Oracle Quell-Hosts an, und klicken Sie auf Speichern.
    i. Dieser Prozess muss für die dr_oracle-Gruppe wiederholt werden und die Management-IP/den Host für DR/Ziel Oracle-Host hinzufügen.

Im Folgenden werden die Typen und Anmeldedaten für Zugangsdaten, entweder für On-Premises mit ONTAP oder CVO in AWS, erstellt.

Lokal

1. Konfigurieren Sie die Anmeldedaten.
2. Credential-Typen Erstellen. Bei Lösungen, die ONTAP verwenden, müssen Sie den Anmeldeinformationstyp so konfigurieren, dass er mit den Einträgen für Benutzernamen und Kennwort übereinstimmt.
   a. Navigieren Sie zu Administration → Credential Types, und klicken Sie auf Add.
   b. Geben Sie den Namen und eine Beschreibung an.
   c. Fügen Sie den folgenden Inhalt in die Eingabekonfiguration ein:

```
fields:
  - id: dst_cluster_username
    type: string
    label: Destination Cluster Username
  - id: dst_cluster_password
    type: string
    label: Destination Cluster Password
    secret: true
  - id: src_cluster_username
    type: string
    label: Source Cluster Username
  - id: src_cluster_password
    type: string
    label: Source Cluster Password
    secret: true
```

   d. Fügen Sie den folgenden Inhalt in die Konfiguration des Injektors ein, und klicken Sie dann auf Speichern:

```
extra_vars:
  dst_cluster_username: '{{ dst_cluster_username }}'
  dst_cluster_password: '{{ dst_cluster_password }}'
  src_cluster_username: '{{ src_cluster_username }}'
  src_cluster_password: '{{ src_cluster_password }}'
```

3. Credential für ONTAP erstellen
   a. Navigieren Sie zu Resources → Credentials, und klicken Sie auf Add.
   b. Geben Sie den Namen und die Organisationsdetails für die ONTAP Credentials ein
   c. Wählen Sie den im vorherigen Schritt erstellten Anmeldeinformationstyp aus.
   d. Geben Sie unter „Typdetails“ den Benutzernamen und das Kennwort für Ihre Quell- und Zielcluster ein.
   e. Klicken Sie Auf Speichern
4. Credential für Oracle erstellen

a. Navigieren Sie zu Resources → Credentials, und klicken Sie auf Add.

b. Geben Sie den Namen und die Organisationsdetails für Oracle ein

c. Wählen Sie den Typ der Geräteanmeldeinformationen aus.

d. Geben Sie unter „Typdetails“ den Benutzernamen und das Kennwort für die Oracle-Hosts ein.

e. Wählen Sie die richtige Privilege-Eskalationsmethode aus, und geben Sie Benutzernamen und Kennwort ein.

f. Klicken Sie Auf Speichern

g. Wiederholen Sie den Vorgang, falls dies für eine andere Anmeldedaten für den dr_oracle-Host erforderlich ist.

CVO

1. Konfigurieren Sie die Anmeldedaten.
2. Erstellen von Anmeldungstypen. Bei Lösungen, die ONTAP nutzen, müssen Sie den Anmeldeinformationstyp so konfigurieren, dass er mit den Einträgen für Benutzername und Passwort übereinstimmt, werden auch Einträge für Cloud Central und AWS hinzugefügt.
   a. Navigieren Sie zu Administration → Credential Types, und klicken Sie auf Add.
   b. Geben Sie den Namen und eine Beschreibung an.
   c. Fügen Sie den folgenden Inhalt in die Eingabekonfiguration ein:

```
fields:
  - id: dst_cluster_username
    type: string
    label: CVO Username
  - id: dst_cluster_password
    type: string
    label: CVO Password
    secret: true
  - id: cvo_svm_password
    type: string
    label: CVO SVM Password
    secret: true
  - id: src_cluster_username
    type: string
    label: Source Cluster Username
  - id: src_cluster_password
    type: string
    label: Source Cluster Password
    secret: true
  - id: regular_id
    type: string
    label: Cloud Central ID
    secret: true
  - id: email_id
    type: string
    label: Cloud Manager Email
    secret: true
  - id: cm_password
    type: string
    label: Cloud Manager Password
    secret: true
  - id: access_key
    type: string
    label: AWS Access Key
    secret: true
  - id: secret_key
    type: string
    label: AWS Secret Key
    secret: true
  - id: token
    type: string
    label: Cloud Central Refresh Token
    secret: true
```

d. Fügen Sie den folgenden Inhalt in die Konfiguration des Injektors ein, und klicken Sie auf

Speichern:

```
extra_vars:
  dst_cluster_username: '{{ dst_cluster_username }}'
  dst_cluster_password: '{{ dst_cluster_password }}'
  cvo_svm_password: '{{ cvo_svm_password }}'
  src_cluster_username: '{{ src_cluster_username }}'
  src_cluster_password: '{{ src_cluster_password }}'
  regular_id: '{{ regular_id }}'
  email_id: '{{ email_id }}'
  cm_password: '{{ cm_password }}'
  access_key: '{{ access_key }}'
  secret_key: '{{ secret_key }}'
  token: '{{ token }}'
```

3. Credential für ONTAP/CVO/AWS erstellen
   a. Navigieren Sie zu Resources → Credentials, und klicken Sie auf Add.
   b. Geben Sie den Namen und die Organisationsdetails für die ONTAP Credentials ein
   c. Wählen Sie den im vorherigen Schritt erstellten Anmeldeinformationstyp aus.
   d. Geben Sie unter „Type Details“ den Benutzernamen und das Kennwort für Ihre Quell- und CVO-Cluster, Cloud Central/Manager, AWS Access/Secret Key und Cloud Central Refresh Token ein.
   e. Klicken Sie Auf Speichern
4. Credential für Oracle (Quelle) erstellen
   a. Navigieren Sie zu Resources → Credentials, und klicken Sie auf Add.
   b. Geben Sie den Namen und die Organisationsdetails für Oracle Host ein
   c. Wählen Sie den Typ der Geräteanmeldeinformationen aus.
   d. Geben Sie unter „Typdetails“ den Benutzernamen und das Kennwort für die Oracle-Hosts ein.
   e. Wählen Sie die richtige Privilege-Eskalationsmethode aus, und geben Sie Benutzernamen und Kennwort ein.
   f. Klicken Sie Auf Speichern
5. Credential für Oracle Destination erstellen
   a. Navigieren Sie zu Resources → Credentials, und klicken Sie auf Add.
   b. Geben Sie den Namen und die Organisationsdetails für den DR Oracle-Host ein
   c. Wählen Sie den Typ der Geräteanmeldeinformationen aus.
   d. Geben Sie unter „Typdetails“ den Benutzernamen (ec2-user oder wenn Sie ihn von der Standardeinstellung geändert haben, geben Sie diesen ein) und den SSH Private Key ein
   e. Wählen Sie die richtige Methode zur Eskalation von Berechtigungen (sudo) aus, und geben Sie bei Bedarf den Benutzernamen und das Kennwort ein.
   f. Klicken Sie Auf Speichern

Erstellen Sie ein Projekt

1. Gehen Sie zu Ressourcen → Projekte, und klicken Sie auf Hinzufügen.
   a. Geben Sie den Namen und die Organisationsdetails ein.
   b. Wählen Sie im Feld Quellenkontrolle Credential Type die Option Git aus.
   c. Eingabe <https://github.com/NetApp-Automation/na_oracle19c_data_protection.git> Als URL für die Quellensteuerung.
   d. Klicken Sie auf Speichern .
   e. Das Projekt muss gelegentlich synchronisiert werden, wenn sich der Quellcode ändert.

Globale Variablen konfigurieren

Die in diesem Abschnitt definierten Variablen gelten für alle Oracle Hosts, Datenbanken und den ONTAP Cluster.

1. Geben Sie Ihre umgebungsspezifischen Parameter in das folgende eingebettete globale Variablen oder Vars-Formular ein.

Die blauen Elemente müssen an Ihre Umgebung angepasst werden.

Lokal

```
# Oracle Data Protection global user configuration variables
# Ontap env specific config variables
hosts_group: "ontap"
ca_signed_certs: "false"

# Inter-cluster LIF details
src_nodes:
  - "AFF-01"
  - "AFF-02"

dst_nodes:
  - "DR-AFF-01"
  - "DR-AFF-02"

create_source_intercluster_lifs: "yes"

source_intercluster_network_port_details:
  using_dedicated_ports: "yes"
  using_ifgrp: "yes"
  using_vlans: "yes"
  failover_for_shared_individual_ports: "yes"
  ifgrp_name: "a0a"
  vlan_id: "10"
  ports:
    - "e0b"
    - "e0g"
  broadcast_domain: "NFS"
  ipspace: "Default"
  failover_group_name: "iclifs"

source_intercluster_lif_details:
  - name: "icl_1"
    address: "10.0.0.1"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "AFF-01"
  - name: "icl_2"
    address: "10.0.0.2"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "AFF-02"

create_destination_intercluster_lifs: "yes"
```

```
destination_intercluster_network_port_details:
  using_dedicated_ports: "yes"
  using_ifgrp: "yes"
  using_vlans: "yes"
  failover_for_shared_individual_ports: "yes"
  ifgrp_name: "a0a"
  vlan_id: "10"
  ports:
    - "e0b"
    - "e0g"
  broadcast_domain: "NFS"
  ipspace: "Default"
  failover_group_name: "iclifs"

destination_intercluster_lif_details:
  - name: "icl_1"
    address: "10.0.0.3"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "DR-AFF-01"
  - name: "icl_2"
    address: "10.0.0.4"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "DR-AFF-02"

# Variables for SnapMirror Peering
passphrase: "your-passphrase"

# Source & Destination List
dst_cluster_name: "dst-cluster-name"
dst_cluster_ip: "dst-cluster-ip"
dst_vserver: "dst-vserver"
dst_nfs_lif: "dst-nfs-lif"
src_cluster_name: "src-cluster-name"
src_cluster_ip: "src-cluster-ip"
src_vserver: "src-vserver"

# Variable for Oracle Volumes and SnapMirror Details
cg_snapshot_name_prefix: "oracle"
src_orabinary_vols:
  - "binary_vol"
src_db_vols:
  - "db_vol"
src_archivelog_vols:
  - "log_vol"
```

```
snapmirror_policy: "async_policy_oracle"

# Export Policy Details
export_policy_details:
  name: "nfs_export_policy"
  client_match: "0.0.0.0/0"
  ro_rule: "sys"
  rw_rule: "sys"

# Linux env specific config variables
mount_points:
  - "/u01"
  - "/u02"
  - "/u03"
hugepages_nr: "1234"
redhat_sub_username: "xxx"
redhat_sub_password: "xxx"

# DB env specific install and config variables
recovery_type: "scn"
control_files:
  - "/u02/oradata/CDB2/control01.ctl"
  - "/u03/orareco/CDB2/control02.ctl"
```

CVO

```
############################################
### Ontap env specific config variables ###
############################################

#Inventory group name
#Default inventory group name - "ontap"
#Change only if you are changing the group name either in
inventory/hosts file or in inventory groups in case of AWX/Tower
hosts_group: "ontap"

#CA_signed_certificates (ONLY CHANGE to "true" IF YOU ARE USING CA
SIGNED CERTIFICATES)
ca_signed_certs: "false"

#Names of the Nodes in the Source ONTAP Cluster
src_nodes:
  - "AFF-01"
  - "AFF-02"

#Names of the Nodes in the Destination CVO Cluster
```

```
dst_nodes:
  - "DR-AFF-01"
  - "DR-AFF-02"

#Define whether or not to create intercluster lifs on source cluster
(ONLY CHANGE to "No" IF YOU HAVE ALREADY CREATED THE INTERCLUSTER LIFS)
create_source_intercluster_lifs: "yes"

source_intercluster_network_port_details:
  using_dedicated_ports: "yes"
  using_ifgrp: "yes"
  using_vlans: "yes"
  failover_for_shared_individual_ports: "yes"
  ifgrp_name: "a0a"
  vlan_id: "10"
  ports:
    - "e0b"
    - "e0g"
  broadcast_domain: "NFS"
  ipspace: "Default"
  failover_group_name: "iclifs"

source_intercluster_lif_details:
  - name: "icl_1"
    address: "10.0.0.1"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "AFF-01"
  - name: "icl_2"
    address: "10.0.0.2"
    netmask: "255.255.255.0"
    home_port: "a0a-10"
    node: "AFF-02"

##########################################
### CVO Deployment Variables ###
##########################################

####### Access Keys Variables ######

# Region where your CVO will be deployed.
region_deploy: "us-east-1"

########### CVO and Connector Vars ########

# AWS Managed Policy required to give permission for IAM role creation.
```

```
aws_policy: "arn:aws:iam::1234567:policy/OCCM"

# Specify your aws role name, a new role is created if one already does
not exist.
aws_role_name: "arn:aws:iam::1234567:policy/OCCM"

# Name your connector.
connector_name: "awx_connector"

# Name of the key pair generated in AWS.
key_pair: "key_pair"

# Name of the Subnet that has the range of IP addresses in your VPC.
subnet: "subnet-12345"

# ID of your AWS secuirty group that allows access to on-prem
resources.
security_group: "sg-123123123"

# You Cloud Manager Account ID.
account: "account-A23123A"

# Name of the your CVO instance
cvo_name: "test_cvo"

# ID of the VPC in AWS.
vpc: "vpc-123123123"

#############################################################################
############################
# Variables for - Add on-prem ONTAP to Connector in Cloud Manager
#############################################################################
############################

# For Federated users, Client ID from API Authentication Section of
Cloud Central to generate access token.
sso_id: "123123123123123123123"

# For regular access with username and password, please specify "pass"
as the connector_access. For SSO users, use "refresh_token" as the
variable.
connector_access: "pass"

#############################################################################
############################
# Variables for SnapMirror Peering
#############################################################################
```

```
##############################
passphrase: "your-passphrase"

#########################################################################
##############################
# Source & Destination List
#########################################################################
##############################
#Please Enter Destination Cluster Name
dst_cluster_name: "dst-cluster-name"

#Please Enter Destination Cluster (Once CVO is Created Add this
Variable to all templates)
dst_cluster_ip: "dst-cluster-ip"

#Please Enter Destination SVM to create mirror relationship
dst_vserver: "dst-vserver"

#Please Enter NFS Lif for dst vserver (Once CVO is Created Add this
Variable to all templates)
dst_nfs_lif: "dst-nfs-lif"

#Please Enter Source Cluster Name
src_cluster_name: "src-cluster-name"

#Please Enter Source Cluster
src_cluster_ip: "src-cluster-ip"

#Please Enter Source SVM
src_vserver: "src-vserver"

#########################################################################
##############################
# Variable for Oracle Volumes and SnapMirror Details
#########################################################################
##############################
#Please Enter Source Snapshot Prefix Name
cg_snapshot_name_prefix: "oracle"

#Please Enter Source Oracle Binary Volume(s)
src_orabinary_vols:
  - "binary_vol"
#Please Enter Source Database Volume(s)
src_db_vols:
  - "db_vol"
#Please Enter Source Archive Volume(s)
```

```
src_archivelog_vols:
  - "log_vol"
#Please Enter Destination Snapmirror Policy
snapmirror_policy: "async_policy_oracle"

###########################################################################
##############################
# Export Policy Details
###########################################################################
##############################
#Enter the destination export policy details (Once CVO is Created Add
this Variable to all templates)
export_policy_details:
  name: "nfs_export_policy"
  client_match: "0.0.0.0/0"
  ro_rule: "sys"
  rw_rule: "sys"

###########################################################################
##############################
### Linux env specific config variables ###
###########################################################################
##############################

#NFS Mount points for Oracle DB volumes
mount_points:
  - "/u01"
  - "/u02"
  - "/u03"

# Up to 75% of node memory size divided by 2mb. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB.
# Leave it blank if hugepage is not configured on the host.
hugepages_nr: "1234"

# RedHat subscription username and password
redhat_sub_username: "xxx"
redhat_sub_password: "xxx"

#####################################################
### DB env specific install and config variables ###
#####################################################
#Recovery Type (leave as scn)
recovery_type: "scn"
```

```
#Oracle Control Files
control_files:
  - "/u02/oradata/CDB2/control01.ctl"
  - "/u03/orareco/CDB2/control02.ctl"
```

Automation Playbooks

Es gibt vier separate Playbooks, die ausgeführt werden müssen.

1. Playbook zur Einrichtung Ihrer Umgebung, vor Ort oder CVO
2. Playbook für die Replizierung von Oracle Binaries und Datenbanken nach einem Zeitplan
3. Playbook für die Replizierung von Oracle Logs nach einem Zeitplan
4. Playbook für die Wiederherstellung Ihrer Datenbank auf einem Ziel-Host

ONTAP/CVO Einrichtung

ONTAP und CVO Setup

Konfigurieren und starten Sie die Jobvorlage.

1. Erstellen Sie die Job-Vorlage.
   a. Navigieren Sie zu Ressourcen → Vorlagen → Hinzufügen, und klicken Sie auf Job Template hinzufügen.
   b. Geben Sie den Namen „ONTAP/CVO Setup“ ein
   c. Wählen Sie den Jobtyp aus; Ausführen konfiguriert das System anhand eines Playbooks.
   d. Wählen Sie den entsprechenden Bestand, das Projekt, das Playbook und die Zugangsdaten für das Playbook aus.
   e. Wählen Sie das Playbook „ontap_Setup.yml“ für eine On-Premises-Umgebung aus oder wählen Sie das playbook cvo_Setup.yml zur Replizierung in eine CVO Instanz aus.
   f. Fügen Sie globale Variablen, die aus Schritt 4 kopiert wurden, in das Feld Vorlagenvariablen unter der Registerkarte YAML ein.
   g. Klicken Sie auf Speichern .
2. Starten Sie die Jobvorlage.
   a. Navigieren Sie zu Ressourcen → Vorlagen.
   b. Klicken Sie auf die gewünschte Vorlage und dann auf Starten.

   Wir verwenden diese Vorlage und kopieren sie in andere Playbooks.

Replizierung für Binär- und Datenbank-Volumes

Planung des Binary and Database Replication Playbook

Konfigurieren und starten Sie die Jobvorlage.

1. Kopieren Sie die zuvor erstellte Jobvorlage.
   a. Navigieren Sie zu Ressourcen → Vorlagen.
   b. Suchen Sie die ONTAP/CVO Setup-Vorlage und klicken Sie rechts ganz auf Copy Template
   c. Klicken Sie auf Vorlage bearbeiten in der kopierten Vorlage, und ändern Sie den Namen in Binary and Database Replication Playbook.
   d. Behalten Sie für die Vorlage denselben Bestand, dasselbe Projekt und dieselben Anmeldeinformationen bei.
   e. Wählen Sie das Playbook ora_Replication_cg.yml als ausführtes Playbook aus.
   f. Die Variablen bleiben die gleichen, aber die CVO Cluster-IP muss in der Variablen dst_Cluster_ip festgelegt werden.
   g. Klicken Sie auf Speichern .
2. Planen Sie die Jobvorlage.
   a. Navigieren Sie zu Ressourcen → Vorlagen.
   b. Klicken Sie auf die Playbook-Vorlage „Binary and Database Replication“, und klicken Sie anschließend oben auf „Schedules“.

c. Klicken Sie auf Hinzufügen, fügen Sie den Namenszeitplan für die Binärdatei und die Datenbankreplikation hinzu, wählen Sie das Startdatum/die Startzeit am Anfang der Stunde, wählen Sie die Zeitzone Lokale Zeitzone und die Häufigkeit aus. Ausführungshäufigkeit wird häufig aktualisiert, dass die SnapMirror Replizierung aktualisiert wird.

> Für die Log-Volume-Replizierung wird ein separater Zeitplan erstellt, sodass der Zeitplan in einer häufigeren Kadenz repliziert werden kann.

Replizierung für Protokoll-Volumes

Planen des Log Replication Playbook

Konfigurieren und starten Sie die Jobvorlage

1. Kopieren Sie die zuvor erstellte Jobvorlage.
   a. Navigieren Sie zu Ressourcen → Vorlagen.
   b. Suchen Sie die ONTAP/CVO Setup-Vorlage und klicken Sie rechts ganz auf Copy Template
   c. Klicken Sie auf Vorlage bearbeiten in der kopierten Vorlage, und ändern Sie den Namen in Log Replication Playbook.
   d. Behalten Sie für die Vorlage denselben Bestand, dasselbe Projekt und dieselben Anmeldeinformationen bei.
   e. Wählen Sie als auszuführenden Playbook die ora_Replication_logs.yml aus.
   f. Die Variablen bleiben die gleichen, aber die CVO Cluster-IP muss in der Variablen dst_Cluster_ip festgelegt werden.
   g. Klicken Sie auf Speichern .
2. Planen Sie die Jobvorlage.
   a. Navigieren Sie zu Ressourcen → Vorlagen.
   b. Klicken Sie auf die Playbook-Vorlage für Protokollreplikation, und klicken Sie anschließend oben auf „Schedules“.
   c. Klicken Sie auf Hinzufügen, fügen Sie den Namensplan für die Protokollreplizierung hinzu, wählen Sie das Startdatum/die Startzeit am Beginn der Stunde, wählen Sie die Zeitzone Lokal und die Häufigkeit der Ausführung aus. Ausführungshäufigkeit wird häufig aktualisiert, dass die SnapMirror Replizierung aktualisiert wird.

> Es wird empfohlen, den Protokollplan so einzustellen, dass er jede Stunde aktualisiert wird, um sicherzustellen, dass die Wiederherstellung auf die letzte stündliche Aktualisierung erfolgt.

Wiederherstellen und Wiederherstellen von Datenbanken

Planen des Log Replication Playbook

Konfigurieren und starten Sie die Jobvorlage.

1. Kopieren Sie die zuvor erstellte Jobvorlage.
   a. Navigieren Sie zu Ressourcen → Vorlagen.
   b. Suchen Sie die ONTAP/CVO Setup-Vorlage und klicken Sie rechts ganz auf Copy Template

c. Klicken Sie auf Vorlage bearbeiten auf der kopierten Vorlage, und ändern Sie den Namen in „Playbook wiederherstellen und wiederherstellen“.

d. Behalten Sie für die Vorlage denselben Bestand, dasselbe Projekt und dieselben Anmeldeinformationen bei.

e. Wählen Sie die ora_Recovery.yml als auszuführenden Playbook aus.

f. Die Variablen bleiben die gleichen, aber die CVO Cluster-IP muss in der Variablen dst_Cluster_ip festgelegt werden.

g. Klicken Sie auf Speichern .

Dieses Playbook wird erst ausgeführt, nachdem Sie bereit sind, Ihre Datenbank am Remote-Standort wiederherzustellen.

Oracle Database Wird Wiederhergestellt

1. Daten-Volumes für Oracle-Produktionsdatenbanken vor Ort werden über NetApp SnapMirror Replizierung auf einen redundanten ONTAP Cluster im sekundären Datacenter oder Cloud Volume ONTAP in der Public Cloud gesichert. In einer vollständig konfigurierten Disaster-Recovery-Umgebung sind die Recovery von Computing-Instanzen im sekundären Datacenter oder in der Public Cloud Standby und im Notfall bereit, die Produktionsdatenbank wiederherzustellen. Die Standby-Computing-Instanzen werden mit On-Prem-Instanzen synchronisiert, indem paraellel-Updates auf OS-Kernel-Patch ausgeführt oder ein Upgrade in einem Lockstep durchgeführt wird.
2. In dieser demonstrierten Lösung wird das Oracle Binary Volume zum Ziel repliziert und an einer Zielinstanz gemountet, um den Oracle Software Stack zu erstellen. Dieser Ansatz zur Wiederherstellung von Oracle hat den Vorteil, dass Oracle in letzter Minute bei einem Ausfall neu installiert wird. Es garantiert, dass die Oracle Installation vollständig mit der aktuellen Installation der On-Prem-Produktionssoftware und den Patch-Leveln synchronisiert ist. Dies kann jedoch je nach Struktur der Softwarelizenzierung mit Oracle für das replizierte Oracle Binary Volume am Recovery-Standort zusätzliche Konsequenzen haben oder diese nicht haben. Der Benutzer wird empfohlen, sich mit seinem Softwarelizenzierungspersonal zu erkundigen, um die potenziellen Lizenzierungsanforderungen für Oracle zu bewerten, bevor er sich für denselben Ansatz entscheidet.
3. Der Standby-Oracle-Host am Ziel ist mit den Oracle-Vorbedingung-Konfigurationen konfiguriert.
4. Die SnapMirror-Spiegelungen werden beschädigt und die Volumes werden beschreibbar gemacht und auf den Standby-Oracle Host eingebunden.
5. Das Oracle Recovery-Modul führt die folgenden Aufgaben zur Wiederherstellung und dem Start von Oracle am Recovery-Standort aus, nachdem alle DB-Volumes auf der Standby-Compute-Instanz gemountet wurden.
    a. Sync the Control file: Wir haben duplizierte Oracle Steuerdateien auf verschiedenen Datenbank-Volumes implementiert, um die kritische Datenbankkontrolldatei zu schützen. Eine ist auf dem Daten-Volume und eine ist auf dem Log-Volume. Da Daten und Protokoll-Volumes unterschiedlich häufig repliziert werden, sind sie zum Zeitpunkt der Wiederherstellung nicht synchron.
    b. Relink Oracle Binary: Da die Oracle-Binärdatei auf einen neuen Host verlagert wird, braucht es eine Relink.
    c. Recovery von Oracle Datenbank: Der Recovery-Mechanismus ruft die letzte Systemänderungsnummer in der letzten verfügbaren archivierten Protokolldatei von Oracle ab und stellt die Oracle Datenbank wieder her, um alle Geschäftstransaktionen wiedergewonnen zu haben, die zum Zeitpunkt eines Ausfalls auf den DR-Standort repliziert werden konnten. Die Datenbank wird dann in einer neuen Inkarnation gestartet, um Benutzerverbindungen und Geschäftstransaktionen am Recovery-Standort durchzuführen.

Bevor Sie das Recovery-Playbook ausführen, stellen Sie sicher, dass Sie Folgendes haben:
Vergewissern Sie sich, dass es über /etc/oratab und /etc/oraInst.loc vom Oracle-Quellhost zum Zielhost kopiert wird