



# **On-Premises/Hybrid-Cloud**

## **NetApp Solutions**

NetApp  
May 14, 2024

This PDF was generated from [https://docs.netapp.com/de-de/netapp-solutions/databases/automation\\_ora\\_asa\\_iscsi.html](https://docs.netapp.com/de-de/netapp-solutions/databases/automation_ora_asa_iscsi.html) on May 14, 2024. Always check docs.netapp.com for the latest.

# Inhalt

- On-Premises/Hybrid-Cloud ..... 1
  - TR-4983: Vereinfachte, automatisierte Oracle-Implementierung auf NetApp ASA mit iSCSI ..... 1
  - NVA-1155: Oracle 19c RAC-Datenbanken on FlexPod Datacenter with Cisco UCS and NetApp AFF A800 over FC - Design and Deployment Guide..... 17
  - TR-4250: SAP with Oracle on UNIX and NFS with NetApp Clustered Data ONTAP and SnapManager for SAP 3.4 ..... 17
  - Implementieren Von Oracle Database ..... 17
  - Lösungsüberblick ..... 39
  - TR-4794: Oracle Databases on NetApp EF-Series ..... 63

# On-Premises/Hybrid-Cloud

## TR-4983: Vereinfachte, automatisierte Oracle-Implementierung auf NetApp ASA mit iSCSI

Allen Cao, Niyaz Mohamed, NetApp

### Zweck

NetApp ASA Systeme bieten moderne Lösungen für Ihre SAN-Infrastruktur. Sie vereinfachen die Skalierung und ermöglichen es Ihnen, geschäftskritische Applikationen wie Datenbanken zu beschleunigen, stets verfügbar zu sein (99.9999 % Uptime) sowie TCO und Kohlendioxidausstoß zu reduzieren. Die NetApp ASA Systeme umfassen Modelle der A-Serie, die für Applikationen mit höchsten Performance-Anforderungen konzipiert wurden, sowie Modelle der C-Serie, die für kostengünstige Implementierungen mit großer Kapazität optimiert sind. Gemeinsam bieten die Systeme der ASA A-Series und C-Series eine außergewöhnliche Performance, die eine bessere Kundenerfahrung ermöglicht, Ergebnisse schneller liefert, geschäftskritische Daten verfügbar, geschützt und sicher hält, und sie bieten mehr effektive Kapazität für alle Workloads, unterstützt mit der effektivsten Garantie der Branche.

In dieser Dokumentation wird die vereinfachte Implementierung von Oracle-Datenbanken in einer SAN-Umgebung mit ASA Systemen mithilfe von Ansible-Automatisierung demonstriert. Die Oracle-Datenbank wird in einer eigenständigen Neustartkonfiguration mit iSCSI-Protokoll für den Datenzugriff und Oracle ASM für das Management von Datenbankfestplatten auf dem ASA Storage-Array bereitgestellt. Darüber hinaus bietet es mit dem UI-Tool NetApp SnapCenter Informationen zum Backup, zur Wiederherstellung und zum Klonen von Oracle Datenbanken für einen speichereffizienten Betrieb von Datenbanken in NetApp ASA Systemen.

Diese Lösung eignet sich für folgende Anwendungsfälle:

- Automatisierte Implementierung von Oracle Datenbanken in NetApp ASA Systemen als primären Datenbank-Storage
- Backup und Restore von Oracle Datenbanken in NetApp ASA Systemen mit dem Tool NetApp SnapCenter
- Klonen von Oracle Datenbanken für Entwicklung/Test oder andere Anwendungsfälle in NetApp ASA Systemen mit dem Tool NetApp SnapCenter

### Zielgruppe

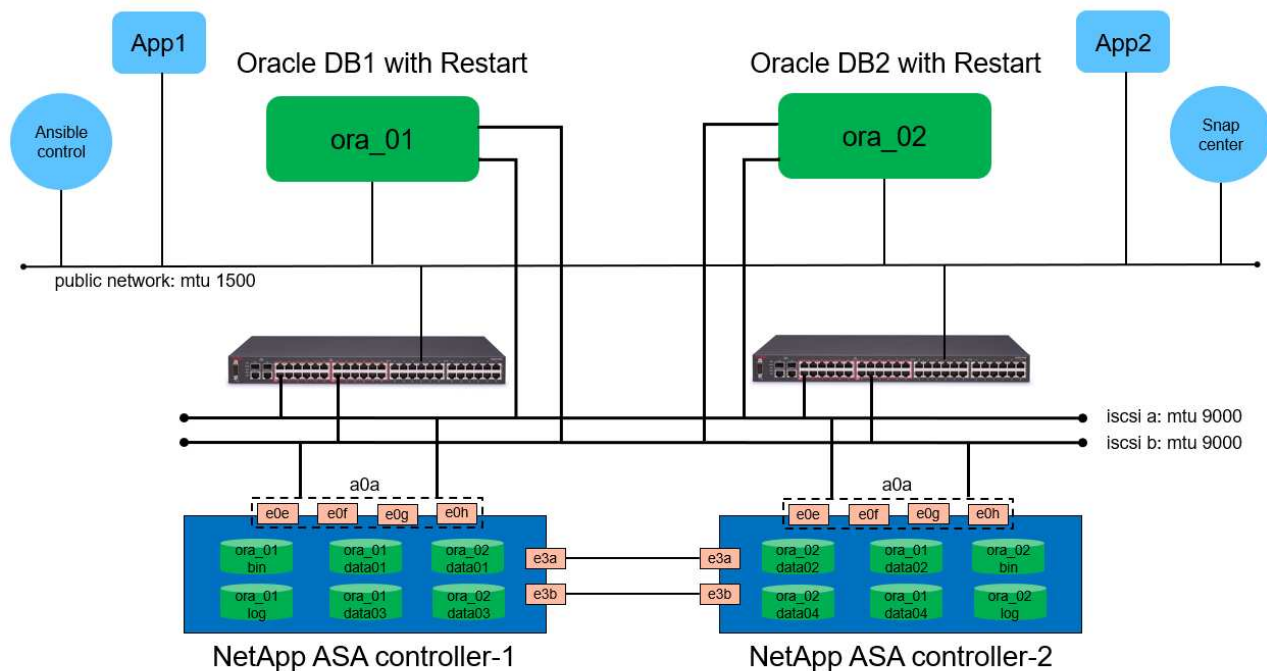
Diese Lösung ist für folgende Personen gedacht:

- Ein DBA, der Oracle in NetApp ASA Systemen einsetzen möchte.
- Ein Solution Architect für Datenbanken, der Oracle-Workloads in NetApp ASA-Systemen testen möchte.
- Ein Storage-Administrator, der eine Oracle Datenbank auf NetApp ASA Systemen implementieren und managen möchte.
- Ein Applikationseigentümer, der eine Oracle Datenbank in NetApp ASA Systemen einrichten möchte.

### Test- und Validierungsumgebung der Lösung

Die Lösung wurde in einer Testumgebung getestet und validiert. Siehe Abschnitt [\[Key Factors for Deployment Consideration\]](#) Finden Sie weitere Informationen.

## Simplified, Automated Oracle Database Deployment on NetApp ASA with iSCSI



NetApp

### Hardware- und Softwarekomponenten

Hardware		
NetApp ASA A400	Version 9.13.1P1	2 NS224-Shelfs, 48 NVMe-AFF-Laufwerke mit einer Gesamtkapazität von 69.3 tib
UCSB-B200-M4	Intel® Xeon® CPU E5-2690 v4 @ 2,60 GHz	VMware ESXi Cluster mit 4 Nodes
Software		
Redhat Linux	RHEL-8.6, 4.18.0-372.9.1.el8.x86_64-Kernel	Bereitstellung der RedHat Subscription für Tests
Windows Server	2022 Standard, 10.0.20348 Build 20348	Hosting von SnapCenter-Servern
Oracle Grid Infrastructure	Version 19.18	RU-Patch p34762026_190000_Linux-x86-64.zip angewendet
Oracle Datenbank	Version 19.18	RU-Patch p34765931_190000_Linux-x86-64.zip angewendet
Oracle OPatch	Version 12.2.0.1.36	Neuestes Patch p6880880_190000_Linux-x86-64.zip
SnapCenter Server	Version 4.9P1	Workgroup-Bereitstellung

VMware vSphere Hypervisor	Version 6.5.0.20000	VMware Tools, Version: 11365 - Linux, 12352 - Windows
Öffnen Sie JDK	Version java-1.8.0-openjdk.x86_64	Anforderungen für SnapCenter Plugin auf DB VMs

## Konfiguration der Oracle-Datenbank in der Laborumgebung

Server	* Datenbank*	DB-Speicher
ora_01	NTAP1(NTAP1_PDB1,NTAP1_PDB2,NTAP1_PDB3)	ISCSI-luns auf ASA A400
ora_02	NTAP2(NTAP2_PDB1,NTAP2_PDB2,NTAP2_PDB3)	ISCSI-luns auf ASA A400

## Wichtige Faktoren für die Implementierung

- **Speicherlayout der Oracle-Datenbank.** in dieser automatisierten Oracle-Bereitstellung stellen wir vier Datenbankvolumes bereit, um Oracle-Binärdaten, -Daten und -Protokolle standardmäßig zu hosten. Dann erstellen wir zwei ASM-Festplattengruppen aus Daten und Protokoll-luns. Innerhalb der +DATA asm-Festplattengruppe stellen wir zwei Daten-luns in einem Volume auf jedem ASA A400-Cluster-Node bereit. Innerhalb der +LOGS asm-Festplattengruppe erstellen wir zwei luns in einem Protokoll-Volume auf einem einzelnen ASA A400 Knoten. Mehrere luns in einem ONTAP Volume bieten im Allgemeinen eine bessere Performance.
- **Implementierung mehrerer DB-Server.** die Automatisierungslösung kann eine Oracle-Container-Datenbank auf mehreren DB-Servern in einem einzelnen Ansible-Playbook bereitstellen. Unabhängig von der Anzahl der DB-Server bleibt die Playbook-Ausführung gleich. Bei Server-Implementierungen mit mehreren DB baut das Playbook mit einem Algorithmus auf, um Datenbank-luns optimal auf Dual-Controllern der ASA A400 zu platzieren. Die binären und logs luns der ungeraden Zahl DB Server in Server Hosts Index Platz auf Controller 1. Die binären und Protokolle luns der geraden Zahl DB-Server in den Server-Hosts Index Platz auf Controller 2. Die DB-Daten-luns sind gleichmäßig auf zwei Controller verteilt. Oracle ASM kombiniert die Daten-luns auf zwei Controllern in einer einzigen ASM-Festplattengruppe, um die Verarbeitungsleistung beider Controller voll auszuschöpfen.
- **ISCSI-Konfiguration.** die Datenbank-VMs werden über das iSCSI-Protokoll zum Speicherzugriff mit dem ASA-Speicher verbunden. Sie sollten duale Pfade auf jedem Controller-Knoten konfigurieren, um Redundanz zu gewährleisten, und iSCSI Multi-Path auf dem DB-Server für Multi-Path-Storage-Zugriff einrichten. Aktivieren Sie Jumbo Frame im Speichernetzwerk, um Performance und Durchsatz zu maximieren.
- **Oracle ASM-Redundanzebene, die für jede von Ihnen erstellte Oracle ASM-Laufwerksgruppe verwendet werden soll.** Da die ASA A400 den Speicher in RAID DP für den Datenschutz auf Clusterplattenebene konfiguriert, sollten Sie dies verwenden `External Redundancy`, Das bedeutet, dass die Option Oracle ASM nicht erlaubt, den Inhalt der Datenträgergruppe zu spiegeln.
- **Datenbanksicherung.** NetApp bietet eine SnapCenter Software Suite für Datenbank-Backup, -Wiederherstellung und -Klonen mit einer benutzerfreundlichen Benutzeroberfläche. NetApp empfiehlt die Implementierung eines solchen Management Tools, damit Snapshot Backups (unter einer Minute), schnelle Datenbank-Restores (in Minuten) und Datenbankklone möglich sind.

## Lösungsimplementierung

In den folgenden Abschnitten werden schrittweise Verfahren für die automatisierte Bereitstellung und den Schutz von Oracle 19c in NetApp ASA A400 mit direkt gemounteten Datenbank-luns über iSCSI an DB-VM in

einem einzelnen Knoten beschrieben. Starten Sie die Konfiguration mit Oracle ASM als Datenbank-Volume-Manager neu.

## Voraussetzungen für die Bereitstellung

Die Bereitstellung erfordert die folgenden Voraussetzungen.

1. Es wird davon ausgegangen, dass das NetApp ASA-Speicher-Array installiert und konfiguriert wurde. Dies umfasst iSCSI-Broadcast-Domäne, LACP-Schnittstellengruppen a0a auf beiden Controller-Nodes, iSCSI-VLAN-Ports (a0a-<iscsi-a-vlan-id>, a0a-<iscsi-b-vlan-id>) auf beiden Controller-Nodes. Unter dem folgenden Link finden Sie detaillierte Schritt-für-Schritt-Anleitungen, wenn Sie Hilfe benötigen. "[Detaillierter Leitfaden - ASA A400](#)"
2. Stellen Sie eine Linux-VM als Ansible-Controller-Node bereit, wobei die neueste Version von Ansible und Git installiert ist. Details finden Sie unter folgendem Link: "[Erste Schritte mit der Automatisierung von NetApp Lösungen](#)". In Abschnitt - Setup the Ansible Control Node for CLI deployments on RHEL / CentOS Oder Setup the Ansible Control Node for CLI deployments on Ubuntu / Debian.
3. Klonen Sie eine Kopie des NetApp Toolkit zur Implementierungsautomatisierung für iSCSI.

```
git clone https://bitbucket.ngage.netapp.com/scm/ns-  
bb/na_oracle_deploy_iscsi.git
```

4. Stellen Sie einen Windows-Server bereit, um das UI-Tool NetApp SnapCenter mit der neuesten Version auszuführen. Details finden Sie unter folgendem Link: "[Installieren Sie den SnapCenter-Server](#)".
5. Erstellen Sie zwei RHEL Oracle DB Server entweder Bare Metal oder virtualisierte VM. Erstellen Sie einen Admin-Benutzer auf DB-Servern mit sudo ohne Passwortberechtigung und aktivieren Sie die SSH-Authentifizierung für private/öffentliche Schlüssel zwischen Ansible-Host und Oracle DB-Server-Hosts. Stellen Sie die folgenden Oracle 19c-Installationsdateien auf DB-Servern /tmp/Archive-Verzeichnis bereit.

```
installer_archives:  
  - "LINUX.X64_193000_grid_home.zip"  
  - "p34762026_190000_Linux-x86-64.zip"  
  - "LINUX.X64_193000_db_home.zip"  
  - "p34765931_190000_Linux-x86-64.zip"  
  - "p6880880_190000_Linux-x86-64.zip"
```



Stellen Sie sicher, dass Sie mindestens 50G im Oracle VM Root-Volume zugewiesen haben, um ausreichend Speicherplatz für die Erstellung von Oracle Installationsdateien zu haben.

6. Sehen Sie sich das folgende Video an:

[Vereinfachte und automatisierte Oracle-Implementierung auf NetApp ASA mit iSCSI](#)

## Automatisierungsparameter-Dateien

In dem Playbook „Ansible“ werden die Installations- und Konfigurationsaufgaben von Datenbanken mit vordefinierten Parametern ausgeführt. Für diese Oracle-Automatisierungslösung gibt es drei benutzerdefinierte Parameterdateien, die vor der Ausführung des Playbooks Benutzereingaben erfordern.

- Hosts: Legen Sie Ziele fest, für die das Automatisierungs-Playbook ausgeführt wird.
- vars/vars.yml - die globale Variablendatei, die Variablen definiert, die für alle Ziele gelten.
- Host\_VARS/Host\_Name.yml - die lokale Variablendatei, die Variablen definiert, die nur auf ein lokales Ziel angewendet werden. In unserem Anwendungsbeispiel handelt es sich um die Oracle DB-Server.

Zusätzlich zu diesen benutzerdefinierten Variablendateien gibt es mehrere standardmäßige Variablendateien, die Standardparameter enthalten, die nicht geändert werden müssen, sofern dies nicht erforderlich ist. Die folgenden Abschnitte zeigen, wie die benutzerdefinierten Variablendateien konfiguriert werden.

## Konfiguration von Parameterdateien

## 1. Ansible Ziel hosts Dateikonfiguration:

```
# Enter NetApp ASA controller management IP address
[ontap]
172.16.9.32

# Enter Oracle servers names to be deployed one by one, follow by
each Oracle server public IP address, and ssh private key of admin
user for the server.
[oracle]
ora_01 ansible_host=10.61.180.21 ansible_ssh_private_key_file
=ora_01.pem
ora_02 ansible_host=10.61.180.23 ansible_ssh_private_key_file
=ora_02.pem
```

## 2. Weltweit vars/vars.yml Dateikonfiguration

```
#####
#####
#####
                                Oracle 19c deployment global user
configurable variables                                #####
#####
                                Consolidate all variables from ONTAP, linux
and oracle                                #####
#####
#####

#####
#####
#####
                                ONTAP env specific config variables
#####
#####
#####
#####

# Enter the supported ONTAP platform: on-prem, aws-fsx.
ontap_platform: on-prem

# Enter ONTAP cluster management user credentials
username: "xxxxxxxx"
password: "xxxxxxxx"

##### on-prem platform specific user defined variables #####

# Enter Oracle SVM iSCSI lif addresses. Each controller configures
```



```

with dual paths iscsi_a, iscsi_b for redundancy
ora_iscsi_lif_mgmt:
  - {name: '{{ svm_name }}_lif_mgmt', address: 172.21.253.220, netmask:
255.255.255.0, vlan_name: ora_mgmt, vlan_id: 3509}

ora_iscsi_lifs_node1:
  - {name: '{{ svm_name }}_lif_1a', address: 172.21.234.221,
netmask: 255.255.255.0, vlan_name: ora_iscsi_a, vlan_id: 3490}
  - {name: '{{ svm_name }}_lif_1b', address: 172.21.235.221,
netmask: 255.255.255.0, vlan_name: ora_iscsi_b, vlan_id: 3491}
ora_iscsi_lifs_node2:
  - {name: '{{ svm_name }}_lif_2a', address: 172.21.234.223,
netmask: 255.255.255.0, vlan_name: ora_iscsi_a, vlan_id: 3490}
  - {name: '{{ svm_name }}_lif_2b', address: 172.21.235.223,
netmask: 255.255.255.0, vlan_name: ora_iscsi_b, vlan_id: 3491}

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

# Enter RHEL subscription to enable repo
redhat_sub_username: xxxxxxxx
redhat_sub_password: "xxxxxxxx"

#####
#####
###                               Oracle DB env specific config variables
###
#####
#####

# Enter Database domain name
db_domain: solutions.netapp.com

# Enter initial password for all required Oracle passwords. Change
them after installation.
initial_pwd_all: xxxxxxxx

```

### 3. Lokaler DB-Server host\_vars/host\_name.yml Konfiguration

```
# User configurable Oracle host specific parameters

# Enter container database SID. By default, a container DB is
created with 3 PDBs within the CDB
oracle_sid: NTAP1

# Enter database shared memory size or SGA. CDB is created with SGA
at 75% of memory_limit, MB. The grand total of SGA should not exceed
75% available RAM on node.
memory_limit: 8192
```

## Ausführung des Playbook

Das Automatisierungs-Toolkit enthält insgesamt sechs Playbooks. Jede führt unterschiedliche Aufgabenblöcke aus und erfüllt unterschiedliche Zwecke.

```
0-all_playbook.yml - execute playbooks from 1-4 in one playbook run.  
1-ansible_requirements.yml - set up Ansible controller with required  
libs and collections.  
2-linux_config.yml - execute Linux kernel configuration on Oracle DB  
servers.  
3-ontap_config.yml - configure ONTAP svm/volumes/luns for Oracle  
database and grant DB server access to luns.  
4-oracle_config.yml - install and configure Oracle on DB servers for  
grid infrastructure and create a container database.  
5-destroy.yml - optional to undo the environment to dismantle all.
```

Es gibt drei Optionen, um Playbooks mit den folgenden Befehlen auszuführen.

1. Führen Sie alle Playbooks für die Implementierung in einem kombinierten Durchlauf aus.

```
ansible-playbook -i hosts 0-all_playbook.yml -u admin -e  
@vars/vars.yml
```

2. Führen Sie Playbooks einzeln mit der Zahlenfolge von 1 bis 4 aus.

```
ansible-playbook -i hosts 1-ansible_requirements.yml -u admin -e  
@vars/vars.yml
```

```
ansible-playbook -i hosts 2-linux_config.yml -u admin -e  
@vars/vars.yml
```

```
ansible-playbook -i hosts 3-ontap_config.yml -u admin -e  
@vars/vars.yml
```

```
ansible-playbook -i hosts 4-oracle_config.yml -u admin -e  
@vars/vars.yml
```

3. Führen Sie 0-all\_Playbook.yml mit einem Tag aus.

```
ansible-playbook -i hosts 0-all_playbook.yml -u admin -e  
@vars/vars.yml -t ansible_requirements
```

```
ansible-playbook -i hosts 0-all_playbook.yml -u admin -e  
@vars/vars.yml -t linux_config
```

```
ansible-playbook -i hosts 0-all_playbook.yml -u admin -e  
@vars/vars.yml -t ontap_config
```

```
ansible-playbook -i hosts 0-all_playbook.yml -u admin -e  
@vars/vars.yml -t oracle_config
```

#### 4. Die Umgebung rückgängig machen

```
ansible-playbook -i hosts 5-destroy.yml -u admin -e @vars/vars.yml
```

### Validierung nach der Ausführung

Melden Sie sich nach der Ausführung des Playbooks als oracle-Benutzer beim oracle DB-Server an, um zu überprüfen, ob die Grid-Infrastruktur und die Datenbank von Oracle erfolgreich erstellt wurden. Im Folgenden sehen Sie ein Beispiel für die Validierung von Oracle-Datenbanken auf Host ora\_01.

## 1. Die Grid-Infrastruktur und die erstellten Ressourcen validieren

```
[oracle@ora_01 ~]$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
devtmpfs	7.7G	40K	7.7G	1%	/dev
tmpfs	7.8G	1.1G	6.7G	15%	/dev/shm
tmpfs	7.8G	312M	7.5G	4%	/run
tmpfs	7.8G	0	7.8G	0%	/sys/fs/cgroup
/dev/mapper/rhel-root	44G	38G	6.8G	85%	/
/dev/sda1	1014M	258M	757M	26%	/boot
tmpfs	1.6G	12K	1.6G	1%	/run/user/42
tmpfs	1.6G	4.0K	1.6G	1%	/run/user/1000
/dev/mapper/ora_01_biny_01p1	40G	21G	20G	52%	/u01

```
[oracle@ora_01 ~]$ asm
[oracle@ora_01 ~]$ crsctl stat res -t
```

```
-----
-----
Name          Target  State          Server          State
details
-----
-----
Local Resources
-----
-----
ora.DATA.dg
              ONLINE  ONLINE         ora_01          STABLE
ora.LISTENER.lsnr
              ONLINE  INTERMEDIATE   ora_01          Not All
Endpoints Re
gistered, STABLE
ora.LOGS.dg
              ONLINE  ONLINE         ora_01          STABLE
ora.asm
              ONLINE  ONLINE         ora_01
Started, STABLE
ora.ons
              OFFLINE OFFLINE        ora_01          STABLE
-----
-----
Cluster Resources
-----
```

```

-----
ora.cssd
  1          ONLINE  ONLINE          ora_01          STABLE
ora.diskmon
  1          OFFLINE OFFLINE          STABLE
ora.driver.afd
  1          ONLINE  ONLINE          ora_01          STABLE
ora.evmd
  1          ONLINE  ONLINE          ora_01          STABLE
ora.ntap1.db
  1          ONLINE  ONLINE          ora_01
Open,HOME=/u01/app/o

racle/product/19.0.0

/NTAP1,STABLE
-----
-----
[oracle@ora_01 ~]$

```



Ignorieren Sie die Not All Endpoints Registered Unter Statusdetails. Dies resultiert aus einem Konflikt der manuellen und dynamischen Datenbankregistrierung mit dem Listener und kann sicher ignoriert werden.

2. Überprüfen Sie, ob der ASM-Filtertreiber wie erwartet funktioniert.


```

[oracle@ora_01 ~]$ asmcmd
ASMCMD> lsdg
State      Type      Rebal  Sector  Logical_Sector  Block      AU
Total_MB   Free_MB   Req_mir_free_MB  Usable_file_MB  Offline_disks
Voting_files  Name
MOUNTED    EXTERN    N      512      512      4096      4194304
327680     318644      0      318644      0
N  DATA/
MOUNTED    EXTERN    N      512      512      4096      4194304
81920      78880      0      78880      0
N  LOGS/
ASMCMD> lsdk
Path
AFD:ORA_01_DAT1_01
AFD:ORA_01_DAT1_03
AFD:ORA_01_DAT1_05
AFD:ORA_01_DAT1_07
AFD:ORA_01_DAT2_02
AFD:ORA_01_DAT2_04
AFD:ORA_01_DAT2_06
AFD:ORA_01_DAT2_08
AFD:ORA_01_LOGS_01
AFD:ORA_01_LOGS_02
ASMCMD> afd_state
ASMCMD-9526: The AFD state is 'LOADED' and filtering is 'ENABLED' on
host 'ora_01'
ASMCMD>


```

3. Melden Sie sich bei Oracle Enterprise Manager Express an, um die Datenbank zu validieren.

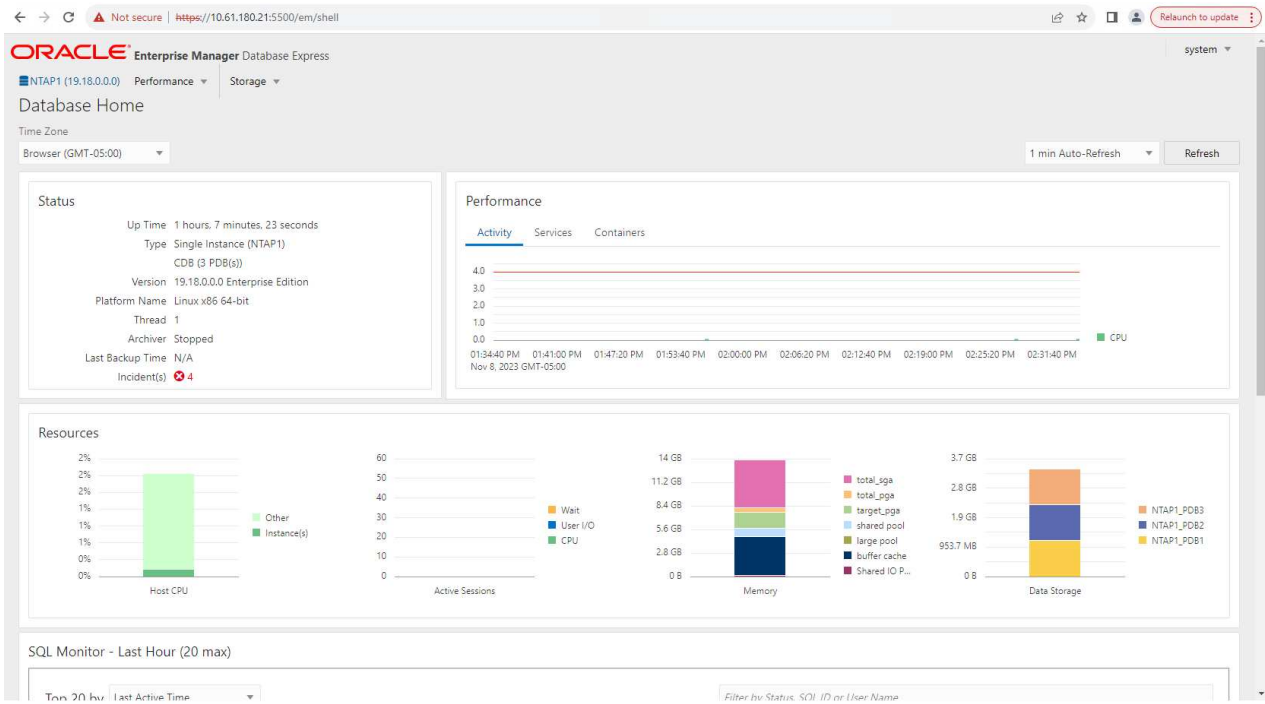
← → ↻ ⚠ Not secure | https://10.61.180.21:5500/em/login



# ORACLE ENTERPRISE MANAGER DATABASE EXPRESS



Copyright 2013, 2020, Oracle and/or its affiliates. All rights reserved.





Enable additional port from sqlplus for login to individual container database or PDBs.

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	NTAP1_PDB1	READ WRITE	NO
4	NTAP1_PDB2	READ WRITE	NO
5	NTAP1_PDB3	READ WRITE	NO

```
SQL> alter session set container=NTAP1_PDB1;
```

Session altered.

```
SQL> select dbms_xdb_config.gethttpsport() from dual;
```

```
DBMS_XDB_CONFIG.GETHTTPS()
-----
0
```

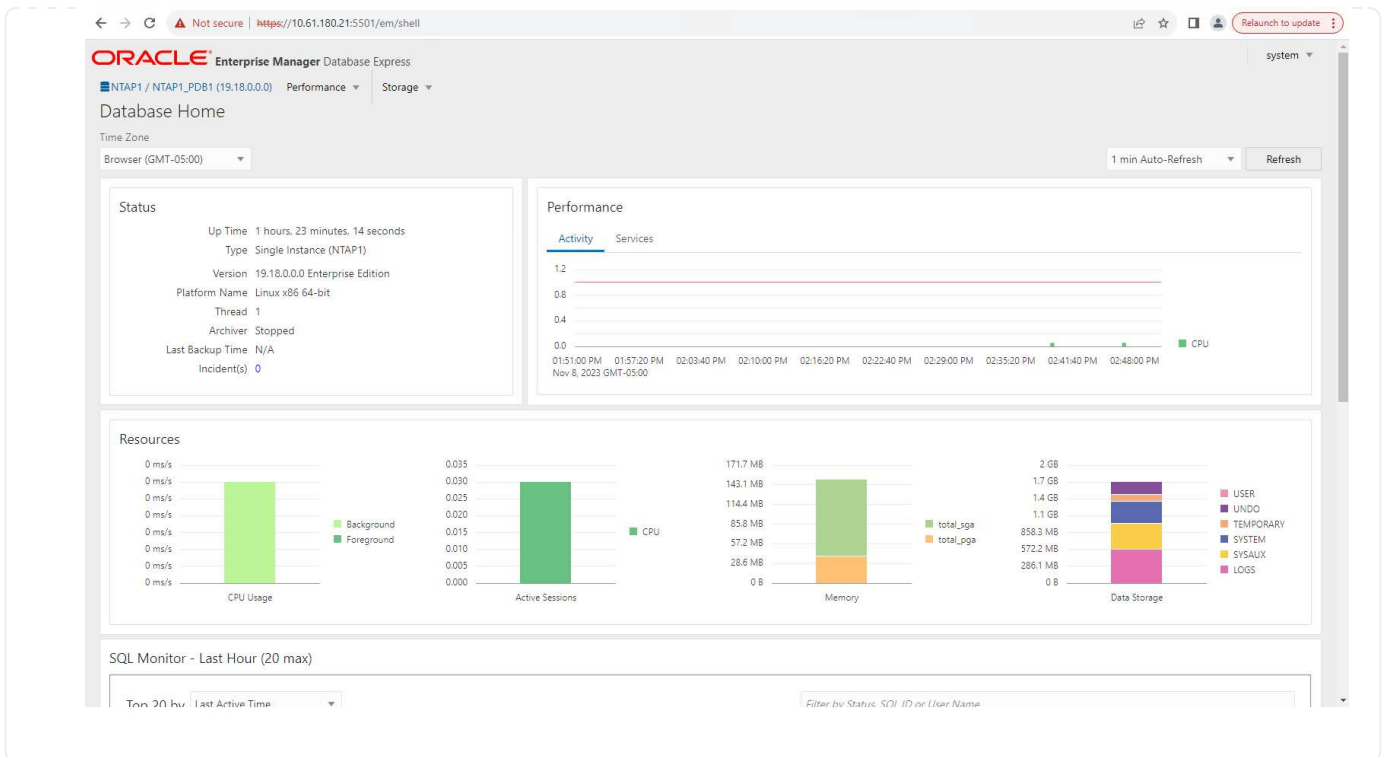
```
SQL> exec DBMS_XDB_CONFIG.SETHTTPS(5501);
```

PL/SQL procedure successfully completed.

```
SQL> select dbms_xdb_config.gethttpsport() from dual;
```

```
DBMS_XDB_CONFIG.GETHTTPS()
-----
5501
```

login to NTAP1\_PDB1 from port 5501.



## Backup, Wiederherstellung und Klonen von Oracle mit SnapCenter

Siehe TR-4979 "[Vereinfachtes, automatisiertes Oracle in VMware Cloud on AWS mit Gast-Mounted FSX ONTAP](#)" Abschnitt Oracle backup, restore, and clone with SnapCenter Bietet Details zur Einrichtung von SnapCenter und zur Ausführung von Datenbank-Backup-, Wiederherstellungs- und Klon-Workflows.

## Wo Sie weitere Informationen finden

Weitere Informationen zu den in diesem Dokument beschriebenen Daten finden Sie in den folgenden Dokumenten bzw. auf den folgenden Websites:

- NetApp ASA: REIN FLASH-BASIERTES SAN-ARRAY

["https://www.netapp.com/data-storage/all-flash-san-storage-array/"](https://www.netapp.com/data-storage/all-flash-san-storage-array/)

- Installieren der Oracle Grid-Infrastruktur für einen eigenständigen Server mit einer neuen Datenbankinstallation

["https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-oracle-grid-infrastructure-for-a-standalone-server-with-a-new-database-installation.html#GUID-0B1CEE8C-C893-46AA-8A6A-7B5FAAEC72B3"](https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-oracle-grid-infrastructure-for-a-standalone-server-with-a-new-database-installation.html#GUID-0B1CEE8C-C893-46AA-8A6A-7B5FAAEC72B3)

- Installieren und Konfigurieren von Oracle Database mithilfe von Antwortdateien

["https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-and-configuring-oracle-database-using-response-files.html#GUID-D53355E9-E901-4224-9A2A-B882070EDDF7"](https://docs.oracle.com/en/database/oracle/oracle-database/19/ladbi/installing-and-configuring-oracle-database-using-response-files.html#GUID-D53355E9-E901-4224-9A2A-B882070EDDF7)

- Verwenden Sie Red hat Enterprise Linux 8.2 mit ONTAP

## **NVA-1155: Oracle 19c RAC-Datenbanken on FlexPod Datacenter with Cisco UCS and NetApp AFF A800 over FC - Design and Deployment Guide**

Allen Cao, NetApp

Dieser Design- und Implementierungsleitfaden für Oracle 19c RAC-Datenbanken unter FlexPod Datacenter mit Cisco UCS und NetApp AFF A800 über FC bietet Details zum Lösungsdesign und schrittweise Implementierungsprozesse für das Hosting von Oracle RAC-Datenbanken auf der neuesten FlexPod Datacenter-Infrastruktur mit Oracle Linux 8.2 Betriebssystem und einen Red hat kompatiblen Kernel.

["NVA-1155: Oracle 19c RAC-Datenbanken on FlexPod Datacenter with Cisco UCS and NetApp AFF A800 over FC"](#)

## **TR-4250: SAP with Oracle on UNIX and NFS with NetApp Clustered Data ONTAP and SnapManager for SAP 3.4**

Nils Bauer, NetApp

TR-4250 erläutert die Herausforderungen bei der Entwicklung von Storage-Lösungen zur Unterstützung von SAP Business Suite Produkten unter Verwendung einer Oracle Database. Das Hauptaugenmerk dieses Dokuments liegt auf dem allgemeinen Bedarf an Storage-Infrastruktur-Design, -Implementierung, -Betrieb und -Management-Herausforderungen. Geschäftliche und IT-Führungskräfte stehen dabei vor der Herausforderung, die auf der neuesten Generation von SAP-Lösungen basieren. Die Empfehlungen in diesem Dokument sind allgemein, nicht spezifisch für eine SAP-Anwendung oder Größe und Umfang der SAP-Implementierung. TR-4250 geht davon aus, dass der Leser über die grundlegenden Kenntnisse der Technologie und des Betriebs der NetApp- und SAP-Produkte verfügt. TR-4250 wurde entwickelt nach dem Zusammenspiel des technischen Personals von NetApp, SAP, Oracle und unseren Kunden.

["TR-4250: SAP with Oracle on UNIX and NFS with NetApp Clustered Data ONTAP and SnapManager for SAP 3.4"](#)

## **Implementieren Von Oracle Database**

### **Lösungsüberblick**

#### **Automated Deployment of Oracle19c for ONTAP on NFS**

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 sich mit Ansible die Bereitstellung und Konfiguration von Oracle 19c mit NetApp ONTAP automatisieren lässt. Da Storage-Administratoren, Systemadministratoren und DBAs in der Lage sind, neuen Storage konsistent und schnell bereitzustellen, Datenbankserver zu konfigurieren und Oracle 19c Software zu installieren, profitieren Sie von folgenden Vorteilen:

- Vermeiden Sie Designkomplexität und menschliche Fehler und implementieren Sie eine wiederholbare, konsistente Implementierung und Best Practices

- Verkürzung der Zeit für Provisionierung von Speicher, Konfiguration von DB-Hosts sowie für Oracle-Installation
- Erhöhen Sie die Produktivität von Datenbank-Administratoren, -Systemen und -Storage-Administratoren
- Vereinfachen Sie die Skalierung von Storage und Datenbanken

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:

- Erstellen und Konfigurieren von ONTAP-NFS-Storage für Oracle-Datenbank
- Installation von Oracle 19c auf RedHat Enterprise Linux 7/8 oder Oracle Linux 7/8
- Konfiguration von Oracle 19c auf ONTAP NFS Storage

Weitere Informationen erhalten Sie in den Übersichtsvideos unten.

### **AWX/Tower-Implementierungen**

Teil 1: Erste Schritte, Anforderungen, Automatisierungsdetails und erste AWX/Tower-Konfiguration

#### [AWX-Bereitstellung](#)

Teil 2: Variablen und Ausführen des Playbooks

#### [AWX Playbook ausführen](#)

### **CLI-Implementierung**

Teil 1: Erste Schritte, Anforderungen, Automatisierungsdetails und Ansible Control Host Setup

#### [CLI-Implementierung](#)

Teil 2: Variablen und Ausführen des Playbooks

#### [CLI Playbook ausgeführt](#)

### **Erste Schritte**

Diese Lösung wurde für die Ausführung in einer AWX/Tower-Umgebung oder über CLI auf einem Ansible-Kontroll-Host 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. Füllen Sie die Variablen aus, die für Ihre Umgebung spezifisch sind, und kopieren Sie sie in die Felder Extra Vars in Ihrer Job-Vorlage.
2. Nachdem die zusätzlichen Vars zu Ihrer Job-Vorlage hinzugefügt wurden, können Sie die Automatisierung starten.
3. Die Job-Vorlage wird in drei Phasen ausgeführt, indem Tags für `ontap_config`, `linux_config` und

oracle\_config angegeben werden.

#### CLI über den Ansible-Steuerhost

1. Konfigurieren des Linux-Hosts, sodass er als Ansible-Steuerhost verwendet werden kann"[Klicken Sie hier für detaillierte Anweisungen](#)"
2. Nach der Konfiguration des Ansible-Steuerhosts können Sie das Ansible Automation-Repository git klonen.
3. Bearbeiten Sie die Hostdatei mit den IPs und/oder Hostnamen Ihrer ONTAP-Clusterverwaltung und der Management-IPs des Oracle-Servers.
4. Geben Sie die Variablen ein, die für Ihre Umgebung spezifisch sind, und kopieren Sie sie in die `vars.yml` Datei:
5. Jeder Oracle Host verfügt über eine variable Datei, die über einen Hostnamen mit Host-spezifischen Variablen identifiziert wird.
6. Nachdem alle variablen Dateien abgeschlossen wurden, können Sie das Playbook in drei Phasen ausführen, indem Sie Tags für angeben `ontap_config`, `linux_config`, und `oracle_config`.

#### Anforderungen

Umgebung	Anforderungen
Ansible-Umgebung	AWX/Tower- oder Linux-Host, um der Ansible-Steuerungshost zu sein
	Ansible v.2.10 und höher
	Python 3
	Python Libraries - netapp-lib - xmltodict - jmespath
ONTAP	ONTAP Version 9.3 - 9.7
	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
	Oracle Installationsdateien auf Oracle Servern

#### Automatisierungsdetails

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.

Rolle	Aufgaben
<b>ontap_config</b>	Vorabprüfung der ONTAP-Umgebung
	Erstellung NFS-basierter SVM für Oracle
	Erstellung einer Exportrichtlinie
	Erstellung von Volumes für Oracle
	Erstellung von NFS LIFs
<b>linux_config</b>	Erstellen von Bereitstellungspunkten und Mounten von NFS Volumes
	Überprüfen Sie die NFS-Mounts
	OS-spezifische Konfiguration
	Erstellen von Oracle Verzeichnissen
	konfigurieren Sie hugepages
	Deaktivieren Sie SELinux und den Firewall-Daemon
	Aktivieren und starten Sie den Chronyd-Dienst
	Erhöhen Sie die harte Grenze für den Dateideskriptor
	Erstellen Sie eine pam.d-Sitzungsdatei
<b>oracle_config</b>	Oracle Software-Installation
	Oracle Listener erstellen
	Erstellen Sie Oracle Datenbanken
	Konfiguration der Oracle Umgebung
	PDB-Status speichern
	Aktivieren Sie den Instanzarchivierungsmodus
	Aktivieren Sie den DNFS-Client
	Aktivieren Sie das automatische Starten und Herunterfahren der Datenbank zwischen einem Neustart des Betriebssystems

### Standardparameter

Um die Automatisierung zu vereinfachen, haben wir viele erforderliche Oracle Implementierungsparameter 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.

### Implementierungsanleitungen

Laden Sie vor dem Start die folgenden Oracle-Installations- und Patch-Dateien herunter, und legen Sie sie in den ein `/tmp/archive` Verzeichnis mit Lese-, Schreib- und Lesezugriff für alle Benutzer auf jedem zu implementierenden DB-Server. Die Automatisierungsaufgaben suchen nach den benannten Installationsdateien in diesem speziellen Verzeichnis für die Installation und Konfiguration von Oracle.

```
LINUX.X64_193000_db_home.zip -- 19.3 base installer  
p31281355_190000_Linux-x86-64.zip -- 19.8 RU patch  
p6880880_190000_Linux-x86-64.zip -- opatch version 12.2.0.1.23
```

## 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-Implementierungsverfahren](#)" Oder "[Hier geht es zur CLI-Implementierung](#)".

## Schritt-für-Schritt-Anweisungen zur Implementierung

### AWX/Tower-Implementierung der Oracle 19c-Datenbank

#### 1. Erstellen Sie das Inventar, die Gruppe, die Hosts und die Anmeldeinformationen 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. Wenn es Bestandsvariablen gibt, fügen Sie diese in das Feld Variablen ein.
  - e. Navigieren Sie zum Untermenü Gruppen, und klicken Sie auf Hinzufügen.
  - f. Geben Sie den Namen der Gruppe für ONTAP ein, fügen Sie die Gruppenvariablen ein (falls vorhanden) und klicken Sie auf Speichern.
  - g. Wiederholen Sie den Vorgang für eine andere Gruppe für Oracle.
  - h. Wählen Sie die erstellte ONTAP-Gruppe aus, gehen Sie zum Untermenü Hosts und klicken Sie auf Neuen Host hinzufügen.
  - i. Geben Sie die IP-Adresse der ONTAP-Cluster-Management-IP an, fügen Sie die Host-Variablen ein (falls vorhanden), und klicken Sie auf Speichern.
  - j. Dieser Prozess muss für die Oracle-Gruppe und für Oracle-Host(s)-Management-IP/Hostname wiederholt werden.
2. Erstellen von Anmeldungstypen. 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: username
    type: string
    label: Username
  - id: password
    type: string
    label: Password
    secret: true
  - id: vsadmin_password
    type: string
    label: vsadmin_password
    secret: true
```

1. Fügen Sie folgenden Inhalt in die Konfiguration des Injektors ein:

```
extra_vars:
  password: '{{ password }}'
  username: '{{ username }}'
  vsadmin_password: '{{ vsadmin_password }}'
```

1. Konfigurieren Sie die Anmeldedaten.

- a. Navigieren Sie zu Resources → Credentials, und klicken Sie auf Add.
- b. Geben Sie den Namen und die Organisationsdetails für ONTAP ein.
- c. Wählen Sie den benutzerdefinierten Anmeldungstyp aus, den Sie für ONTAP erstellt haben.
- d. Geben Sie unter „Type Details“ den Benutzernamen, das Passwort und das vsadmin\_password ein.
- e. Klicken Sie auf Zurück zur Anmeldeinformation, und klicken Sie auf Hinzufügen.
- f. Geben Sie den Namen und die Organisationsdetails für Oracle ein.
- g. Wählen Sie den Typ der Geräteanmeldeinformationen aus.
- h. Geben Sie unter „Typedetails“ den Benutzernamen und das Kennwort für die Oracle-Hosts ein.
- i. Wählen Sie die richtige Privilege-Eskalationsmethode aus, und geben Sie Benutzernamen und Kennwort ein.

## 2. 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\\_deploy.git](https://github.com/NetApp-Automation/na_oracle19c_deploy.git)> Als URL für die Quellensteuerung.
  - d. Klicken Sie auf Speichern .



e. Das Projekt muss gelegentlich synchronisiert werden, wenn sich der Quellcode ändert.

### 3. Oracle Host\_vars konfigurieren

Die in diesem Abschnitt definierten Variablen werden auf jeden einzelnen Oracle Server und jede Datenbank angewendet.

1. Geben Sie Ihre umgebungsspezifischen Parameter in das folgende eingebettete Oracle-Hosts-Variablen oder Host\_vars-Formular ein.



Die blauen Elemente müssen an Ihre Umgebung angepasst werden.

#### Host VARS-Konfiguration

```
#####
#####          Host Variables Configuration          #####
#####

# Add your Oracle Host
ansible_host: "10.61.180.15"

# Oracle db log archive mode: true - ARCHIVELOG or false - NOARCHIVELOG
log_archive_mode: "true"

# Number of pluggable databases per container instance identified by sid.
Pdb_name specifies the prefix for container database naming in this case
cdb2_pdb1, cdb2_pdb2, cdb2_pdb3
oracle_sid: "cdb2"
pdb_num: "3"
pdb_name: "{{ oracle_sid }}_pdb"

# CDB listener port, use different listener port for additional CDB on
same host
listener_port: "1523"

# CDB is created with SGA at 75% of memory_limit, MB. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB. The grand total SGA should not exceed 75% available RAM on node.
memory_limit: "5464"

# Set "em_configuration: DBEXPRESS" to install enterprise manager express
and choose a unique port from 5500 to 5599 for each sid on the host.
# Leave them black if em express is not installed.
em_configuration: "DBEXPRESS"
em_express_port: "5501"

# {{groups.oracle[0]}} represents first Oracle DB server as defined in
```

Oracle hosts group [oracle]. For concurrent multiple Oracle DB servers deployment, [0] will be incremented for each additional DB server. For example, `{{groups.oracle[1]}}` represents DB server 2, `{{groups.oracle[2]}}` represents DB server 3 ... As a good practice and the default, minimum three volumes is allocated to a DB server with corresponding /u01, /u02, /u03 mount points, which store oracle binary, oracle data, and oracle recovery files respectively. Additional volumes can be added by click on "More NFS volumes" but the number of volumes allocated to a DB server must match with what is defined in global vars file by volumes\_nfs parameter, which dictates how many volumes are to be created for each DB server.

host\_datastores\_nfs:

```
- {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
  lif: "172.21.94.200", size: "25"}
- {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
  lif: "172.21.94.200", size: "25"}
- {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
  lif: "172.21.94.200", size: "25"}
```

1. Füllen Sie alle Variablen in die blauen Felder ein.
2. Klicken Sie nach Abschluss der Variablen auf die Schaltfläche Kopieren im Formular, um alle Variablen zu kopieren, die an AWX oder Tower übertragen werden sollen.
3. Navigieren Sie zurück zu AWX oder Tower, und gehen Sie zu Ressourcen → Hosts, und wählen Sie und öffnen Sie die Konfigurationsseite für den Oracle-Server.
4. Klicken Sie auf der Registerkarte Details auf Bearbeiten und fügen Sie die kopierten Variablen aus Schritt 1 in das Feld Variablen unter der Registerkarte YAML ein.
5. Klicken Sie auf Speichern .
6. Wiederholen Sie diesen Vorgang für alle weiteren Oracle Server im System.

#### 4. 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.

```
#####
##### Oracle 19c deployment global user configuration variables #####
##### Consolidate all variables from ontap, linux and oracle #####
#####

#####
### 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 ONTAP Cluster
```

```
nodes:
```

- "AFF-01"
- "AFF-02"

```
#Storage VLANs
```

```
#Add additional rows for vlans as necessary
```

```
storage_vlans:
```

- {vlan\_id: "203", name: "infra\_NFS", protocol: "NFS"}

```
More Storage VLANsEnter Storage VLANs details
```

```
#Details of the Data Aggregates that need to be created
```

```
#If Aggregate creation takes longer, subsequent tasks of creating volumes  
may fail.
```

```
#There should be enough disks already zeroed in the cluster, otherwise  
aggregate create will zero the disks and will take long time
```

```
data_aggregates:
```

- {aggr\_name: "aggr01\_node01"}
- {aggr\_name: "aggr01\_node02"}

```
#SVM name
```

```
svm_name: "ora_svm"
```

```
# SVM Management LIF Details
```

```
svm_mgmt_details:
```

- {address: "172.21.91.100", netmask: "255.255.255.0", home\_port: "e0M"}

```
# NFS storage parameters when data_protocol set to NFS. Volume named after  
Oracle hosts name identified by mount point as follow for oracle DB server  
1. Each mount point dedicates to a particular Oracle files: u01 - Oracle  
binary, u02 - Oracle data, u03 - Oracle redo. Add additional volumes by  
click on "More NFS volumes" and also add the volumes list to corresponding  
host_vars as host_datastores_nfs variable. For multiple DB server  
deployment, additional volumes sets needs to be added for additional DB
```

server. Input variable "{{groups.oracle[1]}}\_u01",  
"{{groups.oracle[1]}}\_u02", and "{{groups.oracle[1]}}\_u03" as vol\_name for  
second DB server. Place volumes for multiple DB servers alternately  
between controllers for balanced IO performance, e.g. DB server 1 on  
controller node1, DB server 2 on controller node2 etc. Make sure match lif  
address with controller node.

volumes\_nfs:

- {vol\_name: "{{groups.oracle[0]}}\_u01", aggr\_name: "aggr01\_node01",  
lif: "172.21.94.200", size: "25"}
- {vol\_name: "{{groups.oracle[0]}}\_u02", aggr\_name: "aggr01\_node01",  
lif: "172.21.94.200", size: "25"}
- {vol\_name: "{{groups.oracle[0]}}\_u03", aggr\_name: "aggr01\_node01",  
lif: "172.21.94.200", size: "25"}

#NFS LIFs IP address and netmask

nfs\_lifs\_details:

- address: "172.21.94.200" #for node-1  
netmask: "255.255.255.0"
- address: "172.21.94.201" #for node-2  
netmask: "255.255.255.0"

#NFS client match

client\_match: "172.21.94.0/24"

#####  
### 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 ###
#####

db_domain: "your.domain.com"

# Set initial password for all required Oracle passwords. Change them
after installation.

initial_pwd_all: "netappl23"

```

1. Alle Variablen in blaue Felder eintragen.
2. Klicken Sie nach Abschluss der Variablen auf die Schaltfläche Kopieren im Formular, um alle Variablen zu kopieren, die an AWX oder Tower übertragen werden sollen, in die folgende Jobvorlage.

## 5. 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 und die Beschreibung ein
  - c. Wählen Sie den Jobtyp aus. Führen Sie die Konfiguration des Systems anhand eines Playbooks aus, und prüfen Sie, ob ein Playbook trocken ausgeführt wird, ohne das System tatsächlich zu konfigurieren.
  - d. Wählen Sie den entsprechenden Bestand, das Projekt, das Playbook und die Zugangsdaten für das Playbook aus.
  - e. Wählen Sie all\_Playbook.yml als Standard-Playbook aus, das ausgeführt werden soll.
  - f. Fügen Sie globale Variablen, die aus Schritt 4 kopiert wurden, in das Feld Vorlagenvariablen unter der Registerkarte YAML ein.
  - g. Aktivieren Sie das Kontrollkästchen Aufforderung zum Starten im Feld Job-Tags.
  - h. 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.
  - c. Wenn Sie beim Start nach Job-Tags gefragt werden, geben Sie Anforderungen\_config ein. Möglicherweise müssen Sie unter Requirements\_config auf die Zeile Job-Tag erstellen klicken, um die Job-Tag-Nummer einzugeben.



Requirements\_config stellt sicher, dass Sie über die richtigen Bibliotheken verfügen, um die anderen Rollen auszuführen.

1. Klicken Sie auf Weiter und dann auf Start, um den Job zu starten.

2. Klicken Sie auf Ansicht → Jobs, um die Jobausgabe und den Fortschritt zu überwachen.
3. Wenn Sie zur Einführung von Job-Tags aufgefordert werden, geben sie `ontap_config` ein. Sie müssen möglicherweise direkt unter `ontap_config` auf die Zeile „Job Tag erstellen“ klicken, um das Job-Tag einzugeben.
4. Klicken Sie auf Weiter und dann auf Start, um den Job zu starten.
5. Klicken Sie auf Ansicht → Jobs, um die Jobausgabe und den Fortschritt zu überwachen
6. Führen Sie nach Abschluss der Rolle `ontap_config` den Prozess für `linux_config` erneut aus.
7. Navigieren Sie zu Ressourcen → Vorlagen.
8. Wählen Sie die gewünschte Vorlage aus, und klicken Sie dann auf Starten.
9. Wenn Sie beim Start aufgefordert werden, geben Sie die Job-Tags in `linux_config` ein, müssen Sie möglicherweise die Zeile „Job-Tag erstellen“ direkt unter `linux_config` auswählen, um das Job-Tag einzugeben.
10. Klicken Sie auf Weiter und dann auf Start, um den Job zu starten.
11. Wählen Sie Ansicht → Jobs, um die Jobausgabe und den Fortschritt zu überwachen.
12. Führen Sie nach Abschluss der Rolle `linux_config` den Prozess für `oracle_config` erneut aus.
13. Gehen Sie zu Ressourcen → Vorlagen.
14. Wählen Sie die gewünschte Vorlage aus, und klicken Sie dann auf Starten.
15. Wenn Sie beim Start nach Job-Tags gefragt werden, geben sie `oracle_config` ein. Sie müssen möglicherweise die Zeile „Job Tag erstellen“ direkt unter `oracle_config` auswählen, um das Job-Tag einzugeben.
16. Klicken Sie auf Weiter und dann auf Start, um den Job zu starten.
17. Wählen Sie Ansicht → Jobs, um die Jobausgabe und den Fortschritt zu überwachen.

## 6. Implementieren Sie zusätzliche Datenbank auf demselben Oracle Host

Der Oracle Teil des Playbook erstellt pro Ausführung eine einzelne Oracle-Container-Datenbank auf einem Oracle-Server. Führen Sie die folgenden Schritte aus, um zusätzliche Container-Datenbanken auf demselben Server zu erstellen.

1. Host\_Vars-Variablen überarbeiten.
  - a. Zurück zu Schritt 2 - Oracle Host\_Vars konfigurieren.
  - b. Ändern Sie Oracle SID zu einer anderen Namenskonvention.
  - c. Ändern Sie den Listener-Port in eine andere Zahl.
  - d. Ändern Sie den EM Express-Port in eine andere Nummer, wenn Sie EM Express installieren.
  - e. Kopieren Sie die überarbeiteten Hostvariablen in das Feld Oracle Host Variables auf der Registerkarte Host Configuration Detail.
2. Starten Sie die Jobvorlage für die Bereitstellung nur mit dem tag `oracle_config`.
3. Melden Sie sich beim Oracle-Server als Oracle-Benutzer an und führen Sie die folgenden Befehle aus:

```
ps -ef | grep ora
```



Auf diese Weise werden die oracle-Prozesse aufgeführt, wenn die Installation wie erwartet abgeschlossen wurde und die oracle DB gestartet wurde

4. Melden Sie sich bei der Datenbank an, um die db-Konfigurationseinstellungen und die PDBs zu überprüfen, die mit den folgenden Befehlssätzen erstellt wurden.

```

[oracle@localhost ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 12:52:51 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL>

SQL> select name, log_mode from v$database;
NAME          LOG_MODE
-----
CDB2          ARCHIVELOG

SQL> show pdbs

      CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
          2 PDB$SEED                                READ ONLY  NO
          3 CDB2_PDB1                                READ WRITE NO
          4 CDB2_PDB2                                READ WRITE NO
          5 CDB2_PDB3                                READ WRITE NO

col svrname form a30
col dirname form a30
select svrname, dirname, nfsversion from v$dnfs_servers;

SQL> col svrname form a30
SQL> col dirname form a30
SQL> select svrname, dirname, nfsversion from v$dnfs_servers;

SVRNAME                                DIRNAME                                NFSVERSION
-----
172.21.126.200                        /rhelora03_u02                        NFSv3.0
172.21.126.200                        /rhelora03_u03                        NFSv3.0
172.21.126.200                        /rhelora03_u01                        NFSv3.0

```

Dies bestätigt, dass dNFS richtig funktioniert.

5. Stellen Sie über Listener eine Verbindung zur Datenbank her, um die Konfiguration des Oracle Listener mit dem folgenden Befehl zu überprüfen. Wechseln Sie zum entsprechenden Listener-Port und



## Datenbankdienstnamen.

```
[oracle@localhost ~]$ sqlplus
system@//localhost:1523/cdb2_pdb1.cie.netapp.com

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 13:19:57 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Enter password:
Last Successful login time: Wed May 05 2021 17:11:11 -04:00

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL> show user
USER is "SYSTEM"
SQL> show con_name
CON_NAME
CDB2_PDB1
```

Dies bestätigt, dass Oracle Listener ordnungsgemäß funktioniert.

## Wo Hilfe benötigt wird?

Wenn Sie Hilfe mit dem Toolkit benötigen, nehmen Sie bitte an der Teil ["NetApp Solution Automation Community Support Slack Channel"](#) Und suchen Sie den Kanal zur Lösungsautomatisierung, um Ihre Fragen zu stellen oder zu fragen.

## Schritt-für-Schritt-Anweisungen zur Implementierung

In diesem Dokument wird die Implementierung von Oracle 19c über die Befehlszeilenschnittstelle (cli) der Automatisierung beschrieben.

### CLI-Implementierung einer Oracle 19c Datenbank

In diesem Abschnitt werden die Schritte beschrieben, die für die Vorbereitung und Implementierung der Oracle19c-Datenbank mit der CLI erforderlich sind. Stellen Sie sicher, dass Sie den geprüft haben ["Erste Schritte und Abschnitt zu den Anforderungen"](#) Und Ihre Umgebung entsprechend vorbereitet.

## Oracle19c repo herunterladen

1. Führen Sie auf Ihrem ansible-Controller den folgenden Befehl aus:

```
git clone https://github.com/NetApp-Automation/na_oracle19c_deploy.git
```

2. Ändern Sie nach dem Herunterladen des Projektarchivs die Verzeichnisse in `na_oracle19c_Deploy` `<cd na_oracle19c_deploy>`.

## Bearbeiten Sie die Host-Datei

Vor der Bereitstellung Folgendes abschließen:

1. Bearbeiten Sie die Host-Datei `na_oracle19c_Deploy`.
2. Ändern Sie unter `[ONTAP]` die IP-Adresse in Ihre Cluster-Management-IP.
3. Fügen Sie unter der Gruppe `[oracle]` die namen der oracle-Hosts hinzu. Der Host-Name muss für seine IP-Adresse entweder über DNS oder über die Hosts-Datei aufgelöst werden, oder er muss im Host angegeben werden.
4. Speichern Sie nach Abschluss dieser Schritte alle Änderungen.

Im folgenden Beispiel wird eine Host-Datei dargestellt:

```
#ONTAP Host

[ontap]

"10.61.184.183"

#Oracle hosts

[oracle]

"rtpora01"

"rtpora02"
```

Dieses Beispiel führt das Playbook aus und implementiert oracle 19c gleichzeitig auf zwei oracle DB Servern. Sie können auch mit nur einem DB-Server testen. In diesem Fall müssen Sie nur eine Host-Variablendatei konfigurieren.



Das Playbook wird unabhängig davon, wie viele Oracle Hosts und Datenbanken Sie implementieren, auf dieselbe Weise ausgeführt.

## Bearbeiten Sie die Datei `Host_Name.yml` unter `Host_vars`

Jeder Oracle Host verfügt über seine Host-Variablendatei, die durch den Hostnamen identifiziert wird, der Host-spezifische Variablen enthält. Sie können einen beliebigen Namen für Ihren Host angeben. Bearbeiten und kopieren Sie die `host_vars` Fügen Sie sie im Abschnitt Host VARS Config in Ihre gewünschte ein `host_name.yml` Datei:



Die blauen Elemente müssen an Ihre Umgebung angepasst werden.

## Host VARS-Konfiguration

```
#####
##### Host Variables Configuration #####
#####

# Add your Oracle Host
ansible_host: "10.61.180.15"

# Oracle db log archive mode: true - ARCHIVELOG or false - NOARCHIVELOG
log_archive_mode: "true"

# Number of pluggable databases per container instance identified by sid.
Pdb_name specifies the prefix for container database naming in this case
cdb2_pdb1, cdb2_pdb2, cdb2_pdb3
oracle_sid: "cdb2"
pdb_num: "3"
pdb_name: "{{ oracle_sid }}_pdb"

# CDB listener port, use different listener port for additional CDB on
same host
listener_port: "1523"

# CDB is created with SGA at 75% of memory_limit, MB. Consider how many
databases to be hosted on the node and how much ram to be allocated to
each DB. The grand total SGA should not exceed 75% available RAM on node.
memory_limit: "5464"

# Set "em_configuration: DBEXPRESS" to install enterprise manager express
and choose a unique port from 5500 to 5599 for each sid on the host.
# Leave them black if em express is not installed.
em_configuration: "DBEXPRESS"
em_express_port: "5501"

# {{groups.oracle[0]}} represents first Oracle DB server as defined in
Oracle hosts group [oracle]. For concurrent multiple Oracle DB servers
deployment, [0] will be incremented for each additional DB server. For
example, {{groups.oracle[1]}} represents DB server 2,
"{{groups.oracle[2]}}" represents DB server 3 ... As a good practice and
the default, minimum three volumes is allocated to a DB server with
corresponding /u01, /u02, /u03 mount points, which store oracle binary,
oracle data, and oracle recovery files respectively. Additional volumes
can be added by click on "More NFS volumes" but the number of volumes
allocated to a DB server must match with what is defined in global vars
file by volumes_nfs parameter, which dictates how many volumes are to be
created for each DB server.
```

```

host_datastores_nfs:
  - {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
    lif: "172.21.94.200", size: "25"}

```

## Bearbeiten Sie die Datei Vars.yml

Der vars.yml Die Datei konsolidiert alle umgebungsspezifischen Variablen (ONTAP, Linux oder Oracle) für die Implementierung von Oracle.

1. Bearbeiten und kopieren Sie die Variablen aus dem Abschnitt VARS und fügen Sie diese Variablen in Ihr ein vars.yml Datei:

```

#####
##### Oracle 19c deployment global user configuration variables #####
##### Consolidate all variables from ontap, linux and oracle #####
#####

#####

### 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 ONTAP Cluster
nodes:
  - "AFF-01"
  - "AFF-02"

#Storage VLANs
#Add additional rows for vlans as necessary
storage_vlans:
  - {vlan_id: "203", name: "infra_NFS", protocol: "NFS"}
More Storage VLANsEnter Storage VLANs details

```

```

#Details of the Data Aggregates that need to be created
#If Aggregate creation takes longer, subsequent tasks of creating volumes
may fail.
#There should be enough disks already zeroed in the cluster, otherwise
aggregate create will zero the disks and will take long time
data_aggregates:
  - {aggr_name: "aggr01_node01"}
  - {aggr_name: "aggr01_node02"}

#SVM name
svm_name: "ora_svm"

# SVM Management LIF Details
svm_mgmt_details:
  - {address: "172.21.91.100", netmask: "255.255.255.0", home_port: "e0M"}

# NFS storage parameters when data_protocol set to NFS. Volume named after
Oracle hosts name identified by mount point as follow for oracle DB server
1. Each mount point dedicates to a particular Oracle files: u01 - Oracle
binary, u02 - Oracle data, u03 - Oracle redo. Add additional volumes by
click on "More NFS volumes" and also add the volumes list to corresponding
host_vars as host_datastores_nfs variable. For multiple DB server
deployment, additional volumes sets needs to be added for additional DB
server. Input variable "{{groups.oracle[1]}}_u01",
 "{{groups.oracle[1]}}_u02", and "{{groups.oracle[1]}}_u03" as vol_name for
second DB server. Place volumes for multiple DB servers alternately
between controllers for balanced IO performance, e.g. DB server 1 on
controller node1, DB server 2 on controller node2 etc. Make sure match lif
address with controller node.

volumes_nfs:
  - {vol_name: "{{groups.oracle[0]}}_u01", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u02", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}
  - {vol_name: "{{groups.oracle[0]}}_u03", aggr_name: "aggr01_node01",
lif: "172.21.94.200", size: "25"}

#NFS LIFs IP address and netmask

nfs_lifs_details:
  - address: "172.21.94.200" #for node-1
    netmask: "255.255.255.0"
  - address: "172.21.94.201" #for node-2
    netmask: "255.255.255.0"

```

```

#NFS client match

client_match: "172.21.94.0/24"

#####
### 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 ###
#####

db_domain: "your.domain.com"

# Set initial password for all required Oracle passwords. Change them
after installation.

initial_pwd_all: "netapp123"

```

## Führen Sie das Playbook aus

Nach Abschluss der erforderlichen Umgebungsvoraussetzungen und Kopieren der Variablen in `vars.yml` Und `your_host.yml`, Sie sind jetzt bereit, die Playbooks zu implementieren.



<username> muss an Ihre Umgebung angepasst werden.

1. Führen Sie das ONTAP Playbook durch, indem Sie die richtigen Tags und den richtigen ONTAP Cluster-Benutzernamen eingeben. Geben Sie das Passwort für den ONTAP Cluster ein, und vsadmin, wenn Sie

dazu aufgefordert werden.

```
ansible-playbook -i hosts all_playbook.yml -u username -k -K -t  
ontap_config -e @vars/vars.yml
```

2. Führen Sie das Linux-Playbook aus, um den Linux-Teil der Bereitstellung auszuführen. Eingabe für admin ssh Passwort sowie sudo Passwort.

```
ansible-playbook -i hosts all_playbook.yml -u username -k -K -t  
linux_config -e @vars/vars.yml
```

3. Führen Sie das Oracle Playbook aus, um den Oracle Teil der Implementierung auszuführen. Eingabe für admin ssh Passwort sowie sudo Passwort.

```
ansible-playbook -i hosts all_playbook.yml -u username -k -K -t  
oracle_config -e @vars/vars.yml
```

### Implementieren Sie zusätzliche Datenbanken auf demselben Oracle Host

Der Oracle Teil des Playbook erstellt pro Ausführung eine einzelne Oracle-Container-Datenbank auf einem Oracle-Server. Gehen Sie wie folgt vor, um eine zusätzliche Container-Datenbank auf demselben Server zu erstellen:

1. Ändern der Variablen Host\_Vars.
  - a. Gehen Sie zurück zu Schritt 3 - Bearbeiten Sie den `host_name.yml` Datei unter `host_vars`.
  - b. Ändern Sie Oracle SID zu einer anderen Namenskonvention.
  - c. Ändern Sie den Listener-Port in eine andere Zahl.
  - d. Ändern Sie den EM Express-Port in eine andere Nummer, wenn Sie EM Express installiert haben.
  - e. Kopieren Sie die überarbeiteten Hostvariablen in die Oracle-Host-Variablendatei unter `host_vars`.
2. Führen Sie das Playbook mit dem `oracle_config` Tag wie oben in dargestellt [Führen Sie das Playbook aus](#).

### Oracle-Installation validieren

1. Melden Sie sich beim Oracle-Server als Oracle-Benutzer an und führen Sie die folgenden Befehle aus:

```
ps -ef | grep ora
```



Auf diese Weise werden die oracle-Prozesse aufgeführt, wenn die Installation wie erwartet abgeschlossen wurde und die oracle DB gestartet wurde

2. Melden Sie sich bei der Datenbank an, um die db-Konfigurationseinstellungen und die PDBs zu überprüfen, die mit den folgenden Befehlssätzen erstellt wurden.

```
[oracle@localhost ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 12:52:51 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL>

SQL> select name, log_mode from v$database;
NAME          LOG_MODE
-----
CDB2          ARCHIVELOG

SQL> show pdbs

      CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
          2 PDB$SEED                                READ ONLY  NO
          3 CDB2_PDB1                                READ WRITE NO
          4 CDB2_PDB2                                READ WRITE NO
          5 CDB2_PDB3                                READ WRITE NO

col svrname form a30
col dirname form a30
select svrname, dirname, nfsversion from v$dnfs_servers;

SQL> col svrname form a30
SQL> col dirname form a30
SQL> select svrname, dirname, nfsversion from v$dnfs_servers;

SVRNAME                                DIRNAME                                NFSVERSION
-----
172.21.126.200                        /rhelora03_u02                        NFSv3.0
172.21.126.200                        /rhelora03_u03                        NFSv3.0
172.21.126.200                        /rhelora03_u01                        NFSv3.0
```

Dies bestätigt, dass dNFS richtig funktioniert.

3. Stellen Sie über Listener eine Verbindung zur Datenbank her, um die Konfiguration des Oracle Listener mit dem folgenden Befehl zu überprüfen. Wechseln Sie zum entsprechenden Listener-Port und



```
[oracle@localhost ~]$ sqlplus
system@//localhost:1523/cdb2_pdb1.cie.netapp.com

SQL*Plus: Release 19.0.0.0.0 - Production on Thu May 6 13:19:57 2021
Version 19.8.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Enter password:
Last Successful login time: Wed May 05 2021 17:11:11 -04:00

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.8.0.0.0

SQL> show user
USER is "SYSTEM"
SQL> show con_name
CON_NAME
CDB2_PDB1
```

Dies bestätigt, dass Oracle Listener ordnungsgemäß funktioniert.

### Wo Hilfe benötigt wird?

Wenn Sie Hilfe mit dem Toolkit benötigen, nehmen Sie bitte an der Teil ["NetApp Solution Automation Community Support Slack Channel"](#) Und suchen Sie den Kanal zur Lösungsautomatisierung, um Ihre Fragen zu stellen oder zu fragen.

## Lösungsüberblick

### 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**

**<starke Klasse=„groß“>On-Prem</strong> <stark></strong>**

Umgebung	Anforderungen
<b>Ansible-Umgebung</b>	AWX/Tower
	Ansible v.2.10 und höher
	Python 3
	Python Libraries - netapp-lib - xmltodict - jmespath
<b>ONTAP</b>	ONTAP Version 9.8 +
	Zwei Datenaggregate
	NFS vlan und iffrp wurden erstellt
<b>Oracle Server</b>	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</strong>**

Umgebung	Anforderungen
<b>Ansible-Umgebung</b>	AWX/Tower
	Ansible v.2.10 und höher
	Python 3
	Python Libraries - netapp-lib - xmltodict - jmespath
<b>ONTAP</b>	ONTAP Version 9.8 +
	Zwei Datenaggregate
	NFS vlan und iffrp wurden erstellt
<b>Oracle Server</b>	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
<b>Cloud Manager/AWS</b>	AWS Zugriff/geheimer Schlüssel
	NetApp Cloud Manager Konto
	NetApp Cloud Manager – Token für die Aktualisierung



### <starke Klasse=„groß“>On-Prem</strong> <stark></strong>

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</strong>

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
<b>cvo_Setup</b>	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
<b>Ora_Replication_cg</b>	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
<b>Ora_Replication_log</b>	Schalten Sie das aktuelle Protokoll für jede Datenbank in /etc/oratab um
	Snapshot vom Oracle Log Volume erstellt
	Snapmirror Aktualisiert
<b>Ora_Erholung</b>	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

### 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](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 security group that allows access to on-prem
resources.
security_group: "sg-123123123"

# Your 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-Einrichtung

#### 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

### Zeitplanung für das Playbook „Binary and Database Replication“

#### 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 parallel-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

## TR-4794: Oracle Databases on NetApp EF-Series

Mitch Blackburn, Ebin Kadavy, NetApp

TR-4794 soll Storage-Administratoren und Datenbankadministratoren helfen, Oracle erfolgreich auf NetApp EF-Series Storage zu implementieren.

["TR-4794: Oracle Databases on NetApp EF-Series"](#)

## Copyright-Informationen

Copyright © 2024 NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtsinhabers in keiner Form und durch keine Mittel – weder grafische noch elektronische oder mechanische, einschließlich Fotokopieren, Aufnehmen oder Speichern in einem elektronischen Abrufsystem – auch nicht in Teilen, vervielfältigt werden.

Software, die von urheberrechtlich geschütztem NetApp Material abgeleitet wird, unterliegt der folgenden Lizenz und dem folgenden Haftungsausschluss:

DIE VORLIEGENDE SOFTWARE WIRD IN DER VORLIEGENDEN FORM VON NETAPP ZUR VERFÜGUNG GESTELLT, D. H. OHNE JEGLICHE EXPLIZITE ODER IMPLIZITE GEWÄHRLEISTUNG, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE STILLSCHWEIGENDE GEWÄHRLEISTUNG DER MARKTGÄNGIGKEIT UND EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, DIE HIERMIT AUSGESCHLOSSEN WERDEN. NETAPP ÜBERNIMMT KEINERLEI HAFTUNG FÜR DIREKTE, INDIREKTE, ZUFÄLLIGE, BESONDERE, BEISPIELHAFTE SCHÄDEN ODER FOLGESCHÄDEN (EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE BESCHAFFUNG VON ERSATZWAREN ODER -DIENSTLEISTUNGEN, NUTZUNGS-, DATEN- ODER GEWINNVERLUSTE ODER UNTERBRECHUNG DES GESCHÄFTSBETRIEBS), UNABHÄNGIG DAVON, WIE SIE VERURSACHT WURDEN UND AUF WELCHER HAFTUNGSTHEORIE SIE BERUHEN, OB AUS VERTRAGLICH FESTGELEGTER HAFTUNG, VERSCHULDENSUNABHÄNGIGER HAFTUNG ODER DELIKTSHAFTUNG (EINSCHLIESSLICH FAHRLÄSSIGKEIT ODER AUF ANDEREM WEGE), DIE IN IRGEND EINER WEISE AUS DER NUTZUNG DIESER SOFTWARE RESULTIEREN, SELBST WENN AUF DIE MÖGLICHKEIT DERARTIGER SCHÄDEN HINGEWIESEN WURDE.

NetApp behält sich das Recht vor, die hierin beschriebenen Produkte jederzeit und ohne Vorankündigung zu ändern. NetApp übernimmt keine Verantwortung oder Haftung, die sich aus der Verwendung der hier beschriebenen Produkte ergibt, es sei denn, NetApp hat dem ausdrücklich in schriftlicher Form zugestimmt. Die Verwendung oder der Erwerb dieses Produkts stellt keine Lizenzierung im Rahmen eines Patentrechts, Markenrechts oder eines anderen Rechts an geistigem Eigentum von NetApp dar.

Das in diesem Dokument beschriebene Produkt kann durch ein oder mehrere US-amerikanische Patente, ausländische Patente oder anhängige Patentanmeldungen geschützt sein.

ERLÄUTERUNG ZU „RESTRICTED RIGHTS“: Nutzung, Vervielfältigung oder Offenlegung durch die US-Regierung unterliegt den Einschränkungen gemäß Unterabschnitt (b)(3) der Klausel „Rights in Technical Data – Noncommercial Items“ in DFARS 252.227-7013 (Februar 2014) und FAR 52.227-19 (Dezember 2007).

Die hierin enthaltenen Daten beziehen sich auf ein kommerzielles Produkt und/oder einen kommerziellen Service (wie in FAR 2.101 definiert) und sind Eigentum von NetApp, Inc. Alle technischen Daten und die Computersoftware von NetApp, die unter diesem Vertrag bereitgestellt werden, sind gewerblicher Natur und wurden ausschließlich unter Verwendung privater Mittel entwickelt. Die US-Regierung besitzt eine nicht ausschließliche, nicht übertragbare, nicht unterlizenzierbare, weltweite, limitierte unwiderrufliche Lizenz zur Nutzung der Daten nur in Verbindung mit und zur Unterstützung des Vertrags der US-Regierung, unter dem die Daten bereitgestellt wurden. Sofern in den vorliegenden Bedingungen nicht anders angegeben, dürfen die Daten ohne vorherige schriftliche Genehmigung von NetApp, Inc. nicht verwendet, offengelegt, vervielfältigt, geändert, aufgeführt oder angezeigt werden. Die Lizenzrechte der US-Regierung für das US-Verteidigungsministerium sind auf die in DFARS-Klausel 252.227-7015(b) (Februar 2014) genannten Rechte beschränkt.

## Markeninformationen

NETAPP, das NETAPP Logo und die unter <http://www.netapp.com/TM> aufgeführten Marken sind Marken von NetApp, Inc. Andere Firmen und Produktnamen können Marken der jeweiligen Eigentümer sein.