



# **Implementieren Von Oracle Database**

## **NetApp Solutions**

NetApp  
April 26, 2024

This PDF was generated from <https://docs.netapp.com/de-de/netapp-solutionshttps://docs.netapp.com/us-en/ontap-apps-dbs/oracle/oracle-overview.html> on April 26, 2024. Always check docs.netapp.com for the latest.

# Inhalt

|  |   |
|--|---|
| Implementieren Von Oracle Database ..... | 1 |
| Lösungsüberblick .....                   | 1 |

# 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 `variables` Dateien abgeschlossen wurden, können Sie das Playbook in drei Phasen ausführen, indem Sie Tags für `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                                |

| Umgebung      | Anforderungen   |
|---------------|---|
| 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   |
|---------------|--|
| ontap_config  | Vorabprüfung der ONTAP-Umgebung  |
|               | Erstellung NFS-basierter SVM für Oracle  |
|               | Erstellung einer Exportrichtlinie  |
|               | Erstellung von Volumes für Oracle  |
|               | Erstellung von NFS LIFs  |
| linux_config  | 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   |
| oracle_config | 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.

- Navigieren Sie zu Ressourcen → Inventar → Hinzufügen, und klicken Sie auf Inventar hinzufügen.
- Geben Sie den Namen und die Organisationsdetails ein, und klicken Sie auf Speichern.
- Klicken Sie auf der Seite Inventar auf den erstellten Bestand.
- Wenn es Bestandsvariablen gibt, fügen Sie diese in das Feld Variablen ein.
- Navigieren Sie zum Untermenü Gruppen, und klicken Sie auf Hinzufügen.
- 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 „Typdetails“ 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 blank 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: "netapp123"

```

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

## 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.



## 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.