



Automatisierte Bereitstellung von Oracle 19c für ONTAP auf NFS

NetApp database solutions

NetApp
August 18, 2025

Inhalt

- Automatisierte Bereitstellung von Oracle 19c für ONTAP auf NFS 1
- Lösungsübersicht 1
- Automatisierte Bereitstellung von Oracle19c für ONTAP auf NFS 1
- Erste Schritte 2
- AWX/Turm 2
- CLI über den Ansible-Steuerungshost 2
- Anforderungen 2
- Automatisierungsdetails 3
- Standardparameter 4
- Bereitstellungsanweisungen 4
- Lizenz 4
- Schrittweises Bereitstellungsverfahren 5
- AWX/Tower-Bereitstellung Oracle 19c-Datenbank 5
- Schrittweises Bereitstellungsverfahren 14
- CLI-Bereitstellung Oracle 19c-Datenbank 14

Automatisierte Bereitstellung von Oracle 19c für ONTAP auf NFS

Lösungsübersicht

Auf dieser Seite wird die automatisierte Methode zum Bereitstellen von Oracle19c auf NetApp ONTAP -Speicher beschrieben.

Automatisierte Bereitstellung von Oracle19c für ONTAP auf NFS

Organisationen automatisieren ihre Umgebungen, um die Effizienz zu steigern, Bereitstellungen zu beschleunigen und den manuellen Aufwand zu reduzieren. Konfigurationsmanagement-Tools wie Ansible werden verwendet, um den Datenbankbetrieb in Unternehmen zu optimieren. In dieser Lösung zeigen wir, wie Sie mit Ansible die Bereitstellung und Konfiguration von Oracle 19c mit NetApp ONTAP automatisieren können. Indem Sie Speicheradministratoren, Systemadministratoren und Datenbankadministratoren die Möglichkeit geben, konsistent und schnell neuen Speicher bereitzustellen, Datenbankserver zu konfigurieren und Oracle 19c-Software zu installieren, erzielen Sie die folgenden Vorteile:

- Beseitigen Sie komplexe Designs und menschliche Fehler und implementieren Sie eine wiederholbare, konsistente Bereitstellung und bewährte Methoden.
- Reduzieren Sie die Zeit für die Bereitstellung von Speicher, die Konfiguration von DB-Hosts und die Oracle-Installation
- Steigern Sie die Produktivität von Datenbankadministratoren, System- und Speicheradministratoren
- Ermöglichen Sie die Skalierung von Speicher und Datenbanken mit Leichtigkeit

NetApp stellt Kunden validierte Ansible-Module und -Rollen zur Verfügung, um die Bereitstellung, Konfiguration und Lebenszyklusverwaltung Ihrer Oracle-Datenbankumgebung zu beschleunigen. Diese Lösung bietet Anweisungen und Ansible-Playbook-Code, um Ihnen zu helfen:

- Erstellen und Konfigurieren von ONTAP NFS-Speicher für Oracle Database
- Installieren Sie Oracle 19c auf RedHat Enterprise Linux 7/8 oder Oracle Linux 7/8
- Konfigurieren Sie Oracle 19c auf ONTAP NFS-Speicher

Weitere Einzelheiten oder den Einstieg finden Sie in den Übersichtsvideos unten.

AWX/Tower-Bereitstellungen

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

[AWX-Bereitstellung](#)

Teil 2: Variablen und Ausführen des Playbooks

[AWX Playbook-Ausführung](#)

CLI-Bereitstellung

Teil 1: Erste Schritte, Anforderungen, Automatisierungsdetails und Einrichtung des Ansible Control Hosts

[CLI-Bereitstellung](#)

Erste Schritte

Diese Lösung wurde für die Ausführung in einer AWX/Tower-Umgebung oder per CLI auf einem Ansible-Steuerungshost entwickelt.

AWX/Turm

Für AWX/Tower-Umgebungen werden Sie durch die Erstellung eines Inventars Ihres ONTAP Clustermanagements und Oracle-Servers (IPs und Hostnamen), das Erstellen von Anmeldeinformationen, das Konfigurieren eines Projekts, das den Ansible-Code von NetApp Automation Github abrufen, und der Jobvorlage geföhrt, die die Automatisierung startet.

1. Füllen Sie die für Ihre Umgebung spezifischen Variablen aus und kopieren Sie sie und fügen Sie sie in die Felder „Extra Vars“ in Ihrer Jobvorlage ein.
2. Nachdem die zusätzlichen Variablen zu Ihrer Jobvorlage hinzugefügt wurden, können Sie die Automatisierung starten.
3. Die Jobvorlage wird in drei Phasen ausgeführt, indem Tags für `ontap_config`, `linux_config` und `oracle_config` angegeben werden.

CLI über den Ansible-Steuerungshost

1. So konfigurieren Sie den Linux-Host, sodass er als Ansible-Steuerungshost verwendet werden kann"[Klicken Sie hier für detaillierte Anweisungen](#)"
2. Nachdem der Ansible-Steuerungshost konfiguriert ist, können Sie das Ansible Automation-Repository per Git klonen.
3. Bearbeiten Sie die Hosts-Datei mit den IPs und/oder Hostnamen der Verwaltungs-IPs Ihres ONTAP Clusters und der Verwaltungs-IPs des Oracle-Servers.
4. Füllen Sie die für Ihre Umgebung spezifischen Variablen aus und kopieren Sie sie in die `vars.yml` Datei.
5. Jeder Oracle-Host verfügt über eine durch seinen Hostnamen identifizierte Variablendatei, die hostspezifische Variablen enthält.
6. Nachdem alle Variablendateien fertiggestellt wurden, können Sie das Playbook in drei Phasen ausführen, indem Sie Tags für `ontap_config`, `linux_config`, Und `oracle_config`.

Anforderungen

Umfeld	Anforderungen
Ansible-Umgebung	AWX/Tower oder Linux-Host als Ansible-Steuerungshost
	Ansible v.2.10 und höher
	Python 3
	Python-Bibliotheken – netapp-lib – xmltodict – jmespath

Umfeld	Anforderungen
* ONTAP*	ONTAP Version 9.3 – 9.7
	Zwei Datenaggregate
	NFS-VLAN und IFGRP erstellt
Oracle-Server	RHEL 7/8
	Oracle Linux 7/8
	Netzwerkschnittstellen für NFS, öffentliche und optionale Verwaltung
	Oracle-Installationsdateien auf Oracle-Servern

Automatisierungsdetails

Diese automatisierte Bereitstellung wird mit einem einzigen Ansible-Playbook entwickelt, das aus drei separaten Rollen besteht. Die Rollen gelten für ONTAP, Linux- und Oracle-Konfigurationen. In der folgenden Tabelle wird beschrieben, welche Aufgaben automatisiert werden.

Rolle	Aufgaben
ontap_config	Vorabprüfung der ONTAP -Umgebung
	Erstellung einer NFS-basierten SVM für Oracle
	Erstellung einer Exportrichtlinie
	Erstellen von Volumes für Oracle
	Erstellung von NFS-LIFs
linux_config	Erstellen Sie Mount-Punkte und mounten Sie NFS-Volumes
	NFS-Mounts überprüfen
	Betriebssystemspezifische Konfiguration
	Erstellen von Oracle-Verzeichnissen
	Konfigurieren Sie Hugepages
	Deaktivieren Sie SELinux und den Firewall-Daemon
	Aktivieren und starten Sie den Chronyd-Dienst
	Erhöhen Sie das harte Limit für Dateideskriptoren
	Erstellen Sie eine pam.d-Sitzungsdatei

Rolle	Aufgaben
oracle_config	Installation der Oracle-Software
	Oracle-Listener erstellen
	Erstellen von Oracle-Datenbanken
	Oracle-Umgebungsconfiguration
	PDB-Status speichern
	Aktivieren Sie den Instanzarchivmodus
	DNFS-Client aktivieren
	Aktivieren Sie das automatische Starten und Herunterfahren der Datenbank zwischen Betriebssystemneustarts

Standardparameter

Um die Automatisierung zu vereinfachen, haben wir viele erforderliche Oracle-Bereitstellungsparameter mit Standardwerten voreingestellt. Für die meisten Bereitstellungen ist es im Allgemeinen nicht erforderlich, die Standardparameter zu ändern. Ein erfahrener Benutzer kann mit Vorsicht Änderungen an den Standardparametern vornehmen. Die Standardparameter befinden sich in jedem Rollenordner im Verzeichnis „Defaults“.

Bereitstellungsanweisungen

Laden Sie vor dem Start die folgenden Oracle-Installations- und Patchdateien herunter und legen Sie sie im `/tmp/archive` Verzeichnis mit Lese-, Schreib- und Ausführungszugriff für alle Benutzer auf jedem bereitzustellenden DB-Server. Die Automatisierungsaufgaben suchen in diesem bestimmten Verzeichnis nach den benannten Installationsdateien für die Oracle-Installation und -Konfiguration.

```
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 im Github-Repository lesen. Indem Sie auf die Inhalte dieses Repositories zugreifen, sie herunterladen, installieren oder verwenden, stimmen Sie den Bedingungen der Lizenz zu. ["hier,"](#) .

Beachten Sie, dass hinsichtlich der Erstellung und/oder Weitergabe abgeleiteter Werke aus den Inhalten dieses Repositories bestimmte Einschränkungen gelten. Bitte lesen Sie unbedingt die Bedingungen der ["Lizenz"](#) bevor Sie den Inhalt verwenden. Wenn Sie nicht allen Bedingungen zustimmen, dürfen Sie nicht auf die Inhalte in diesem Repository zugreifen, sie nicht herunterladen oder verwenden.

Wenn Sie fertig sind, klicken Sie auf ["Hier finden Sie detaillierte AWX/Tower-Bereitstellungsverfahren"](#) oder ["hier für die CLI-Bereitstellung"](#) .

Schrittweises Bereitstellungsverfahren

Auf dieser Seite wird die automatisierte Methode zum Bereitstellen von Oracle19c auf NetApp ONTAP -Speicher beschrieben.

AWX/Tower-Bereitstellung 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 in AWX/Ansible Tower beschrieben, die die Umgebung für die Nutzung automatisierter NetApp -Lösungen vorbereiten.

1. Konfigurieren Sie das Inventar.
 - a. Navigieren Sie zu Ressourcen → Bestände → Hinzufügen und klicken Sie auf Bestand hinzufügen.
 - b. Geben Sie den Namen und die Organisationsdetails ein und klicken Sie auf „Speichern“.
 - c. Klicken Sie auf der Seite „Inventare“ auf das erstellte Inventar.
 - d. Wenn Inventarvariablen vorhanden sind, fügen Sie sie in das Variablenfeld 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 (sofern vorhanden) ein 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 -Clusterverwaltungs-IP ein, fügen Sie die Hostvariablen (falls vorhanden) ein und klicken Sie auf „Speichern“.
 - j. Dieser Vorgang muss für die Verwaltungs-IP/den Verwaltungs-Hostnamen der Oracle-Gruppe und des/der Oracle-Hosts wiederholt werden.
2. Erstellen Sie Anmeldeinformationstypen. Bei Lösungen mit ONTAP müssen Sie den Anmeldeinformationstyp so konfigurieren, dass er mit den Benutzernamen- und Kennworteingaben übereinstimmt.
 - a. Navigieren Sie zu „Administration“ → „Anmeldeinformationstypen“ und klicken Sie auf „Hinzufügen“.
 - b. Geben Sie den Namen und die Beschreibung ein.
 - 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 den folgenden Inhalt in die Injector-Konfiguration ein:

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

1. Konfigurieren Sie die Anmeldeinformationen.

- a. Navigieren Sie zu Ressourcen → Anmeldeinformationen und klicken Sie auf Hinzufügen.
- b. Geben Sie den Namen und die Organisationsdetails für ONTAP ein.
- c. Wählen Sie den benutzerdefinierten Anmeldeinformationstyp aus, den Sie für ONTAP erstellt haben.
- d. Geben Sie unter „Typdetails“ den Benutzernamen, das Kennwort und das vsadmin_password ein.
- e. Klicken Sie auf „Zurück zu Anmeldeinformationen“ und dann auf „Hinzufügen“.
- f. Geben Sie den Namen und die Organisationsdetails für Oracle ein.
- g. Wählen Sie den Anmeldeinformationstyp des Computers aus.
- h. Geben Sie unter „Typdetails“ den Benutzernamen und das Kennwort für die Oracle-Hosts ein.
- i. Wählen Sie die richtige Methode zur Rechteerweiterung aus und geben Sie den Benutzernamen und das Kennwort ein.

2. Erstellen eines Projekts

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 „Anmeldeinformationstyp der Quellcodeverwaltung“ die Option „Git“ aus.
 - c. eingeben `https://github.com/NetApp-Automation/na_oracle19c_deploy.git` als Quellcodeverwaltungs-URL.
 - d. Klicken Sie auf Speichern.
 - e. Das Projekt muss möglicherweise gelegentlich synchronisiert werden, wenn sich der Quellcode ändert.

3. Konfigurieren Sie Oracle host_vars

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

1. Geben Sie Ihre umgebungsspezifischen Parameter in das folgende eingebettete Oracle-Hostvariablen- oder Host_vars-Formular ein.



Die blauen Elemente müssen geändert werden, damit sie zu Ihrer Umgebung passen.

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 aus.
2. Klicken Sie nach Abschluss der Variableneingabe 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, gehen Sie zu Ressourcen → Hosts und wählen Sie die Oracle-Serverkonfigurationsseite aus und öffnen Sie sie.
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. Konfigurieren globaler Variablen

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 geändert werden, damit sie zu Ihrer Umgebung passen.

```
#####  
##### 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. Füllen Sie alle Variablen in den blauen Feldern aus.
2. Klicken Sie nach Abschluss der Variableneingabe auf die Schaltfläche „Kopieren“ im Formular, um alle an AWX oder Tower zu übertragenden Variablen in die folgende Jobvorlage zu kopieren.

5. Konfigurieren und starten Sie die Jobvorlage.

1. Erstellen Sie die Jobvorlage.
 - a. Navigieren Sie zu Ressourcen → Vorlagen → Hinzufügen und klicken Sie auf Jobvorlage hinzufügen.
 - b. Geben Sie den Namen und die Beschreibung ein
 - c. Wählen Sie den Jobtyp aus. „Ausführen“ konfiguriert das System basierend auf einem Playbook und „Überprüfen“ führt einen Probelauf eines Playbooks durch, ohne das System tatsächlich zu konfigurieren.
 - d. Wählen Sie das entsprechende Inventar, Projekt, Playbook und die Anmeldeinformationen für das Playbook aus.
 - e. Wählen Sie all_playbook.yml als auszuführendes Standard-Playbook aus.
 - f. Fügen Sie die aus Schritt 4 kopierten globalen Variablen in das Feld „Vorlagenvariablen“ unter der Registerkarte „YAML“ ein.
 - g. Aktivieren Sie im Feld „Job-Tags“ das Kontrollkästchen „Beim Start auffordern“.
 - 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 zur Eingabe von Job-Tags aufgefordert werden, geben Sie requirements_config ein. Möglicherweise müssen Sie auf die Zeile „Job-Tag erstellen“ unter „requirements_config“ klicken, um das Job-Tag einzugeben.



requirements_config stellt sicher, dass Sie über die richtigen Bibliotheken zum Ausführen der anderen Rollen verfügen.

1. Klicken Sie auf „Weiter“ und dann auf „Starten“, um den Auftrag zu starten.
2. Klicken Sie auf „Anzeigen“ → „Jobs“, um die Jobausgabe und den Fortschritt zu überwachen.
3. Wenn Sie beim Start zur Eingabe von Job-Tags aufgefordert werden, geben Sie ontap_config ein.

Möglicherweise müssen Sie direkt unter `ontap_config` auf die Zeile „Job-Tag erstellen“ klicken, um das Job-Tag einzugeben.

4. Klicken Sie auf „Weiter“ und dann auf „Starten“, um den Auftrag zu starten.
5. Klicken Sie auf Ansicht → Jobs, um die Jobausgabe und den Fortschritt zu überwachen
6. Nachdem die Rolle `ontap_config` abgeschlossen ist, führen Sie 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 zur Eingabe des Job-Tag-Typs in `linux_config` aufgefordert werden, 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 „Starten“, um den Auftrag zu starten.
11. Wählen Sie „Ansicht“ → „Jobs“, um die Jobausgabe und den Fortschritt zu überwachen.
12. Nachdem die Rolle „`linux_config`“ abgeschlossen ist, führen Sie 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 zur Eingabe von Job-Tags aufgefordert werden, geben Sie `oracle_config` ein. Möglicherweise müssen Sie die Zeile „Job-Tag erstellen“ direkt unter „`oracle_config`“ auswählen, um das Job-Tag einzugeben.
16. Klicken Sie auf „Weiter“ und dann auf „Starten“, um den Auftrag zu starten.
17. Wählen Sie „Ansicht“ → „Jobs“, um die Jobausgabe und den Fortschritt zu überwachen.

6. Bereitstellen einer zusätzlichen Datenbank auf demselben Oracle-Host

Der Oracle-Teil des Playbooks erstellt pro Ausführung eine einzelne Oracle-Containerdatenbank auf einem Oracle-Server. Führen Sie die folgenden Schritte aus, um zusätzliche Containerdatenbanken auf demselben Server zu erstellen.

1. Überarbeiten Sie die `host_vars`-Variablen.
 - a. Gehen Sie zurück zu Schritt 2 – Konfigurieren Sie Oracle `host_vars`.
 - b. Ändern Sie die Oracle-SID in eine andere Namenszeichenfolge.
 - c. Ändern Sie den Listener-Port auf eine andere Nummer.
 - d. Ändern Sie den EM Express-Port in eine andere Nummer, wenn Sie EM Express installieren.
 - e. Kopieren Sie die überarbeiteten Hostvariablen und fügen Sie sie in das Feld „Oracle-Hostvariablen“ auf der Registerkarte „Hostkonfigurationsdetails“ ein.
2. Starten Sie die Bereitstellungsauftragsvorlage nur mit dem Tag „`oracle_config`“.
3. Melden Sie sich als Oracle-Benutzer beim Oracle-Server an und führen Sie die folgenden Befehle aus:

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



Hier werden Oracle-Prozesse aufgelistet, wenn die Installation wie erwartet abgeschlossen und Oracle DB gestartet wurde.

4. Melden Sie sich bei der Datenbank an, um die Datenbankkonfigurationseinstellungen und die mit den folgenden Befehlsätzen erstellten PDBs zu überprüfen.

```
[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 ordnungsgemäß funktioniert.

5. Stellen Sie über den Listener eine Verbindung zur Datenbank her, um die Oracle-Listener-Konfiguration 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 der Oracle-Listener ordnungsgemäß funktioniert.

Wo bekomme ich Hilfe?

Wenn Sie Hilfe mit dem Toolkit benötigen, treten Sie bitte dem bei "[Slack-Kanal für den Community-Support von NetApp Solution Automation](#)" und suchen Sie nach dem Kanal zur Lösungsautomatisierung, um Ihre Fragen oder Anfragen zu posten.

Schrittweises Bereitstellungsverfahren

Dieses Dokument beschreibt die Bereitstellung von Oracle 19c mithilfe der Automatisierungs-Befehlszeilenschnittstelle (CLI).

CLI-Bereitstellung Oracle 19c-Datenbank

In diesem Abschnitt werden die erforderlichen Schritte zum Vorbereiten und Bereitstellen der Oracle19c-Datenbank mit der CLI beschrieben. Stellen Sie sicher, dass Sie die "[Abschnitt „Erste Schritte und Anforderungen“](#)" und Ihre Umgebung entsprechend vorbereitet.

Laden Sie das Oracle19c-Repository herunter

1. Führen Sie von Ihrem Ansible-Controller aus den folgenden Befehl aus:

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

2. Wechseln Sie nach dem Herunterladen des Repositorys in das Verzeichnis `na_oracle19c_deploy` `<cd na_oracle19c_deploy>`.

Bearbeiten Sie die Hosts-Datei

Führen Sie vor der Bereitstellung die folgenden Schritte aus:

1. Bearbeiten Sie das Verzeichnis `na_oracle19c_deploy` Ihrer Hosts-Datei.
2. Ändern Sie unter `[ontap]` die IP-Adresse in Ihre Cluster-Verwaltungs-IP.
3. Fügen Sie unter der Gruppe `[oracle]` die Namen der Oracle-Hosts hinzu. Der Hostname muss entweder über DNS oder die Hosts-Datei in seine IP-Adresse aufgelöst werden oder im Host angegeben werden.
4. Nachdem Sie diese Schritte abgeschlossen haben, speichern Sie alle Änderungen.

Das folgende Beispiel zeigt eine Hostdatei:

```
#ONTAP Host

[ontap]

"10.61.184.183"

#Oracle hosts

[oracle]

"rtpora01"

"rtpora02"
```

Dieses Beispiel führt das Playbook aus und stellt Oracle 19c gleichzeitig auf zwei Oracle-DB-Servern bereit. Sie können auch mit nur einem DB-Server testen. In diesem Fall müssen Sie nur eine Hostvariablendatei konfigurieren.



Das Playbook wird unabhängig von der Anzahl der bereitgestellten Oracle-Hosts und -Datenbanken auf die gleiche Weise ausgeführt.

Bearbeiten Sie die Datei `host_name.yml` unter `host_vars`

Jeder Oracle-Host verfügt über eine Hostvariablendatei, die durch seinen Hostnamen identifiziert wird und hostspezifische Variablen enthält. Sie können für Ihren Host einen beliebigen Namen angeben. Bearbeiten und kopieren Sie die `host_vars` aus dem Abschnitt Host VARS Config und fügen Sie es in Ihr gewünschtes `host_name.yml` Datei.



Die blauen Elemente müssen geändert werden, damit sie zu Ihrer Umgebung passen.

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 Oracle-Bereitstellung.

1. Bearbeiten und kopieren Sie die Variablen aus dem Abschnitt VARS und fügen Sie diese Variablen in Ihre 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"

```

Ausführen des Playbooks

Nachdem Sie die erforderlichen Umgebungsvoraussetzungen erfüllt und die Variablen in `vars.yml` Und `your_host.yml` , Sie sind jetzt bereit, die Playbooks bereitzustellen.



<Benutzername> muss geändert werden, damit es Ihrer Umgebung entspricht.

1. Führen Sie das ONTAP -Playbook aus, indem Sie die richtigen Tags und den ONTAP Cluster-Benutzernamen übergeben. Geben Sie das Kennwort für den ONTAP Cluster und vsadmin ein, 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 das Admin-SSH-Passwort sowie das 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 Bereitstellung auszuführen. Eingabe für das Admin-SSH-Passwort sowie das Sudo-Passwort.

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

Bereitstellen einer zusätzlichen Datenbank auf demselben Oracle-Host

Der Oracle-Teil des Playbooks erstellt pro Ausführung eine einzelne Oracle-Containerdatenbank auf einem Oracle-Server. Führen Sie die folgenden Schritte aus, um eine zusätzliche Containerdatenbank auf demselben Server zu erstellen:

1. Überarbeiten Sie die `host_vars`-Variablen.
 - a. Gehen Sie zurück zu Schritt 3 - Bearbeiten Sie die `host_name.yml` Datei unter `host_vars`.
 - b. Ändern Sie die Oracle-SID in eine andere Namenszeichenfolge.
 - c. Ändern Sie den Listener-Port auf eine andere Nummer.
 - d. Ändern Sie den EM Express-Port in eine andere Nummer, wenn Sie EM Express installiert haben.
 - e. Kopieren Sie die überarbeiteten Hostvariablen und fügen Sie sie in die Oracle-Hostvariablendatei unter `host_vars`.
2. Führen Sie das Playbook mit dem `oracle_config` Tag wie oben gezeigt in [Ausführen des Playbooks](#).

Überprüfen der Oracle-Installation

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

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



Hier werden Oracle-Prozesse aufgelistet, wenn die Installation wie erwartet abgeschlossen und Oracle DB gestartet wurde.

2. Melden Sie sich bei der Datenbank an, um die Datenbankkonfigurationseinstellungen und die mit den folgenden Befehlssätzen erstellten PDBs zu überprüfen.

```

[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 ordnungsgemäß funktioniert.

3. Stellen Sie über den Listener eine Verbindung zur Datenbank her, um die Oracle-Listener-Konfiguration 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 der Oracle-Listener ordnungsgemäß funktioniert.

Wo bekomme ich Hilfe?

Wenn Sie Hilfe mit dem Toolkit benötigen, treten Sie bitte dem bei ["Slack-Kanal für den Community-Support von NetApp Solution Automation"](#) und suchen Sie nach dem Kanal zur Lösungsautomatisierung, um Ihre Fragen oder Anfragen zu posten.

Copyright-Informationen

Copyright © 2025 NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtinhabers 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, BEISPIELHAFT 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.