



Implementación automatizada de Oracle 19c para ONTAP en NFS

NetApp database solutions

NetApp
August 18, 2025

Tabla de contenidos

- Implementación automatizada de Oracle 19c para ONTAP en NFS 1
 - Descripción general de la solución 1
 - Implementación automatizada de Oracle19c para ONTAP en NFS 1
 - Empezando 2
 - AWX/Torre 2
 - CLI a través del host de control de Ansible 2
 - Requisitos 2
 - Detalles de la automatización 3
 - Parámetros predeterminados 4
 - Instrucciones de implementación 4
 - Licencia 4
- Procedimiento de implementación paso a paso 5
 - Implementación de AWX/Tower Base de datos Oracle 19c 5
- Procedimiento de implementación paso a paso 14
 - Implementación de CLI de la base de datos Oracle 19c 14

Implementación automatizada de Oracle 19c para ONTAP en NFS

Descripción general de la solución

Esta página describe el método automatizado para implementar Oracle19c en el almacenamiento NetApp ONTAP .

Implementación automatizada de Oracle19c para ONTAP en NFS

Las organizaciones están automatizando sus entornos para ganar eficiencia, acelerar las implementaciones y reducir el esfuerzo manual. Se están utilizando herramientas de gestión de configuración como Ansible para optimizar las operaciones de bases de datos empresariales. En esta solución, demostramos cómo puede utilizar Ansible para automatizar el aprovisionamiento y la configuración de Oracle 19c con NetApp ONTAP. Al permitir que los administradores de almacenamiento, administradores de sistemas y DBA implementen de manera consistente y rápida nuevo almacenamiento, configuren servidores de bases de datos e instalen el software Oracle 19c, obtiene los siguientes beneficios:

- Elimine las complejidades de diseño y los errores humanos, e implemente una implementación consistente y repetible y las mejores prácticas.
- Reducir el tiempo de aprovisionamiento de almacenamiento, configuración de hosts de bases de datos e instalación de Oracle
- Aumente la productividad de los administradores de bases de datos, sistemas y almacenamiento
- Habilite el escalado del almacenamiento y las bases de datos con facilidad

NetApp proporciona a los clientes módulos y roles de Ansible validados para acelerar la implementación, la configuración y la gestión del ciclo de vida de su entorno de base de datos Oracle. Esta solución proporciona instrucciones y código de libro de estrategias de Ansible para ayudarle a:

- Crear y configurar el almacenamiento NFS de ONTAP para Oracle Database
- Instalar Oracle 19c en RedHat Enterprise Linux 7/8 u Oracle Linux 7/8
- Configurar Oracle 19c en el almacenamiento NFS de ONTAP

Para obtener más detalles o comenzar, consulte los videos de descripción general a continuación.

Implementaciones de AWX/Torre

Parte 1: Primeros pasos, requisitos, detalles de automatización y configuración inicial de AWX/Tower

[Implementación de AWX](#)

Parte 2: Variables y ejecución del libro de jugadas

[AWX Playbook Run](#)

Implementación de CLI

Parte 1: Primeros pasos, requisitos, detalles de automatización y configuración del host de control de Ansible

[Implementación de CLI](#)

Empezando

Esta solución ha sido diseñada para ejecutarse en un entorno AWX/Tower o mediante CLI en un host de control Ansible.

AWX/Torre

Para los entornos AWX/Tower, se le guiará a través de la creación de un inventario de su administración de clústeres ONTAP y del servidor Oracle (IP y nombres de host), la creación de credenciales, la configuración de un proyecto que extrae el código Ansible de NetApp Automation Github y la plantilla de trabajo que inicia la automatización.

1. Complete las variables específicas de su entorno y cópielas y péguelas en los campos Variables adicionales en su plantilla de trabajo.
2. Una vez que se hayan agregado las variables adicionales a su plantilla de trabajo, puede iniciar la automatización.
3. La plantilla de trabajo se ejecuta en tres fases especificando etiquetas para `ontap_config`, `linux_config` y `oracle_config`.

CLI a través del host de control de Ansible

1. Para configurar el host Linux para que pueda usarse como host de control de Ansible "[Haga clic aquí para obtener instrucciones detalladas](#)".
2. Una vez configurado el host de control de Ansible, puedes clonar el repositorio de automatización de Ansible.
3. Edite el archivo de hosts con las IP o los nombres de host de su administración de clúster ONTAP y las IP de administración del servidor Oracle.
4. Complete las variables específicas de su entorno y cópielas y péguelas en el `vars.yml` archivo.
5. Cada host de Oracle tiene un archivo de variables identificado por su nombre de host que contiene variables específicas del host.
6. Una vez completados todos los archivos de variables, puede ejecutar el libro de estrategias en tres fases especificando etiquetas para `ontap_config`, `linux_config`, y `oracle_config`.

Requisitos

Ambiente	Requisitos
Entorno Ansible	AWX/Tower o host Linux para ser el host de control de Ansible
	Ansible v.2.10 y superior
	Python 3
	Bibliotecas de Python: <code>netapp-lib</code> , <code>xmltodict</code> y <code>jmespath</code>

Ambiente	Requisitos
* ONTAP*	Versión 9.3-9.7 de ONTAP
	Dos agregados de datos
	VLAN NFS y grupo IF creados
Servidores Oracle	RHEL 7/8
	Oracle Linux 7/8
	Interfaces de red para NFS, públicas y de gestión opcional
	Archivos de instalación de Oracle en servidores Oracle

Detalles de la automatización

Esta implementación automatizada está diseñada con un único manual de Ansible que consta de tres roles separados. Los roles son para configuraciones de ONTAP, Linux y Oracle. En la siguiente tabla se describen qué tareas se están automatizando.

Role	Tareas
ontap_config	Comprobación previa del entorno ONTAP
	Creación de SVM basada en NFS para Oracle
	Creación de política de exportación
	Creación de volúmenes para Oracle
	Creación de LIF de NFS
configuración de linux	Crear puntos de montaje y montar volúmenes NFS
	Verificar montajes NFS
	Configuración específica del sistema operativo
	Crear directorios de Oracle
	Configurar páginas enormes
	Deshabilitar SELinux y el demonio de firewall
	Habilitar e iniciar el servicio chronyd
	Aumentar el límite máximo del descriptor de archivo
	Crear archivo de sesión pam.d

Role	Tareas
configuración oracle	Instalación del software de Oracle
	Crear un escucha de Oracle
	Crear bases de datos Oracle
	Configuración del entorno de Oracle
	Guardar el estado del PDB
	Habilitar el modo de archivo de instancias
	Habilitar el cliente DNFS
	Habilitar el inicio y apagado automático de la base de datos entre reinicios del sistema operativo

Parámetros predeterminados

Para simplificar la automatización, hemos preestablecido muchos parámetros de implementación de Oracle requeridos con valores predeterminados. Por lo general, no es necesario cambiar los parámetros predeterminados para la mayoría de las implementaciones. Un usuario más avanzado puede realizar cambios en los parámetros predeterminados con precaución. Los parámetros predeterminados se encuentran en cada carpeta de función, en el directorio de valores predeterminados.

Instrucciones de implementación

Antes de comenzar, descargue los siguientes archivos de instalación y parches de Oracle y colóquelos en el directorio `/tmp/archive` Directorio con acceso de lectura, escritura y ejecución para todos los usuarios en cada servidor de base de datos que se implementará. Las tareas de automatización buscan los archivos de instalación nombrados en ese directorio particular para la instalación y configuración de 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
```

Licencia

Debes leer la información de la licencia tal como se indica en el repositorio de Github. Al acceder, descargar, instalar o utilizar el contenido de este repositorio, usted acepta los términos de la licencia establecida ["aquí"](#) .

Tenga en cuenta que existen ciertas restricciones en torno a la producción y/o intercambio de cualquier trabajo derivado del contenido de este repositorio. Asegúrese de leer los términos del ["Licencia"](#) antes de utilizar el contenido. Si no está de acuerdo con todos los términos, no acceda, descargue ni utilice el contenido de este repositorio.

Cuando esté listo, haga clic en ["Aquí para conocer los procedimientos detallados de implementación de AWX/Tower"](#) o ["aquí para la implementación de CLI"](#) .

Procedimiento de implementación paso a paso

Esta página describe el método automatizado para implementar Oracle19c en el almacenamiento NetApp ONTAP .

Implementación de AWX/Tower Base de datos Oracle 19c

1. Cree el inventario, el grupo, los hosts y las credenciales para su entorno

Esta sección describe la configuración del inventario, los grupos, los hosts y las credenciales de acceso en AWX/Ansible Tower que preparan el entorno para consumir soluciones automatizadas de NetApp .

1. Configurar el inventario.
 - a. Vaya a Recursos → Inventarios → Agregar y haga clic en Agregar inventario.
 - b. Proporcione el nombre y los detalles de la organización y haga clic en Guardar.
 - c. En la página Inventarios, haga clic en el inventario creado.
 - d. Si hay variables de inventario, péguelas en el campo de variables.
 - e. Vaya al submenú Grupos y haga clic en Agregar.
 - f. Proporcione el nombre del grupo para ONTAP, pegue las variables del grupo (si hay alguna) y haga clic en Guardar.
 - g. Repita el proceso para otro grupo de Oracle.
 - h. Seleccione el grupo ONTAP creado, vaya al submenú Hosts y haga clic en Agregar nuevo host.
 - i. Proporcione la dirección IP de la IP de administración del clúster ONTAP , pegue las variables del host (si hay alguna) y haga clic en Guardar.
 - j. Este proceso debe repetirse para el grupo Oracle y el nombre de host/IP de administración de host(s) Oracle.
2. Crear tipos de credenciales. Para las soluciones que involucran ONTAP, debe configurar el tipo de credencial para que coincida con las entradas de nombre de usuario y contraseña.
 - a. Vaya a Administración → Tipos de credenciales y haga clic en Agregar.
 - b. Proporcione el nombre y la descripción.
 - c. Pegue el siguiente contenido en Configuración de entrada:

```
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. Pegue el siguiente contenido en la configuración del inyector:

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

1. Configurar las credenciales.

- a. Vaya a Recursos → Credenciales y haga clic en Agregar.
- b. Ingrese el nombre y los detalles de la organización para ONTAP.
- c. Seleccione el tipo de credencial personalizado que creó para ONTAP.
- d. En Detalles de tipo, ingrese el nombre de usuario, la contraseña y vsadmin_password.
- e. Haga clic en Volver a Credencial y haga clic en Agregar.
- f. Ingrese el nombre y los detalles de la organización de Oracle.
- g. Seleccione el tipo de credencial de la máquina.
- h. En Detalles de tipo, ingrese el nombre de usuario y la contraseña para los hosts de Oracle.
- i. Seleccione el método de escalada de privilegios correcto e ingrese el nombre de usuario y la contraseña.

2. Crear un proyecto

1. Vaya a Recursos → Proyectos y haga clic en Agregar.
 - a. Introduzca el nombre y los detalles de la organización.
 - b. Seleccione Git en el campo Tipo de credencial de control de fuente.
 - c. ingresar `https://github.com/NetApp-Automation/na_oracle19c_deploy.git` como la URL de control de origen.
 - d. Haga clic en Guardar.
 - e. Es posible que el proyecto necesite sincronizarse ocasionalmente cuando cambie el código fuente.

3. Configurar host_vars de Oracle

Las variables definidas en esta sección se aplican a cada servidor y base de datos Oracle individual.

1. Ingrese los parámetros específicos de su entorno en el siguiente formato host_vars o variables de host de Oracle integradas.



Los elementos en azul deben cambiarse para que coincidan con su entorno.

Configuración de VARS del host

```
#####
##### 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. Complete todas las variables en los campos azules.
2. Después de completar el ingreso de variables, haga clic en el botón Copiar en el formulario para copiar todas las variables que se transferirán a AWX o Tower.
3. Regrese a AWX o Tower y vaya a Recursos → Hosts, y seleccione y abra la página de configuración del servidor Oracle.
4. En la pestaña Detalles, haga clic en editar y pegue las variables copiadas del paso 1 en el campo Variables debajo de la pestaña YAML.
5. Haga clic en Guardar.
6. Repita este proceso para cualquier servidor Oracle adicional en el sistema.

4. Configurar variables globales

Las variables definidas en esta sección se aplican a todos los hosts, bases de datos y el clúster ONTAP de Oracle.

1. Ingrese los parámetros específicos de su entorno en el siguiente formato de variables o variables globales integradas.



Los elementos en azul deben cambiarse para que coincidan con su entorno.

```
#####
##### 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. Complete todas las variables en los campos azules.
2. Después de completar el ingreso de variables, haga clic en el botón Copiar en el formulario para copiar todas las variables que se transferirán a AWX o Tower en la siguiente plantilla de trabajo.

5. Configurar y ejecutar la plantilla de trabajo.

1. Crear la plantilla de trabajo.
 - a. Vaya a Recursos → Plantillas → Agregar y haga clic en Agregar plantilla de trabajo.
 - b. Introduzca el nombre y la descripción
 - c. Seleccione el tipo de trabajo; Ejecutar configura el sistema basándose en un libro de estrategias y Verificar realiza una ejecución en seco de un libro de estrategias sin configurar realmente el sistema.
 - d. Seleccione el inventario, el proyecto, el libro de jugadas y las credenciales correspondientes para el libro de jugadas.
 - e. Seleccione all_playbook.yml como el libro de jugadas predeterminado que se ejecutará.
 - f. Pegue las variables globales copiadas del paso 4 en el campo Variables de plantilla debajo de la pestaña YAML.
 - g. Marque la casilla Preguntar al iniciar en el campo Etiquetas de trabajo.
 - h. Haga clic en Guardar.
2. Iniciar la plantilla de trabajo.
 - a. Vaya a Recursos → Plantillas.
 - b. Haga clic en la plantilla deseada y luego haga clic en Iniciar.
 - c. Cuando se le solicite durante el inicio etiquetas de trabajo, escriba requirements_config. Es posible que necesite hacer clic en la línea Crear etiqueta de trabajo debajo de requirements_config para ingresar la etiqueta de trabajo.



requirements_config garantiza que tenga las bibliotecas correctas para ejecutar los otros roles.

1. Haga clic en Siguiente y luego en Iniciar para comenzar el trabajo.
2. Haga clic en Ver → Trabajos para supervisar el progreso y los resultados del trabajo.
3. Cuando se le solicite durante el inicio etiquetas de trabajo, escriba ontap_config. Es posible que necesites hacer clic en la línea Crear "Etiqueta de trabajo" justo debajo de ontap_config para ingresar la etiqueta de trabajo.
4. Haga clic en Siguiente y luego en Iniciar para comenzar el trabajo.

5. Haga clic en Ver → Trabajos para supervisar el resultado y el progreso del trabajo.
6. Una vez completada la función ontap_config, ejecute nuevamente el proceso para linux_config.
7. Vaya a Recursos → Plantillas.
8. Seleccione la plantilla deseada y luego haga clic en Iniciar.
9. Cuando se le solicite durante el inicio el tipo de etiquetas de trabajo en linux_config, es posible que deba seleccionar la línea Crear "etiqueta de trabajo" justo debajo de linux_config para ingresar la etiqueta de trabajo.
10. Haga clic en Siguiente y luego en Iniciar para comenzar el trabajo.
11. Seleccione Ver → Trabajos para supervisar el progreso y los resultados del trabajo.
12. Una vez completada la función linux_config, ejecute nuevamente el proceso para oracle_config.
13. Vaya a Recursos → Plantillas.
14. Seleccione la plantilla deseada y luego haga clic en Iniciar.
15. Cuando se le solicite durante el inicio etiquetas de trabajo, escriba oracle_config. Es posible que necesites seleccionar la línea Crear "Etiqueta de trabajo" justo debajo de oracle_config para ingresar la etiqueta de trabajo.
16. Haga clic en Siguiente y luego en Iniciar para comenzar el trabajo.
17. Seleccione Ver → Trabajos para supervisar el progreso y los resultados del trabajo.

6. Implementar una base de datos adicional en el mismo host de Oracle

La parte de Oracle del libro de estrategias crea una única base de datos de contenedor de Oracle en un servidor Oracle por cada ejecución. Para crear bases de datos de contenedores adicionales en el mismo servidor, complete los siguientes pasos.

1. Revisar las variables host_vars.
 - a. Regrese al paso 2: Configurar Oracle host_vars.
 - b. Cambie el SID de Oracle a una cadena de nombre diferente.
 - c. Cambie el puerto de escucha a un número diferente.
 - d. Cambie el puerto EM Express a un número diferente si está instalando EM Express.
 - e. Copie y pegue las variables de host revisadas en el campo Variables de host de Oracle en la pestaña Detalles de configuración de host.
2. Inicie la plantilla de trabajo de implementación con solo la etiqueta oracle_config.
3. Inicie sesión en el servidor Oracle como usuario Oracle y ejecute los siguientes comandos:

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



Esto enumerará los procesos de Oracle si la instalación se completó como se esperaba y se inició Oracle DB

4. Inicie sesión en la base de datos para verificar la configuración de la base de datos y los PDB creados con los siguientes conjuntos de comandos.

```

[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

```

Esto confirma que dNFS está funcionando correctamente.

5. Conéctese a la base de datos a través del escucha para verificar la configuración del escucha de Oracle con el siguiente comando. Cambie al puerto de escucha y al nombre del servicio de base de datos

adecuados.

```
[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
```

Esto confirma que el escucha de Oracle está funcionando correctamente.

¿A dónde acudir para obtener ayuda?

Si necesita ayuda con el kit de herramientas, únase a la ["Canal de Slack de soporte de la comunidad de automatización de soluciones de NetApp"](#) y busca el canal de automatización de soluciones para publicar tus dudas o consultas.

Procedimiento de implementación paso a paso

Este documento detalla la implementación de Oracle 19c mediante la interfaz de línea de comandos de automatización (cli).

Implementación de CLI de la base de datos Oracle 19c

Esta sección cubre los pasos necesarios para preparar e implementar la base de datos Oracle19c con la CLI. Asegúrese de haber revisado el ["Sección de Primeros Pasos y Requisitos"](#) y preparó su entorno en consecuencia.

Descargar el repositorio Oracle19c

1. Desde su controlador ansible, ejecute el siguiente comando:

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

2. Después de descargar el repositorio, cambie los directorios a `na_oracle19c_deploy` `<cd na_oracle19c_deploy>`.

Editar el archivo de hosts

Complete lo siguiente antes de la implementación:

1. Edite el archivo de hosts en el directorio `na_oracle19c_deploy`.
2. En `[ontap]`, cambie la dirección IP a su IP de administración del clúster.
3. En el grupo `[oracle]`, agregue los nombres de los hosts de Oracle. El nombre del host debe resolverse a su dirección IP ya sea a través de DNS o del archivo de hosts, o debe especificarse en el host.
4. Después de completar estos pasos, guarde los cambios.

El siguiente ejemplo representa un archivo host:

```
#ONTAP Host

[ontap]

"10.61.184.183"

#Oracle hosts

[oracle]

"rtpora01"

"rtpora02"
```

Este ejemplo ejecuta el libro de estrategias e implementa Oracle 19c en dos servidores Oracle DB simultáneamente. También puedes probar con un solo servidor de base de datos. En ese caso, solo es necesario configurar un archivo de variable de host.



El libro de jugadas se ejecuta de la misma manera independientemente de cuántos hosts y bases de datos de Oracle implemente.

Edite el archivo `host_name.yml` en `host_vars`

Cada host de Oracle tiene su propio archivo de variables de host identificado por su nombre de host que contiene variables específicas del host. Puede especificar cualquier nombre para su host. Editar y copiar el `host_vars` desde la sección Host VARS Config y péguelo en su lugar deseado `host_name.yml` archivo.



Los elementos en azul deben cambiarse para que coincidan con su entorno.

Configuración de VARS del host

```
#####
##### 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"}

```

Editar el archivo vars.yml

El `vars.yml` El archivo consolida todas las variables específicas del entorno (ONTAP, Linux u Oracle) para la implementación de Oracle.

1. Edite y copie las variables de la sección VARS y péguelas en su `vars.yml` archivo.

```

#####
##### 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 alternatingly
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"

```

Ejecutar el libro de jugadas

Después de completar los requisitos previos del entorno requeridos y copiar las variables en `vars.yml` y `your_host.yml` Ahora estás listo para implementar los playbooks.



<nombre de usuario> debe cambiarse para que coincida con su entorno.

1. Ejecute el libro de jugadas de ONTAP pasando las etiquetas correctas y el nombre de usuario del clúster de ONTAP . Complete la contraseña para el clúster ONTAP y vsadmin cuando se le solicite.

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

2. Ejecute el playbook de Linux para ejecutar la parte de Linux de la implementación. Entrada para la contraseña ssh de administrador así como la contraseña sudo.

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

3. Ejecute el playbook de Oracle para ejecutar la parte de Oracle de la implementación. Entrada para la contraseña ssh de administrador así como la contraseña sudo.

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

Implementar una base de datos adicional en el mismo host de Oracle

La parte de Oracle del libro de estrategias crea una única base de datos de contenedor de Oracle en un servidor Oracle por cada ejecución. Para crear una base de datos de contenedores adicional en el mismo servidor, complete los siguientes pasos:

1. Revise las variables `host_vars`.
 - a. Regrese al paso 3 - Edite el `host_name.yml` archivar bajo `host_vars`.
 - b. Cambie el SID de Oracle a una cadena de nombre diferente.
 - c. Cambie el puerto de escucha a un número diferente.
 - d. Cambie el puerto EM Express a un número diferente si ha instalado EM Express.
 - e. Copie y pegue las variables de host revisadas en el archivo de variables de host de Oracle en `host_vars`.
2. Ejecutar el libro de jugadas con el `oracle_config` etiqueta como se muestra arriba en [Ejecutar el libro de jugadas](#).

Validar la instalación de Oracle

1. Inicie sesión en el servidor Oracle como usuario Oracle y ejecute los siguientes comandos:

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



Esto enumerará los procesos de Oracle si la instalación se completó como se esperaba y se inició Oracle DB

2. Inicie sesión en la base de datos para verificar la configuración de la base de datos y los PDB creados con los siguientes conjuntos de comandos.

```

[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

```

Esto confirma que dNFS está funcionando correctamente.

3. Conéctese a la base de datos a través del escucha para verificar la configuración del escucha de Oracle con el siguiente comando. Cambie al puerto de escucha y al nombre del servicio de base de datos

adecuados.

```
[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
```

Esto confirma que el escucha de Oracle está funcionando correctamente.

¿A dónde acudir para obtener ayuda?

Si necesita ayuda con el kit de herramientas, únase a la ["Canal de Slack de soporte de la comunidad de automatización de soluciones de NetApp"](#) y busca el canal de automatización de soluciones para publicar tus dudas o consultas.

Información de copyright

Copyright © 2025 NetApp, Inc. Todos los derechos reservados. Imprimido en EE. UU. No se puede reproducir este documento protegido por copyright ni parte del mismo de ninguna forma ni por ningún medio (gráfico, electrónico o mecánico, incluidas fotocopias, grabaciones o almacenamiento en un sistema de recuperación electrónico) sin la autorización previa y por escrito del propietario del copyright.

El software derivado del material de NetApp con copyright está sujeto a la siguiente licencia y exención de responsabilidad:

ESTE SOFTWARE LO PROPORCIONA NETAPP «TAL CUAL» Y SIN NINGUNA GARANTÍA EXPRESA O IMPLÍCITA, INCLUYENDO, SIN LIMITAR, LAS GARANTÍAS IMPLÍCITAS DE COMERCIALIZACIÓN O IDONEIDAD PARA UN FIN CONCRETO, CUYA RESPONSABILIDAD QUEDA EXIMIDA POR EL PRESENTE DOCUMENTO. EN NINGÚN CASO NETAPP SERÁ RESPONSABLE DE NINGÚN DAÑO DIRECTO, INDIRECTO, ESPECIAL, EJEMPLAR O RESULTANTE (INCLUYENDO, ENTRE OTROS, LA OBTENCIÓN DE BIENES O SERVICIOS SUSTITUTIVOS, PÉRDIDA DE USO, DE DATOS O DE BENEFICIOS, O INTERRUPTIÓN DE LA ACTIVIDAD EMPRESARIAL) CUALQUIERA SEA EL MODO EN EL QUE SE PRODUJERON Y LA TEORÍA DE RESPONSABILIDAD QUE SE APLIQUE, YA SEA EN CONTRATO, RESPONSABILIDAD OBJETIVA O AGRAVIO (INCLUIDA LA NEGLIGENCIA U OTRO TIPO), QUE SURJAN DE ALGÚN MODO DEL USO DE ESTE SOFTWARE, INCLUSO SI HUBIEREN SIDO ADVERTIDOS DE LA POSIBILIDAD DE TALES DAÑOS.

NetApp se reserva el derecho de modificar cualquiera de los productos aquí descritos en cualquier momento y sin aviso previo. NetApp no asume ningún tipo de responsabilidad que surja del uso de los productos aquí descritos, excepto aquello expresamente acordado por escrito por parte de NetApp. El uso o adquisición de este producto no lleva implícita ninguna licencia con derechos de patente, de marcas comerciales o cualquier otro derecho de propiedad intelectual de NetApp.

Es posible que el producto que se describe en este manual esté protegido por una o más patentes de EE. UU., patentes extranjeras o solicitudes pendientes.

LEYENDA DE DERECHOS LIMITADOS: el uso, la copia o la divulgación por parte del gobierno están sujetos a las restricciones establecidas en el subpárrafo (b)(3) de los derechos de datos técnicos y productos no comerciales de DFARS 252.227-7013 (FEB de 2014) y FAR 52.227-19 (DIC de 2007).

Los datos aquí contenidos pertenecen a un producto comercial o servicio comercial (como se define en FAR 2.101) y son propiedad de NetApp, Inc. Todos los datos técnicos y el software informático de NetApp que se proporcionan en este Acuerdo tienen una naturaleza comercial y se han desarrollado exclusivamente con fondos privados. El Gobierno de EE. UU. tiene una licencia limitada, irrevocable, no exclusiva, no transferible, no sublicenciable y de alcance mundial para utilizar los Datos en relación con el contrato del Gobierno de los Estados Unidos bajo el cual se proporcionaron los Datos. Excepto que aquí se disponga lo contrario, los Datos no se pueden utilizar, desvelar, reproducir, modificar, interpretar o mostrar sin la previa aprobación por escrito de NetApp, Inc. Los derechos de licencia del Gobierno de los Estados Unidos de América y su Departamento de Defensa se limitan a los derechos identificados en la cláusula 252.227-7015(b) de la sección DFARS (FEB de 2014).

Información de la marca comercial

NETAPP, el logotipo de NETAPP y las marcas que constan en <http://www.netapp.com/TM> son marcas comerciales de NetApp, Inc. El resto de nombres de empresa y de producto pueden ser marcas comerciales de sus respectivos propietarios.