



## **Cloud ibrido/on-premise**

### NetApp Solutions

NetApp  
May 03, 2024

# Sommario

- Cloud ibrido/on-premise ..... 1
  - TR-4983: Implementazione di Oracle semplificata e automatizzata su NetApp ASA con iSCSI ..... 1
  - NVA-1155: Database Oracle 19c RAC su data center FlexPod con Cisco UCS e NetApp AFF A800 su FC - Guida alla progettazione e all'implementazione ..... 18
  - TR-4250: SAP con Oracle su UNIX e NFS con NetApp Clustered Data ONTAP e SnapManager per SAP 3.4 ..... 18
  - Implementazione di Oracle Database ..... 18
  - Panoramica della soluzione ..... 39
  - TR-4794: Database Oracle su NetApp EF-Series ..... 62

# Cloud ibrido/on-premise

## TR-4983: Implementazione di Oracle semplificata e automatizzata su NetApp ASA con iSCSI

Allen Cao, Niyaz Mohamed, NetApp

### Scopo

I sistemi NetApp ASA offrono soluzioni moderne per la tua infrastruttura SAN. Essi semplificano su larga scala e ti permettono di accelerare le applicazioni business-critical come i database, assicurano che i tuoi dati siano sempre disponibili (uptime del 99,9999%) e riducono il TCO e l'impronta di carbonio. I sistemi NetApp ASA includono modelli A-Series progettati per le applicazioni più esigenti in termini di performance e modelli C-Series ottimizzati per implementazioni convenienti e con capacità elevata. Insieme, i sistemi ASA A-Series e C-Series offrono performance eccezionali per migliorare l'esperienza dei clienti e ridurre i tempi di risultati, mantenere i dati business-critical disponibili, protetti e sicuri e fornire una capacità più effettiva per qualsiasi carico di lavoro, supportato dalla garanzia più vantaggiosa del settore.

Questa documentazione dimostra l'implementazione semplificata dei database Oracle in un ambiente SAN costruito con sistemi ASA utilizzando l'automazione Ansible. Il database Oracle viene installato in una configurazione di riavvio standalone con protocollo iSCSI per l'accesso ai dati e Oracle ASM per la gestione dei dischi del database sull'array di archiviazione ASA. Il prodotto offre anche informazioni su backup, ripristino e cloning del database Oracle attraverso il tool dell'interfaccia utente di NetApp SnapCenter per il funzionamento efficiente in termini di storage dei database nei sistemi NetApp ASA.

Questa soluzione risolve i seguenti casi di utilizzo:

- Distribuzione automatizzata del database Oracle su sistemi NetApp ASA come storage primario per il database
- Backup e ripristino del database Oracle in sistemi NetApp ASA con il tool NetApp SnapCenter
- Clone del database Oracle per sviluppo/test o altri casi di utilizzo nei sistemi NetApp ASA con il tool NetApp SnapCenter

### Pubblico

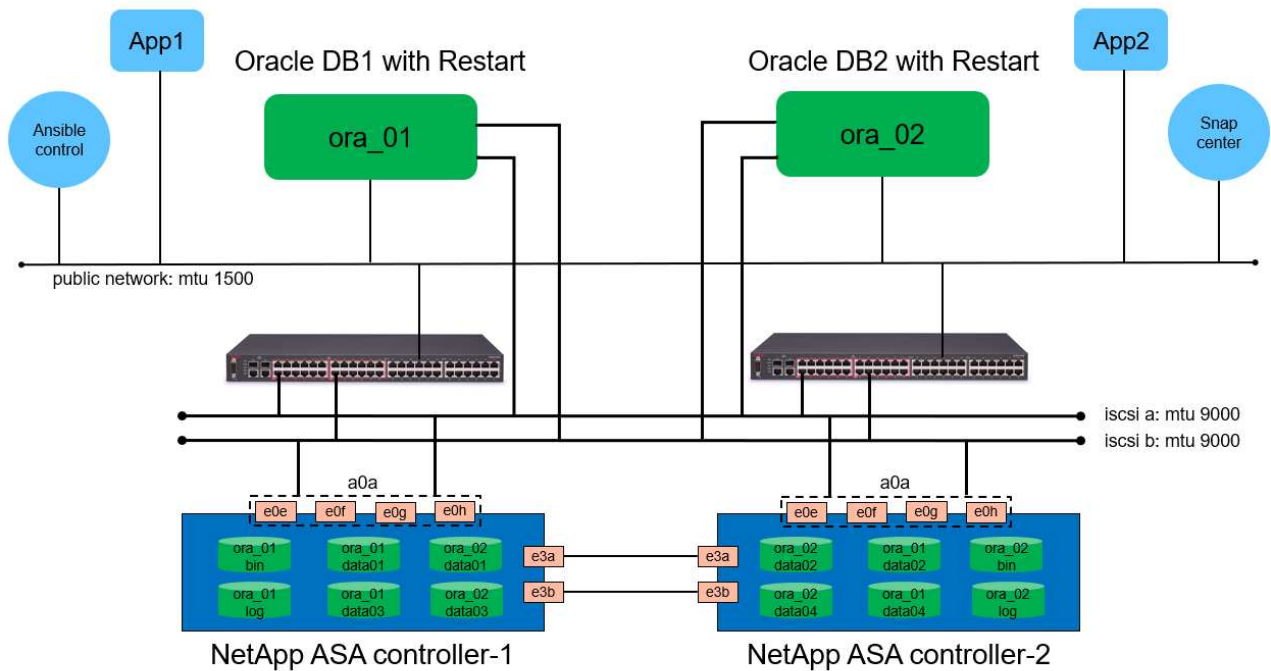
Questa soluzione è destinata alle seguenti persone:

- Un DBA che vorrebbe implementare Oracle su sistemi NetApp ASA.
- Un Solution Architect per database che vorrebbe testare i carichi di lavoro Oracle nei sistemi NetApp ASA.
- Un amministratore dello storage che desidera implementare e gestire un database Oracle su sistemi NetApp ASA.
- Un proprietario di applicazioni che vorrebbe creare un database Oracle nei sistemi NetApp ASA.

### Ambiente di test e convalida della soluzione

Il test e la convalida di questa soluzione sono stati eseguiti in un laboratorio che potrebbe non corrispondere all'ambiente di distribuzione finale. Vedere la sezione [\[Key Factors for Deployment Consideration\]](#) per ulteriori informazioni.

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



NetApp

### Componenti hardware e software

Hardware		
NetApp ASA A400	Versione 9.13.1P1	2 NS224 shelf, 48 dischi AFF NVMe con una capacità totale di 69,3 TiB
UCSB-B200-M4	CPU Intel® Xeon® E5-2690 v4 a 2,60GHz MHz	Cluster VMware ESXi a 4 nodi
Software		
RedHat Linux	Kernel RHEL-8,6, 4.18.0-372.9.1.EL8.x86_64	Implementazione dell'abbonamento a RedHat per il test
Server Windows	2022 Standard, 10.0.20348 Build 20348	Server SnapCenter di hosting
Oracle Grid Infrastructure	Versione 19.18	Patch RU applicata p34762026_190000_Linux-x86-64.zip
Database Oracle	Versione 19.18	Patch RU applicata p34765931_190000_Linux-x86-64.zip
Oracle OPatch	Versione 12.2.0.1.36	Ultima patch p6880880_190000_Linux-x86-64.zip
Server SnapCenter	Versione 4,9P1	Distribuzione di gruppi di lavoro

Hypervisor VMware vSphere	versione 6.5.0.20000	VMware Tools, versione: 11365 - Linux, 12352 - Windows
Aprire JDK	Versione java-1,8.0-openjdk.x86_64	Requisito del plugin SnapCenter per macchine virtuali DB

### Configurazione del database Oracle nell'ambiente di laboratorio

Server	Database	Archiviazione DB
ora_01	NTAP1 (NTAP1_PDB1,NTAP1_PDB2,NTAP1_PDB3)	lun iSCSI su ASA A400
ora_02	NTAP2 (NTAP2_PDB1,NTAP2_PDB2,NTAP2_PDB3)	lun iSCSI su ASA A400

### Fattori chiave per l'implementazione

- **Layout dello storage dei database Oracle.** in questa distribuzione automatizzata di Oracle, vengono forniti quattro volumi di database per l'hosting di file binari, dati e registri Oracle per impostazione predefinita. Creiamo quindi due gruppi di dischi ASM dai dati e dai registri delle lun. All'interno del gruppo di dischi asm +DATA, eseguiamo il provisioning di due lun di dati in un volume su ciascun nodo del cluster ASA A400. All'interno del gruppo di dischi asm +LOGS, vengono create due lun in un volume di registro su un singolo nodo ASA A400. La presenza di diverse lun in un volume ONTAP offre performance generali migliori.
- **Implementazione di più server DB.** la soluzione di automazione può implementare un database container Oracle su più server DB in un singolo playbook Ansible. Indipendentemente dal numero di server di DB, l'esecuzione del playbook rimane invariata. In caso di implementazioni di server con più database, il playbook utilizza un algoritmo per posizionare le lun del database in modo ottimale sui dual controller di ASA A400. Il file binario e registra i lun del server DB con numero dispari negli host del server e la posizione dell'indice sul controller 1. Il file binario e registra i lun del server DB numero pari nell'indice degli host del server sul controller 2. Le lun dei dati del database vengono distribuite in modo uniforme in due controller. Oracle ASM combina le lun dei dati su due controller in un unico gruppo di dischi ASM per sfruttare al massimo la potenza di elaborazione di entrambi i controller.
- **Configurazione iSCSI.** le macchine virtuali del database si connettono allo storage ASA con il protocollo iSCSI per l'accesso allo storage. È necessario configurare percorsi doppi su ciascun nodo del controller per la ridondanza e impostare percorsi multipli iSCSI sul server DB per l'accesso allo storage multi-path. Abilitazione di frame jumbo su storage network per massimizzare performance e throughput.
- **Livello di ridondanza di Oracle ASM da utilizzare per ogni gruppo di dischi Oracle ASM creato.** poiché ASA A400 configura lo spazio di archiviazione in RAID DP per la protezione dei dati a livello di disco del cluster, è necessario utilizzare `External Redundancy`, Il che significa che l'opzione non consente ad Oracle ASM di eseguire il mirroring del contenuto del gruppo di dischi.
- **Backup del database.** NetApp fornisce una suite software SnapCenter per il backup, il ripristino e la clonazione del database con un'interfaccia utente intuitiva. NetApp consiglia di implementare questo strumento di gestione per ottenere veloci backup delle snapshot (in meno di un minuto), rapidi ripristini del database e cloni del database.

## **Implementazione della soluzione**

Nelle sezioni seguenti vengono fornite procedure dettagliate per l'implementazione e la protezione automatizzate di Oracle 19c in NetApp ASA A400 con lun dei database montati direttamente tramite iSCSI e DB VM in una configurazione di riavvio a nodo singolo con Oracle ASM come volume manager del database.

### **Prerequisiti per l'implementazione**

L'implementazione richiede i seguenti prerequisiti.

1. Si presuppone che lo storage array NetApp ASA sia stato installato e configurato. Sono inclusi dominio di broadcast iSCSI, gruppi di interfacce LACP a0a su entrambi i nodi del controller, porte VLAN iSCSI (a0a-<iscsi-a-vlan-id>, a0a-<iscsi-b-vlan-id>) su entrambi i nodi del controller. Il seguente collegamento fornisce istruzioni dettagliate, se è necessaria assistenza. ["Guida dettagliata - ASA A400"](#)
2. Provisioning di una VM Linux come nodo di controller Ansible con l'ultima versione di Ansible e Git installata. Fare riferimento al seguente link per i dettagli: ["Introduzione all'automazione delle soluzioni NetApp"](#) nella sezione - Setup the Ansible Control Node for CLI deployments on RHEL / CentOS oppure Setup the Ansible Control Node for CLI deployments on Ubuntu / Debian.
3. Clonazione di una copia del toolkit di automazione della distribuzione Oracle di NetApp per iSCSI.

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

4. Eseguire il provisioning di un server Windows per eseguire lo strumento dell'interfaccia utente di NetApp SnapCenter con la versione più recente. Fare riferimento al seguente link per i dettagli: ["Installare il server SnapCenter"](#)
5. Costruisci due server RHEL Oracle DB sia bare metal che macchine virtuali virtualizzate. Crea un utente admin su server DB con sudo senza privilegio password e abilita l'autenticazione a chiave privata/pubblica SSH tra host Ansible e host server Oracle DB. Fase successiva ai file di installazione di Oracle 19c nella directory server DB /tmp/archivio.

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



Assicurarsi di aver allocato almeno 50g MB nel volume root di Oracle VM per disporre di spazio sufficiente per preparare i file di installazione di Oracle.

6. Guarda il seguente video:

[Distribuzione Oracle semplificata e automatizzata su NetApp ASA con iSCSI](#)

## File dei parametri di automazione

Il playbook Ansible esegue attività di installazione e configurazione del database con parametri predefiniti. Per questa soluzione di automazione Oracle, esistono tre file di parametri definiti dall'utente che devono essere inseriti dall'utente prima dell'esecuzione del playbook.

- host - definisci gli obiettivi per i quali il playbook di automazione è in esecuzione.
- vars/vars.yml - il file variabile globale che definisce le variabili che si applicano a tutti i target.
- host\_vars/host\_name.yml - il file delle variabili locali che definisce le variabili che si applicano solo a una destinazione locale. Nel nostro caso d'utilizzo, questi sono i server Oracle DB.

Oltre a questi file di variabili definiti dall'utente, esistono diversi file di variabili predefinite che contengono parametri predefiniti che non richiedono modifiche se non necessario. Le sezioni seguenti mostrano come sono configurati i file variabili definiti dall'utente.

## **Configurazione dei file dei parametri**



## 1. Destinazione Ansible hosts configurazione file:

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

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

## 2. Globale vars/vars.yml configurazione dei file

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

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

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

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

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

```

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

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

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

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

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

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

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

```

### 3. Server DB locale host\_vars/host\_name.yml configurazione

```
# User configurable Oracle host specific parameters

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

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

## Esecuzione Playbook

Nel toolkit di automazione sono presenti sei playbook in totale. Ciascuna di esse esegue blocchi di attività diversi e ha scopi diversi.

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

Sono disponibili tre opzioni per eseguire i playbook con i seguenti comandi.

1. Esegui tutti i playbook sull'implementazione in un'unica esecuzione combinata.

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

2. Eseguire i playbook uno alla volta con la sequenza numerica da 1 a 4.

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

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

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

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

3. Esegui 0-all\_playbook.yml con un tag.

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

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

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

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

#### 4. Annullare l'ambiente

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

### Convalida post-esecuzione

Dopo aver eseguito il playbook, effettua l'accesso al server Oracle DB come utente oracle per validare la corretta creazione dell'infrastruttura Oracle Grid e del database. Di seguito viene riportato un esempio di convalida del database Oracle sull'host ora\_01.

1. Convalidare l'infrastruttura di rete e le risorse create.

```
[oracle@ora_01 ~]$ df -h
Filesystem                Size      Used Avail Use% Mounted on
devtmpfs                  7.7G       40K   7.7G   1% /dev
tmpfs                     7.8G      1.1G   6.7G  15% /dev/shm
tmpfs                     7.8G       312M   7.5G   4% /run
tmpfs                     7.8G        0   7.8G   0% /sys/fs/cgroup
/dev/mapper/rhel-root      44G       38G   6.8G  85% /
/dev/sda1                 1014M     258M   757M  26% /boot
tmpfs                     1.6G       12K   1.6G   1% /run/user/42
tmpfs                     1.6G       4.0K   1.6G   1% /run/user/1000
/dev/mapper/ora_01_biny_01p1 40G      21G    20G  52% /u01
[oracle@ora_01 ~]$ asm
[oracle@ora_01 ~]$ crsctl stat res -t
-----
-----
Name                Target  State          Server          State
details
-----
-----
Local Resources
-----
-----
ora.DATA.dg
                ONLINE  ONLINE         ora_01          STABLE
ora.LISTENER.lsnr
                ONLINE  INTERMEDIATE  ora_01          Not All
Endpoints Re
gistered, STABLE
ora.LOGS.dg
                ONLINE  ONLINE         ora_01          STABLE
ora.asm
                ONLINE  ONLINE         ora_01
Started, STABLE
ora.ons
                OFFLINE OFFLINE        ora_01          STABLE
-----
-----
Cluster Resources
-----
```

```

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

racle/product/19.0.0

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

```



Ignorare Not All Endpoints Registered Nei dettagli dello stato. Ciò deriva da un conflitto di registrazione manuale e dinamica del database con il listener e può essere ignorato in modo sicuro.

2. Verificare che il driver del filtro ASM funzioni come previsto.

```


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

```

3. Accedere a Oracle Enterprise Manager Express per convalidare il database.



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




# ORACLE ENTERPRISE MANAGER DATABASE EXPRESS

Username

Password


Container Name

[Log in](#)



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

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


Enterprise Manager Database Express
system ▾

NTAP1 (19.18.0.0.0) Performance ▾ Storage ▾

## Database Home

Time Zone: Browser (GMT-05:00) ▾ 1 min Auto-Refresh ▾ Refresh

**Status**

Up Time 1 hours, 7 minutes, 23 seconds

Type Single Instance (NTAP1)

CDB (3 PDB(s))

Version 19.18.0.0.0 Enterprise Edition

Platform Name Linux x86\_64-bit

Thread 1

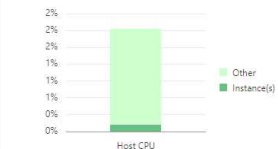
Archiver Stopped

Last Backup Time N/A

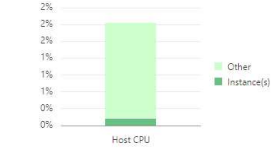
Incident(s) ✖ 4

**Performance**

Activity Services Containers

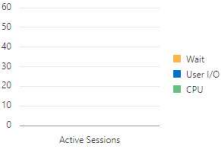


**Resources**



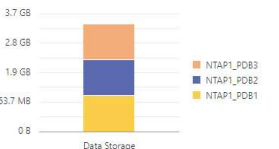
Host CPU: 2%

Active Sessions



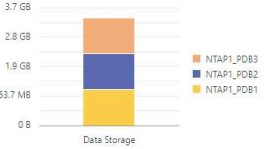
Active Sessions: 0

Memory



Memory: 14 GB

Data Storage



Data Storage: 3.7 GB

**SQL Monitor - Last Hour (20 max)**

Top 20 by Last Active Time ▾ Filter by Status, SQL ID or User Name

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

```
SQL> show pdbs
```

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

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

Session altered.

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

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

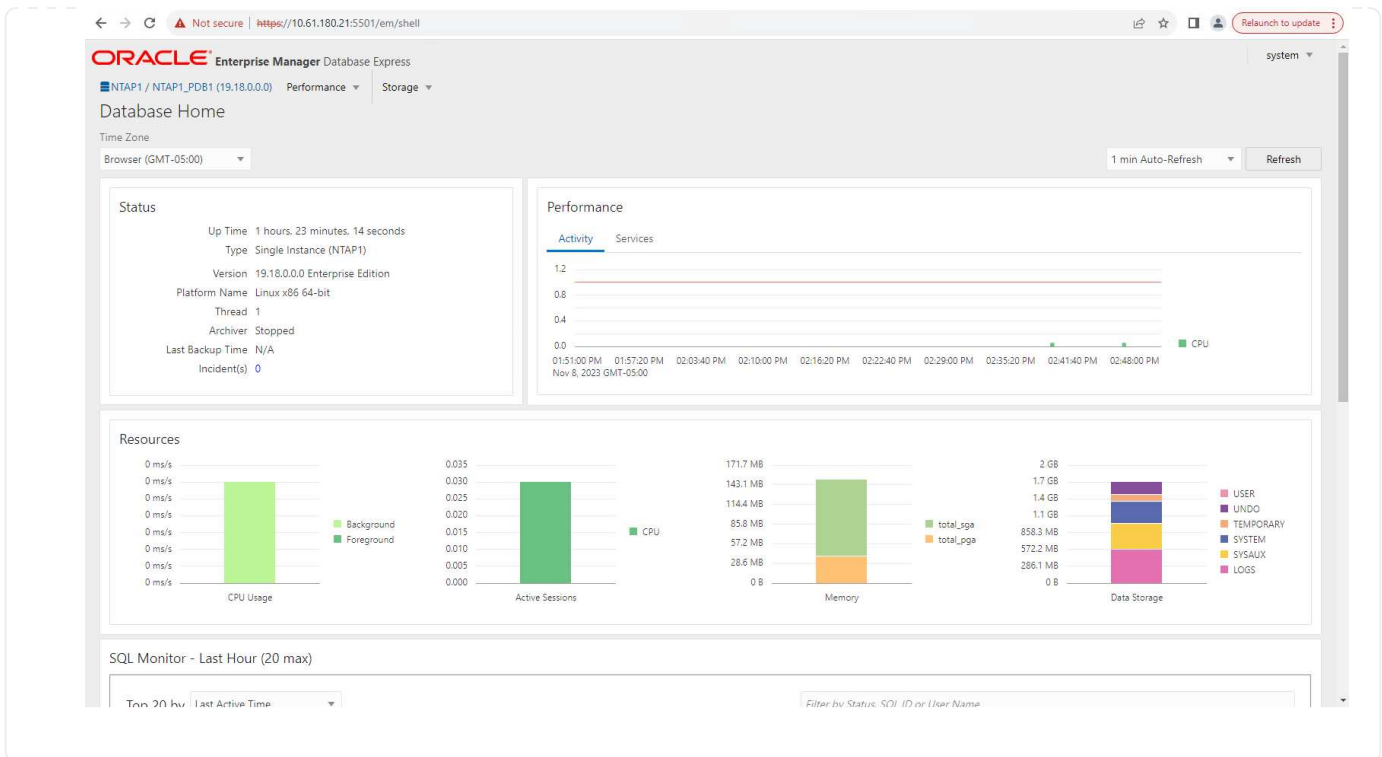
```
SQL> exec DBMS_XDB_CONFIG.SETHTTPS(5501);
```

PL/SQL procedure successfully completed.

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

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

login to NTAP1\_PDB1 from port 5501.



## Backup, ripristino e clonaggio di Oracle con SnapCenter

Fare riferimento a TR-4979 "Oracle semplificata e autogestita in VMware Cloud su AWS con FSX ONTAP montato su guest" sezione Oracle backup, restore, and clone with SnapCenter Per informazioni dettagliate su configurazione di SnapCenter ed esecuzione di flussi di lavoro di backup, ripristino e clonaggio del database.

## Dove trovare ulteriori informazioni

Per ulteriori informazioni sulle informazioni descritte in questo documento, consultare i seguenti documenti e/o siti Web:

- NetApp ASA: ARRAY ALL-FLASH SAN

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

- Installazione di Oracle Grid Infrastructure per un server standalone con un'installazione di un nuovo database

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

- Installazione e configurazione del database Oracle mediante i file di risposta

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

- Utilizza Red Hat Enterprise Linux 8.2 con ONTAP

## **NVA-1155: Database Oracle 19c RAC su data center FlexPod con Cisco UCS e NetApp AFF A800 su FC - Guida alla progettazione e all'implementazione**

Allen Cao, NetApp

Questa guida alla progettazione e all'implementazione dei database Oracle 19c RAC su FlexPod Datacenter con Cisco UCS e NetApp AFF A800 su FC fornisce dettagli sulla progettazione della soluzione e sui processi di implementazione passo-passo per l'hosting dei database Oracle RAC sulla più recente infrastruttura FlexPod Datacenter con Oracle Linux 8.2 Sistema operativo e kernel compatibile con Red Hat.

["NVA-1155: Database Oracle 19c RAC su data center FlexPod con Cisco UCS e NetApp AFF A800 su FC"](#)

## **TR-4250: SAP con Oracle su UNIX e NFS con NetApp Clustered Data ONTAP e SnapManager per SAP 3.4**

Nils Bauer, NetApp

TR-4250 affronta le sfide legate alla progettazione di soluzioni storage per supportare i prodotti di business suite SAP utilizzando un database Oracle. L'obiettivo principale di questo documento è rappresentato dalle sfide comuni di progettazione, implementazione, funzionamento e gestione dell'infrastruttura storage affrontate dai leader aziendali e IT che utilizzano le soluzioni SAP di ultima generazione. Le raccomandazioni contenute in questo documento sono generiche e non sono specifiche di un'applicazione SAP o delle dimensioni e dell'ambito dell'implementazione SAP. TR-4250 presuppone che il lettore abbia una conoscenza di base della tecnologia e del funzionamento dei prodotti NetApp e SAP. TR-4250 è stato sviluppato in base all'interazione dello staff tecnico di NetApp, SAP, Oracle e dei nostri clienti.

["TR-4250: SAP con Oracle su UNIX e NFS con NetApp Clustered Data ONTAP e SnapManager per SAP 3.4"](#)

## **Implementazione di Oracle Database**

### **Panoramica della soluzione**

#### **Implementazione automatica di Oracle19c per ONTAP su NFS**

Le organizzazioni stanno automatizzando i propri ambienti per ottenere efficienze, accelerare le implementazioni e ridurre l'impegno manuale. I tool di gestione della configurazione come Ansible vengono utilizzati per ottimizzare le operazioni dei database aziendali. In questa soluzione, dimostreremo come utilizzare Ansible per automatizzare il provisioning e la configurazione di Oracle 19c con NetApp ONTAP. Consentendo agli amministratori dello storage, agli amministratori di sistema e ai DBA di implementare in modo coerente e rapido nuovo storage, configurare server di database e installare il software Oracle 19c, otterrete i seguenti vantaggi:

- Elimina le complessità di progettazione e gli errori umani e implementa un'implementazione coerente e ripetibile e Best practice
- Riduzione dei tempi di provisioning dello storage, configurazione degli host DB e installazione di Oracle
- Aumenta la produttività di amministratori di database, sistemi e amministratori dello storage

- Scalabilità di storage e database con facilità

NetApp offre ai clienti i moduli e i ruoli Ansible validati per accelerare l'implementazione, la configurazione e la gestione del ciclo di vita del tuo ambiente di database Oracle. Questa soluzione fornisce istruzioni e codice del playbook Ansible per aiutarti a:

- Creare e configurare lo storage NFS ONTAP per il database Oracle
- Installare Oracle 19c su RedHat Enterprise Linux 7/8 o Oracle Linux 7/8
- Configurare Oracle 19c sullo storage NFS ONTAP

Per ulteriori dettagli o per iniziare, consulta i video di panoramica riportati di seguito.

### **Implementazioni AWX/tower**

Parte 1: Introduzione, requisiti, dettagli di automazione e configurazione iniziale AWX/Tower

#### [Implementazione AWX](#)

Parte 2: Variabili ed esecuzione del Playbook

#### [AWX Playbook Run](#)

### **Implementazione della CLI**

Parte 1: Guida introduttiva, requisiti, dettagli di automazione e configurazione host di Ansible Control

#### [Implementazione della CLI](#)

Parte 2: Variabili ed esecuzione del Playbook

#### [Esecuzione del Playbook CLI](#)

### **Per iniziare**

Questa soluzione è stata progettata per essere eseguita in un ambiente AWX/Tower o da CLI su un host di controllo Ansible.

### **AWX/Tower**

Per gli ambienti AWX/tower, viene fornita una guida alla creazione di un inventario della gestione del cluster ONTAP e del server Oracle (IP e nomi host), alla creazione di credenziali, alla configurazione di un progetto che estrae il codice Ansible da NetApp Automation Github e al modello di lavoro che avvia l'automazione.

1. Compilare le variabili specifiche del proprio ambiente, quindi copiarle e incollarle nei campi Extra Vars del modello di lavoro.
2. Una volta aggiunti i var aggiuntivi al modello di lavoro, è possibile avviare l'automazione.
3. Il modello di lavoro viene eseguito in tre fasi specificando i tag per ontap\_CONFIG, linux\_CONFIG e oracle\_CONFIG.

### **CLI tramite l'host di controllo Ansible**

1. Per configurare l'host Linux in modo che sia possibile utilizzarlo come host di controllo Ansible ["fare clic qui per istruzioni dettagliate"](#)

2. Una volta configurato l'host di controllo Ansible, è possibile clonare il repository Ansible Automation.
3. Modificare il file hosts con gli IP e/o i nomi host della gestione del cluster ONTAP e degli IP di gestione del server Oracle.
4. Compilare le variabili specifiche dell'ambiente, quindi copiarle e incollarle in `vars.yml` file.
5. Ogni host Oracle dispone di un file variabile identificato dal relativo nome host che contiene variabili specifiche dell'host.
6. Una volta completati tutti i file variabili, è possibile eseguire il playbook in tre fasi specificando i tag per `ontap_config`, `linux_config`, e `oracle_config`.

## Requisiti

Ambiente	Requisiti
<b>Ambiente Ansible</b>	Host AWX/Tower o Linux come host di controllo Ansible
	Ansible v.2.10 e versioni successive
	Python 3
	Librerie Python - netapp-lib - xmltodict - jmespath
<b>ONTAP</b>	ONTAP versione 9.3 - 9.7
	Due aggregati di dati
	VLAN NFS e ifgrp create
<b>Server Oracle</b>	RHEL 7/8
	Oracle Linux 7/8
	Interfacce di rete per NFS, gestione pubblica e opzionale
	File di installazione Oracle sui server Oracle

## Dettagli sull'automazione

Questa implementazione automatica è progettata con un singolo playbook Ansible che consiste di tre ruoli separati. I ruoli sono per le configurazioni ONTAP, Linux e Oracle. La seguente tabella descrive le attività automatizzate.

Ruolo	Attività
<b>ontap_config</b>	Verifica preliminare dell'ambiente ONTAP
	Creazione di SVM basate su NFS per Oracle
	Creazione di policy di esportazione
	Creazione di volumi per Oracle
	Creazione di LIF NFS

Ruolo	Attività
linux_config	Creare punti di montaggio e montare volumi NFS
	Verificare i montaggi NFS
	Configurazione specifica del sistema operativo
	Creare directory Oracle
	Configurare gli hugepage
	Disattiva SELinux e il daemon del firewall
	Attivare e avviare il servizio chronyd
	aumentare il limite massimo del descrittore di file
	Creare il file di sessione pam.d.
oracle_config	Installazione del software Oracle
	Creare un listener Oracle
	Creare database Oracle
	Configurazione dell'ambiente Oracle
	Salva stato PDB
	Attivare la modalità di archiviazione delle istanze
	Abilitare il client DNFS
	Abilitare l'avvio e lo spegnimento automatici del database tra i riavvii del sistema operativo

### Parametri predefiniti

Per semplificare l'automazione, abbiamo preimpostato molti parametri di implementazione Oracle richiesti con valori predefiniti. In genere non è necessario modificare i parametri predefiniti per la maggior parte delle implementazioni. Un utente più avanzato può apportare modifiche ai parametri predefiniti con cautela. I parametri predefiniti si trovano in ogni cartella di ruoli nella directory dei valori predefiniti.

### Istruzioni per l'implementazione

Prima di iniziare, scaricare i seguenti file di installazione e patch Oracle e inserirli in /tmp/archive Directory con accesso in lettura, scrittura ed esecuzione per tutti gli utenti su ciascun server DB da implementare. Le attività di automazione cercano i file di installazione denominati in quella particolare directory per l'installazione e la configurazione di 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
```

### Licenza

Leggere le informazioni sulla licenza come indicato nel repository Github. Accedendo, scaricando, installando o utilizzando il contenuto di questo repository, l'utente accetta i termini della licenza stabilita "qui".

Si noti che esistono alcune limitazioni relative alla produzione e/o alla condivisione di qualsiasi opera derivata con il contenuto di questo repository. Leggere attentamente i termini del "[Licenza](#)" prima di utilizzare il contenuto. Se non si accettano tutti i termini, non accedere, scaricare o utilizzare il contenuto di questo repository.

Una volta pronti, fare clic su "[Qui per le procedure di implementazione AWX/Tower dettagliate](#)" oppure "[Qui per l'implementazione della CLI](#)".

## Procedura di implementazione passo-passo

### Implementazione AWX/Tower Database Oracle 19c

#### 1. Creare l'inventario, il gruppo, gli host e le credenziali per il proprio ambiente

Questa sezione descrive la configurazione di inventario, gruppi, host e credenziali di accesso in AWX/Ansible Tower che preparano l'ambiente per l'utilizzo delle soluzioni automatizzate di NetApp.

1. Configurare l'inventario.
  - a. Accedere a Resources → Inventories → Add e fare clic su Add Inventory (Aggiungi inventario).
  - b. Fornire il nome e i dettagli dell'organizzazione, quindi fare clic su Save (Salva).
  - c. Nella pagina Inventories (inventari), fare clic sull'inventario creato.
  - d. Se sono presenti variabili di inventario, incollarle nel campo variabili.
  - e. Accedere al sottomenu Groups (gruppi) e fare clic su Add (Aggiungi).
  - f. Fornire il nome del gruppo per ONTAP, incollare le variabili di gruppo (se presenti) e fare clic su Salva.
  - g. Ripetere la procedura per un altro gruppo per Oracle.
  - h. Selezionare il gruppo ONTAP creato, accedere al sottomenu hosts e fare clic su Aggiungi nuovo host.
  - i. Fornire l'indirizzo IP dell'IP di gestione del cluster ONTAP, incollare le variabili host (se presenti) e fare clic su Salva.
  - j. Questo processo deve essere ripetuto per l'IP/nome host di gestione del gruppo Oracle e degli host Oracle.
2. Creare tipi di credenziale. Per le soluzioni che utilizzano ONTAP, è necessario configurare il tipo di credenziale in modo che corrisponda alle voci di nome utente e password.
  - a. Accedere a Administration → Credential Types (Amministrazione tipi di credenziali) e fare clic su Add (Aggiungi).
  - b. Fornire il nome e la descrizione.
  - c. Incollare il seguente contenuto in Input Configuration (Configurazione input):



```
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. Incollare il seguente contenuto nella configurazione dell'iniettore:

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

1. Configurare le credenziali.

- a. Accedere a Resources → Credentials (risorse credenziali) e fare clic su Add (Aggiungi).
- b. Immettere il nome e i dettagli dell'organizzazione per ONTAP.
- c. Selezionare il tipo di credenziale personalizzato creato per ONTAP.
- d. In Dettagli tipo, immettere il nome utente, la password e la password vsadmin\_password.
- e. Fare clic su Torna alla credenziale e fare clic su Aggiungi.
- f. Immettere il nome e i dettagli dell'organizzazione per Oracle.
- g. Selezionare il tipo di credenziale Machine.
- h. In Dettagli tipo, immettere il nome utente e la password per gli host Oracle.
- i. Selezionare il metodo corretto di escalation dei privilegi e immettere il nome utente e la password.

## 2. Creare un progetto

1. Accedere a risorse → progetti e fare clic su Aggiungi.
  - a. Inserire il nome e i dettagli dell'organizzazione.
  - b. Selezionare Git nel campo Source Control Credential Type (tipo credenziale controllo origine).
  - c. invio <[https://github.com/NetApp-Automation/na\\_oracle19c\\_deploy.git](https://github.com/NetApp-Automation/na_oracle19c_deploy.git)> Come URL del controllo di origine.
  - d. Fare clic su Salva.
  - e. Potrebbe essere necessario sincronizzare il progetto occasionalmente quando il codice sorgente cambia.

### 3. Configurare Oracle host\_vars

Le variabili definite in questa sezione vengono applicate a ogni singolo server e database Oracle.

1. Inserire i parametri specifici dell'ambiente nel seguente modulo host Oracle incorporato o host\_vars.



Gli elementi in blu devono essere modificati in base all'ambiente in uso.

#### Config. VAR 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. Inserire tutte le variabili nei campi blu.
2. Una volta completata l'immissione delle variabili, fare clic sul pulsante Copy (Copia) del modulo per copiare tutte le variabili da trasferire su AWX o Tower.
3. Tornare a AWX o Tower e andare a Resources → hosts, quindi selezionare e aprire la pagina di configurazione del server Oracle.
4. Nella scheda Dettagli, fare clic su Modifica e incollare le variabili copiate dal punto 1 nel campo variabili sotto la scheda YAML.
5. Fare clic su Salva.
6. Ripetere questa procedura per tutti i server Oracle aggiuntivi nel sistema.

#### 4. Configurare le variabili globali

Le variabili definite in questa sezione si applicano a tutti gli host Oracle, ai database e al cluster ONTAP.

1. Inserire i parametri specifici dell'ambiente nel seguente formato vars o variabili globali incorporate.



Gli elementi in blu devono essere modificati in base all'ambiente in uso.

```
#####  
##### 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. Inserire tutte le variabili nei campi blu.
2. Una volta completata l'immissione delle variabili, fare clic sul pulsante Copy (Copia) del modulo per copiare tutte le variabili da trasferire a AWX o Tower nel seguente modello di lavoro.

## 5. Configurare e avviare il modello di lavoro.

1. Creare il modello di lavoro.
  - a. Accedere a risorse → modelli → Aggiungi e fare clic su Aggiungi modello di processo.
  - b. Immettere il nome e la descrizione
  - c. Selezionare il tipo di lavoro; Esegui consente di configurare il sistema in base a un playbook e Check esegue un'esecuzione a secco di un playbook senza configurare effettivamente il sistema.
  - d. Seleziona l'inventario, il progetto, il playbook e le credenziali corrispondenti per il playbook.
  - e. Selezionare all\_playbook.yml come playbook predefinito da eseguire.
  - f. Incollare le variabili globali copiate dal passaggio 4 nel campo Template Variables (variabili modello) nella scheda YAML.
  - g. Selezionare la casella prompt all'avvio nel campo Job Tags.
  - h. Fare clic su Salva.
2. Avviare il modello di lavoro.
  - a. Accedere a risorse → modelli.
  - b. Fare clic sul modello desiderato, quindi fare clic su Launch (Avvia).
  - c. Quando richiesto all'avvio per Job Tags, digitare requirements\_config. Potrebbe essere necessario fare clic sulla riga Create Job Tag sotto requirements\_config per inserire il tag del processo.



requirements\_config garantisce di disporre delle librerie corrette per eseguire gli altri ruoli.

1. Fare clic su Avanti, quindi su Avvia per avviare il processo.
2. Fare clic su View → Jobs (Visualizza lavori) per monitorare l'output e l'avanzamento del lavoro.
3. Quando richiesto all'avvio per Job Tags, digitare ontap\_config. Potrebbe essere necessario fare clic sulla riga Create "Job Tag" (Crea tag lavoro) sotto ontap\_config per inserire il tag del lavoro.
4. Fare clic su Avanti, quindi su Avvia per avviare il processo.
5. Fare clic su View → Jobs (Visualizza lavori) per monitorare l'output e l'avanzamento del lavoro
6. Una volta completato il ruolo ontap\_CONFIG, eseguire nuovamente il processo per linux\_CONFIG.
7. Accedere a risorse → modelli.

8. Selezionare il modello desiderato, quindi fare clic su Launch (Avvia).
9. Quando richiesto all'avvio per il tipo di tag del processo in linux\_config, potrebbe essere necessario selezionare la riga Create "job tag" (Crea tag del processo) sotto linux\_config per inserire il tag del processo.
10. Fare clic su Avanti, quindi su Avvia per avviare il processo.
11. Selezionare Visualizza → lavori per monitorare l'output e l'avanzamento del lavoro.
12. Una volta completato il ruolo linux\_config, eseguire nuovamente il processo per oracle\_config.
13. Accedere a risorse → modelli.
14. Selezionare il modello desiderato, quindi fare clic su Launch (Avvia).
15. Quando richiesto all'avvio per Job Tags, digitare oracle\_config. Potrebbe essere necessario selezionare la riga Create "Job Tag" (Crea tag lavoro) sotto oracle\_config per inserire il tag lavoro.
16. Fare clic su Avanti, quindi su Avvia per avviare il processo.
17. Selezionare Visualizza → lavori per monitorare l'output e l'avanzamento del lavoro.

## 6. Implementare database aggiuntivi sullo stesso host Oracle

La parte Oracle del playbook crea un singolo database container Oracle su un server Oracle per ogni esecuzione. Per creare ulteriori database container sullo stesso server, attenersi alla seguente procedura.

1. Rivedere le variabili host\_vars.
  - a. Tornare al passaggio 2 - Configurazione di Oracle host\_vars.
  - b. Modificare il SID Oracle con una stringa di denominazione diversa.
  - c. Modificare la porta del listener con un numero diverso.
  - d. Modificare la porta EM Express con un numero diverso se si installa EM Express.
  - e. Copiare e incollare le variabili host riviste nel campo Oracle host Variables (variabili host Oracle) nella scheda host Configuration Detail (Dettagli configurazione host).
2. Avviare il modello di processo di implementazione con solo il tag oracle\_config.
3. Accedere al server Oracle come utente oracle ed eseguire i seguenti comandi:

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



In questo modo verranno elencati i processi oracle se l'installazione è stata completata come previsto e oracle DB è stato avviato

4. Accedere al database per controllare le impostazioni di configurazione del db e i PDB creati con i seguenti set di comandi.

```

[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

```

Ciò conferma che DNFS funziona correttamente.

5. Connettersi al database tramite listener per controllare la configurazione del listener Oracle con il seguente comando. Passare alla porta del listener e al nome del servizio database appropriati.



```
[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
```

Ciò conferma che Oracle listener funziona correttamente.

### Dove cercare aiuto?

Se hai bisogno di aiuto con il toolkit, iscriviti a ["La community di NetApp Solution Automation supporta il canale slack"](#) e cerca il canale di automazione della soluzione per inviare domande o domande.

### Procedura di implementazione passo-passo

Il presente documento descrive in dettaglio l'implementazione di Oracle 19c utilizzando l'interfaccia a riga di comando (cli) di automazione.

#### Implementazione CLI Database Oracle 19c

Questa sezione descrive i passaggi necessari per preparare e implementare il database Oracle19c con la CLI. Assicurarsi di aver esaminato il ["Guida introduttiva e sezione sui requisiti"](#) e preparò il tuo ambiente di conseguenza.

### Scarica Oracle19c repo

1. Dal controller ansible, esegui il seguente comando:

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

2. Dopo aver scaricato il repository, modificare le directory in na\_oracle19c\_deploy <cd na\_oracle19c\_deploy>.

## Modificare il file hosts

Prima dell'implementazione, completare le seguenti operazioni:

1. Modificare la directory `na_oracle19c_deploy` del file `hosts`.
2. In `[ONTAP]` (indirizzo IP), modificare l'indirizzo IP in base all'IP di gestione del cluster.
3. Nel gruppo `[oracle]`, aggiungere i nomi degli host oracle. Il nome host deve essere risolto nel relativo indirizzo IP tramite DNS o il file `hosts`, oppure deve essere specificato nell'host.
4. Una volta completata questa procedura, salvare le modifiche.

Il seguente esempio illustra un file host:

```
#ONTAP Host

[ontap]

"10.61.184.183"

#Oracle hosts

[oracle]

"rtpora01"

"rtpora02"
```

Questo esempio esegue il playbook e implementa oracle 19c su due server oracle DB contemporaneamente. È inoltre possibile eseguire il test con un solo server DB. In tal caso, è necessario configurare un solo file di variabili host.



Il playbook viene eseguito allo stesso modo, indipendentemente dal numero di host e database Oracle implementati.

## Modificare il file `host_name.yml` in `host_vars`

Ciascun host Oracle dispone di un file di variabili host identificato dal nome host che contiene variabili specifiche dell'host. È possibile specificare qualsiasi nome per l'host. Modificare e copiare `host_vars` Dalla sezione host VARS Config (Configurazione VAR host) e incollarla nel file desiderato `host_name.yml` file.



Gli elementi in blu devono essere modificati in base all'ambiente in uso.

## Config. VAR 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"}
```

## Modificare il file vars.yml

Il vars.yml File consolida tutte le variabili specifiche dell'ambiente (ONTAP, Linux o Oracle) per l'implementazione Oracle.

1. Modificare e copiare le variabili dalla sezione VAR e incollarle nel vars.yml file.

```
#####
##### 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"

```

## Eeguire il manuale

Dopo aver completato i prerequisiti di ambiente richiesti e aver copiato le variabili in `vars.yml` e `your_host.yml`, ora sei pronto per implementare i playbook.



<username> deve essere modificato in base all'ambiente in uso.

1. Avvia la guida ONTAP inserendo i tag corretti e il nome utente del cluster ONTAP. Immettere la password per il cluster ONTAP e vsadmin quando richiesto.

```

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

```

2. Eseguire il playbook Linux per eseguire la parte di distribuzione di Linux. Immettere la password admin ssh e sudo.

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

3. Eseguì il manuale Oracle per eseguire la parte relativa all'implementazione di Oracle. Immettere la password admin ssh e sudo.

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

### Implementare database aggiuntivi sullo stesso host Oracle

La parte Oracle del playbook crea un singolo database container Oracle su un server Oracle per ogni esecuzione. Per creare un database container aggiuntivo sullo stesso server, attenersi alla seguente procedura:

1. Rivedere le variabili host\_vars.
  - a. Tornare al passaggio 3 - modificare host\_name.yml file sotto host\_vars.
  - b. Modificare il SID Oracle con una stringa di denominazione diversa.
  - c. Modificare la porta del listener con un numero diverso.
  - d. Modificare la porta EM Express con un numero diverso se è stato installato EM Express.
  - e. Copiare e incollare le variabili host riviste nel file delle variabili host Oracle in host\_vars.
2. Eseguire il manuale con oracle\_config contrassegnare come illustrato nella [Eseguire il manuale](#).

### Convalidare l'installazione di Oracle

1. Accedere al server Oracle come utente oracle ed eseguire i seguenti comandi:

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



In questo modo verranno elencati i processi oracle se l'installazione è stata completata come previsto e oracle DB è stato avviato

2. Accedere al database per controllare le impostazioni di configurazione del db e i PDB creati con i seguenti set di comandi.

```

[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

```

Ciò conferma che DNFS funziona correttamente.

3. Connettersi al database tramite listener per controllare la configurazione del listener Oracle con il seguente comando. Passare alla porta del listener e al nome del servizio database appropriati.



```
[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
```

Ciò conferma che Oracle listener funziona correttamente.

### Dove cercare aiuto?

Se hai bisogno di aiuto con il toolkit, iscriviti a ["La community di NetApp Solution Automation supporta il canale slack"](#) e cerca il canale di automazione della soluzione per inviare domande o domande.

## Panoramica della soluzione

### Protezione automatica dei dati per database Oracle

Le organizzazioni stanno automatizzando i propri ambienti per ottenere efficienze, accelerare le implementazioni e ridurre l'impegno manuale. I tool di gestione della configurazione come Ansible vengono utilizzati per ottimizzare le operazioni dei database aziendali. In questa soluzione, dimostreremo come utilizzare Ansible per automatizzare la protezione dei dati di Oracle con NetApp ONTAP. Consentendo agli amministratori dello storage, agli amministratori di sistema e ai DBA di configurare in modo rapido e coerente la replica dei dati in un data center offsite o nel cloud pubblico, otterrete i seguenti vantaggi:

- Elimina le complessità di progettazione e gli errori umani e implementa un'implementazione coerente e ripetibile e Best practice
- Riduzione dei tempi di configurazione della replica Intercluster, dell'istanza CVO e del ripristino dei database Oracle
- Aumenta la produttività di amministratori di database, sistemi e amministratori dello storage
- Fornisce un workflow di recovery del database per semplificare il test di uno scenario di DR.

NetApp offre ai clienti i moduli e i ruoli Ansible validati per accelerare l'implementazione, la configurazione e la

gestione del ciclo di vita del tuo ambiente di database Oracle. Questa soluzione fornisce istruzioni e codice del playbook Ansible per aiutarti a:

### On Prem to on premise Replication

- Creare Lifs di intercluster su origine e destinazione
- Stabilire il peering di cluster e vserver
- Creare e inizializzare SnapMirror dei volumi Oracle
- Creare una pianificazione di replica tramite AWX/Tower per file binari, database e registri Oracle
- Ripristinare Oracle DB sulla destinazione e portare il database online

### On Prem to CVO in AWS

- Creare AWS Connector
- Creare un'istanza CVO in AWS
- Aggiungere il cluster on-premise a Cloud Manager
- Creazione di lifs tra cluster sull'origine
- Stabilire il peering di cluster e vserver
- Creare e inizializzare SnapMirror dei volumi Oracle
- Creare una pianificazione di replica tramite AWX/Tower per file binari, database e registri Oracle
- Ripristinare Oracle DB sulla destinazione e portare il database online

Una volta pronti, fare clic su ["qui per iniziare con la soluzione"](#).

## Per iniziare

Questa soluzione è stata progettata per essere eseguita in un ambiente AWX/Tower.

### AWX/Tower

Per gli ambienti AWX/tower, viene fornita una guida alla creazione di un inventario della gestione del cluster ONTAP e del server Oracle (IP e nomi host), alla creazione di credenziali, alla configurazione di un progetto che estrae il codice Ansible da NetApp Automation Github e al modello di lavoro che avvia l'automazione.

1. La soluzione è stata progettata per essere eseguita in uno scenario di cloud privato (da on-premise a on-premise) e in un cloud ibrido (da on-premise a cloud pubblico Cloud Volumes ONTAP [CVO])
2. Compilare le variabili specifiche del proprio ambiente, quindi copiarle e incollarle nei campi Extra Vars del modello di lavoro.
3. Una volta aggiunti i var aggiuntivi al modello di lavoro, è possibile avviare l'automazione.
4. L'automazione viene eseguita in tre fasi (Setup, Replication Schedule for Oracle binaries, Database, Logs e Replication Schedule solo per i registri) e una quarta fase per il ripristino del database in un sito DR.
5. Per istruzioni dettagliate su come ottenere le chiavi e i token necessari per la visita CVO Data Protection ["Raccogliere i prerequisiti per le implementazioni CVO e Connector"](#)

## Requisiti

**<strong class="big"> oN-</strong> <strong>|</strong>**

<b>Ambiente</b>	<b>Requisiti</b>
<b>Ambiente Ansible</b>	AWX/Tower
	Ansible v.2.10 e versioni successive
	Python 3
	Librerie Python - netapp-lib - xmltodict - jmespath
<b>ONTAP</b>	ONTAP versione 9.8 +
	Due aggregati di dati
	VLAN NFS e ifgrp create
<b>Server Oracle</b>	RHEL 7/8
	Oracle Linux 7/8
	Interfacce di rete per NFS, gestione pubblica e opzionale
	Ambiente Oracle esistente on-source e sistema operativo Linux equivalente a destinazione (sito DR o cloud pubblico)

**<strong class="big"> </strong>**

<b>Ambiente</b>	<b>Requisiti</b>
<b>Ambiente Ansible</b>	AWX/Tower
	Ansible v.2.10 e versioni successive
	Python 3
	Librerie Python - netapp-lib - xmltodict - jmespath
<b>ONTAP</b>	ONTAP versione 9.8 +
	Due aggregati di dati
	VLAN NFS e ifgrp create
<b>Server Oracle</b>	RHEL 7/8
	Oracle Linux 7/8
	Interfacce di rete per NFS, gestione pubblica e opzionale
	Ambiente Oracle esistente on-source e sistema operativo Linux equivalente a destinazione (sito DR o cloud pubblico)
	Impostare lo spazio di swap appropriato sull'istanza Oracle EC2, per impostazione predefinita alcune istanze EC2 sono implementate con 0 swap
<b>Cloud Manager/AWS</b>	Chiave segreta/accesso AWS
	NetApp Cloud Manager
	Token di aggiornamento di NetApp Cloud Manager



## **oN**

Questa implementazione automatica è progettata con un singolo playbook Ansible che consiste di tre ruoli separati. I ruoli sono per le configurazioni ONTAP, Linux e Oracle. La seguente tabella descrive le attività automatizzate.

<b>Playbook</b>	<b>Attività</b>
<b>ontap_setup</b>	Verifica preliminare dell'ambiente ONTAP
	Creazione di LIF Intercluster sul cluster di origine (OPZIONALE)
	Creazione di LIF Intercluster sul cluster di destinazione (OPZIONALE)
	Creazione del peering di cluster e SVM
	Creazione di SnapMirror di destinazione e inizializzazione dei volumi Oracle designati
<b>ora_replication_cg</b>	Abilitare la modalità di backup per ogni database in /etc/oratab
	Snapshot dei volumi Oracle Binary e Database
	SnapMirror aggiornato
	Disattivare la modalità di backup per ogni database in /etc/oratab
<b>ora_replication_log</b>	Cambiare il log corrente per ogni database in /etc/oratab
	Snapshot del volume Oracle Log
	SnapMirror aggiornato
<b>ora_recovery</b>	Interrompere SnapMirror
	Abilitare NFS e creare un percorso di giunzione per i volumi Oracle sulla destinazione
	Configurare l'host Oracle DR
	Montare e verificare i volumi Oracle
	Ripristinare e avviare il database Oracle

## 

Questa implementazione automatica è progettata con un singolo playbook Ansible che consiste di tre ruoli separati. I ruoli sono per le configurazioni ONTAP, Linux e Oracle. La seguente tabella descrive le attività automatizzate.

Playbook	Attività
<b>cvo_setup</b>	Verifica preliminare dell'ambiente
	AWS Configure/AWS Access Key ID/Secret Key/Default Region
	Creazione del ruolo AWS
	Creazione dell'istanza di NetApp Cloud Manager Connector in AWS
	Creazione dell'istanza CVO (Cloud Volumes ONTAP) in AWS
	Aggiungere il cluster ONTAP di origine on-premise a NetApp Cloud Manager
	Creazione di SnapMirror di destinazione e inizializzazione dei volumi Oracle designati
<b>ora_replication_cg</b>	Abilitare la modalità di backup per ogni database in /etc/oratab
	Snapshot dei volumi Oracle Binary e Database
	SnapMirror aggiornato
	Disattivare la modalità di backup per ogni database in /etc/oratab
<b>ora_replication_log</b>	Cambiare il log corrente per ogni database in /etc/oratab
	Snapshot del volume Oracle Log
	SnapMirror aggiornato
<b>ora_recovery</b>	Interrompere SnapMirror
	Abilitare NFS e creare un percorso di giunzione per i volumi Oracle sul CVO di destinazione
	Configurare l'host Oracle DR
	Montare e verificare i volumi Oracle
	Ripristinare e avviare il database Oracle

## Parametri predefiniti

Per semplificare l'automazione, abbiamo preimpostato molti parametri Oracle richiesti con valori predefiniti. In genere non è necessario modificare i parametri predefiniti per la maggior parte delle implementazioni. Un utente più avanzato può apportare modifiche ai parametri predefiniti con cautela. I parametri predefiniti si trovano in ogni cartella di ruoli nella directory dei valori predefiniti.

## Licenza

Leggere le informazioni sulla licenza come indicato nel repository Github. Accedendo, scaricando, installando o utilizzando il contenuto di questo repository, l'utente accetta i termini della licenza stabilita ["qui"](#).

Si noti che esistono alcune limitazioni relative alla produzione e/o alla condivisione di qualsiasi opera derivata con il contenuto di questo repository. Leggere attentamente i termini del ["Licenza"](#) prima di utilizzare il contenuto. Se non si accettano tutti i termini, non accedere, scaricare o utilizzare il contenuto di questo repository.

Una volta pronti, fare clic su ["Qui per le procedure AWX/Tower dettagliate"](#).

# Procedura di implementazione passo-passo

## Protezione dei dati Oracle AWX/Tower

### Crea l'inventario, il gruppo, gli host e le credenziali per il tuo ambiente

Questa sezione descrive la configurazione di inventario, gruppi, host e credenziali di accesso in AWX/Ansible Tower che preparano l'ambiente per l'utilizzo delle soluzioni automatizzate di NetApp.

1. Configurare l'inventario.
  - a. Accedere a Resources → Inventories → Add e fare clic su Add Inventory (Aggiungi inventario).
  - b. Fornire il nome e i dettagli dell'organizzazione, quindi fare clic su Save (Salva).
  - c. Nella pagina Inventories (inventari), fare clic sull'inventario creato.
  - d. Accedere al sottomenu Groups (gruppi) e fare clic su Add (Aggiungi).
  - e. Fornire il nome oracle per il primo gruppo e fare clic su Save (Salva).
  - f. Ripetere la procedura per un secondo gruppo denominato dr\_oracle.
  - g. Selezionare il gruppo oracle creato, accedere al sottomenu hosts e fare clic su Add New host (Aggiungi nuovo host).
  - h. Fornire l'indirizzo IP dell'IP di gestione dell'host Oracle di origine e fare clic su Save (Salva).
  - i. Questo processo deve essere ripetuto per il gruppo dr\_oracle e deve essere aggiunto l'IP/nome host di gestione dell'host DR/destinazione Oracle.



Di seguito sono riportate le istruzioni per la creazione dei tipi di credenziale e delle credenziali on-premise con ONTAP o CVO su AWS.

## On-Prem

1. Configurare le credenziali.
2. Creare tipi di credenziale. Per le soluzioni che utilizzano ONTAP, è necessario configurare il tipo di credenziale in modo che corrisponda alle voci di nome utente e password.
  - a. Accedere a Administration → Credential Types (Amministrazione tipi di credenziali) e fare clic su Add (Aggiungi).
  - b. Fornire il nome e la descrizione.
  - c. Incollare il seguente contenuto in Input Configuration (Configurazione input):

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

- d. Incollare il seguente contenuto in Injector Configuration (Configurazione iniettore), quindi fare clic su Save (Salva):

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

3. Crea credenziale per ONTAP
  - a. Accedere a Resources → Credentials (risorse credenziali) e fare clic su Add (Aggiungi).
  - b. Immettere il nome e i dettagli dell'organizzazione per le credenziali ONTAP
  - c. Selezionare il tipo di credenziale creato nel passaggio precedente.
  - d. In Dettagli tipo, immettere il nome utente e la password per i cluster di origine e di destinazione.
  - e. Fare clic su Salva
4. Crea credenziale per Oracle



- a. Accedere a Resources → Credentials (risorse credenziali) e fare clic su Add (Aggiungi).
- b. Immettere il nome e i dettagli dell'organizzazione per Oracle
- c. Selezionare il tipo di credenziale Machine.
- d. In Dettagli tipo, immettere il nome utente e la password per gli host Oracle.
- e. Selezionare il metodo corretto di escalation dei privilegi e immettere il nome utente e la password.
- f. Fare clic su Salva
- g. Ripetere la procedura se necessario per una credenziale diversa per l'host dr\_oracle.

## **CVO**

1. Configurare le credenziali.
2. Creare tipi di credenziale. Per le soluzioni che coinvolgono ONTAP, devi configurare il tipo di credenziale in modo che corrisponda alle voci di nome utente e password, aggiungeremo anche le voci per Cloud Central e AWS.
  - a. Accedere a Administration → Credential Types (Amministrazione tipi di credenziali) e fare clic su Add (Aggiungi).
  - b. Fornire il nome e la descrizione.
  - c. Incollare il seguente contenuto in Input Configuration (Configurazione input):

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

d. Incollare il seguente contenuto in Injector Configuration (Configurazione iniettore) e fare clic su

Save (Salva):

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

### 3. Crea credenziale per ONTAP/CVO/AWS

- a. Accedere a Resources → Credentials (risorse credenziali) e fare clic su Add (Aggiungi).
- b. Immettere il nome e i dettagli dell'organizzazione per le credenziali ONTAP
- c. Selezionare il tipo di credenziale creato nel passaggio precedente.
- d. In Dettagli tipo, immettere il nome utente e la password per i cluster di origine e CVO, Cloud Central/Manager, AWS Access/Secret Key e Cloud Central Refresh Token.
- e. Fare clic su Salva

### 4. Crea credenziale per Oracle (origine)

- a. Accedere a Resources → Credentials (risorse credenziali) e fare clic su Add (Aggiungi).
- b. Immettere il nome e i dettagli dell'organizzazione per l'host Oracle
- c. Selezionare il tipo di credenziale Machine.
- d. In Dettagli tipo, immettere il nome utente e la password per gli host Oracle.
- e. Selezionare il metodo corretto di escalation dei privilegi e immettere il nome utente e la password.
- f. Fare clic su Salva

### 5. Crea credenziale per destinazione Oracle

- a. Accedere a Resources → Credentials (risorse credenziali) e fare clic su Add (Aggiungi).
- b. Inserire il nome e i dettagli dell'organizzazione dell'host Oracle DR
- c. Selezionare il tipo di credenziale Machine.
- d. In Dettagli tipo, immettere il nome utente (ec2-user o se è stato modificato dall'impostazione predefinita) e la chiave privata SSH
- e. Selezionare il metodo corretto di escalation dei privilegi (sudo) e immettere il nome utente e la password, se necessario.
- f. Fare clic su Salva

## Creare un progetto

1. Accedere a risorse → progetti e fare clic su Aggiungi.
  - a. Inserire il nome e i dettagli dell'organizzazione.
  - b. Selezionare Git nel campo Source Control Credential Type (tipo credenziale controllo origine).
  - c. invio <[https://github.com/NetApp-Automation/na\\_oracle19c\\_data\\_protection.git](https://github.com/NetApp-Automation/na_oracle19c_data_protection.git)> Come URL del controllo di origine.
  - d. Fare clic su Salva.
  - e. Potrebbe essere necessario sincronizzare il progetto occasionalmente quando il codice sorgente cambia.

## Configurare le variabili globali

Le variabili definite in questa sezione si applicano a tutti gli host Oracle, ai database e al cluster ONTAP.

1. Inserire i parametri specifici dell'ambiente nel seguente formato vars o variabili globali incorporate.



Gli elementi in blu devono essere modificati in base all'ambiente in uso.

## On-Prem

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

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

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

create_source_intercluster_lifs: "yes"

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

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

create_destination_intercluster_lifs: "yes"
```

```

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

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

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

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

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

```

```

snapmirror_policy: "async_policy_oracle"

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

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

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

```

## CVO

```

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

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

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

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

#Names of the Nodes in the Destination CVO Cluster

```

```

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

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

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

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

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

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

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

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

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

```



```

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

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

# Name your connector.
connector_name: "awx_connector"

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

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

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

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

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

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

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

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

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

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

```

```

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

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

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

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

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

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

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

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

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

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

```

```

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

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

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

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

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

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

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

```

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

### **Playbook per l'automazione**

È necessario eseguire quattro playbook separati.

1. Playbook per la configurazione del tuo ambiente, on-premise o CVO.
2. Playbook per la replica di file binari e database Oracle in base a una pianificazione
3. Playbook per la replica dei registri Oracle in base a una pianificazione
4. Playbook per il ripristino del database su un host di destinazione

## Setup ONTAP/CVO

Configurazione ONTAP e CVO

### Configurare e avviare il modello di lavoro.

1. Creare il modello di lavoro.
  - a. Accedere a risorse → modelli → Aggiungi e fare clic su Aggiungi modello di processo.
  - b. Immettere il nome ONTAP/CVO Setup
  - c. Selezionare il tipo di lavoro; Esegui consente di configurare il sistema in base a una guida.
  - d. Seleziona l'inventario, il progetto, il playbook e le credenziali corrispondenti per il playbook.
  - e. Selezionare il playbook `ontap_setup.yml` per un ambiente on-Prem oppure selezionare `cvo_setup.yml` per la replica su un'istanza CVO.
  - f. Incollare le variabili globali copiate dal passaggio 4 nel campo Template Variables (variabili modello) nella scheda YAML.
  - g. Fare clic su Salva.
2. Avviare il modello di lavoro.
  - a. Accedere a risorse → modelli.
  - b. Fare clic sul modello desiderato, quindi fare clic su Launch (Avvia).



Utilizzeremo questo modello e lo copieremo per gli altri playbook.

### Replica per volumi binari e database

Pianificazione del manuale di replica binario e database

### Configurare e avviare il modello di lavoro.

1. Copiare il modello di lavoro creato in precedenza.
  - a. Accedere a risorse → modelli.
  - b. Individuare il modello di installazione di ONTAP/CVO e fare clic con il pulsante destro del mouse su Copy Template (Copia modello)
  - c. Fare clic su Edit Template (Modifica modello) nel modello copiato e modificare il nome in Binary and Database Replication Playbook (Playbook di replica binario e database).
  - d. Mantenere lo stesso inventario, progetto e credenziali per il modello.
  - e. Selezionare `ora_Replication_cg.yml` come manuale da eseguire.
  - f. Le variabili rimarranno le stesse, ma l'IP del cluster CVO dovrà essere impostato nella variabile `dst_cluster_ip`.
  - g. Fare clic su Salva.
2. Pianificare il modello di lavoro.
  - a. Accedere a risorse → modelli.
  - b. Fare clic sul modello Playbook di replica binario e database, quindi fare clic su Pianificazioni nella parte superiore del set di opzioni.
  - c. Fare clic su Add (Aggiungi), add Name Schedule (Aggiungi pianificazione nome) per la replica binaria e del database, scegliere la data/ora di inizio all'inizio dell'ora, scegliere il fuso orario

locale e la frequenza di esecuzione. La frequenza di esecuzione sarà spesso la replica di SnapMirror verrà aggiornata.



Verrà creata una pianificazione separata per la replica del volume Log, in modo che possa essere replicata con cadenza più frequente.

### Replica per i volumi di log

Pianificazione del Playbook di replica del registro

#### Configurare e avviare il modello di lavoro.

1. Copiare il modello di lavoro creato in precedenza.
  - a. Accedere a risorse → modelli.
  - b. Individuare il modello di installazione di ONTAP/CVO e fare clic con il pulsante destro del mouse su Copy Template (Copia modello)
  - c. Fare clic su Edit Template (Modifica modello) sul modello copiato e modificare il nome in Log Replication Playbook (Playbook replica registro).
  - d. Mantenere lo stesso inventario, progetto e credenziali per il modello.
  - e. Selezionare ora\_Replication\_logs.yml come manuale da eseguire.
  - f. Le variabili rimarranno le stesse, ma l'IP del cluster CVO dovrà essere impostato nella variabile `dst_cluster_ip`.
  - g. Fare clic su Salva.
2. Pianificare il modello di lavoro.
  - a. Accedere a risorse → modelli.
  - b. Fare clic sul modello Log Replication Playbook, quindi fare clic su Schedules (Pianificazioni) nella parte superiore del set di opzioni.
  - c. Fare clic su Add (Aggiungi), Add Name Schedule (Aggiungi pianificazione nome) per Log Replication (replica registro), scegliere Start date/time (Data/ora di inizio) all'inizio dell'ora, scegliere il fuso orario locale e la frequenza di esecuzione. La frequenza di esecuzione sarà spesso la replica di SnapMirror verrà aggiornata.



Si consiglia di impostare la pianificazione del registro per l'aggiornamento ogni ora, in modo da garantire il ripristino dell'ultimo aggiornamento orario.

### Ripristinare e ripristinare il database

Pianificazione del Playbook di replica del registro

#### Configurare e avviare il modello di lavoro.

1. Copiare il modello di lavoro creato in precedenza.
  - a. Accedere a risorse → modelli.
  - b. Individuare il modello di installazione di ONTAP/CVO e fare clic con il pulsante destro del mouse su Copy Template (Copia modello)
  - c. Fare clic su Edit Template (Modifica modello) sul modello copiato e modificare il nome in Restore and Recovery Playbook (Guida per il ripristino e il ripristino).

- d. Mantenere lo stesso inventario, progetto e credenziali per il modello.
- e. Selezionare ora\_recovery.yml come manuale da eseguire.
- f. Le variabili rimarranno le stesse, ma l'IP del cluster CVO dovrà essere impostato nella variabile dst\_cluster\_ip.
- g. Fare clic su Salva.



Questo manuale non verrà eseguito fino a quando non si sarà pronti a ripristinare il database nel sito remoto.

## Ripristino del database Oracle

1. Produzione on-premise i volumi di dati dei database Oracle sono protetti tramite la replica di NetApp SnapMirror su un cluster ONTAP ridondante nel data center secondario o su Cloud Volume ONTAP nel cloud pubblico. In un ambiente di disaster recovery completamente configurato, le istanze di calcolo del recovery nel data center secondario o nel cloud pubblico sono in standby e pronte per il ripristino del database di produzione in caso di disastro. Le istanze di calcolo in standby vengono mantenute in sincronia con le istanze on-premise eseguendo aggiornamenti di parallel sulla patch del kernel del sistema operativo o aggiornando in un passo di blocco.
2. In questa soluzione dimostrata, il volume binario Oracle viene replicato sulla destinazione e montato sull'istanza di destinazione per richiamare lo stack software Oracle. Questo approccio per il ripristino di Oracle ha un vantaggio rispetto a una nuova installazione di Oracle all'ultimo momento in cui si è verificato un disastro. Garantisce che l'installazione di Oracle sia completamente sincronizzata con l'installazione del software di produzione on-premise, con i livelli di patch e così via. Tuttavia, questo potrebbe avere o meno ulteriori implicazioni di licenza software per il volume binario Oracle replicato nel sito di recovery, a seconda di come è strutturato il licensing software con Oracle. Si consiglia all'utente di verificare con il proprio personale addetto alle licenze software per valutare il potenziale requisito di licenza Oracle prima di decidere di utilizzare lo stesso approccio.
3. L'host Oracle di standby nella destinazione viene configurato con le configurazioni dei prerequisiti Oracle.
4. Gli SnapMirror sono rotti e i volumi sono resi scrivibili e montati sull'host Oracle di standby.
5. Il modulo di ripristino Oracle esegue le seguenti attività per il ripristino e l'avvio di Oracle nel sito di ripristino dopo che tutti i volumi DB sono stati montati nell'istanza di calcolo in standby.
  - a. Sincronizza il file di controllo: Abbiamo implementato file di controllo Oracle duplicati su diversi volumi di database per proteggere file di controllo critici del database. Uno si trova sul volume di dati e l'altro sul volume di log. Poiché i volumi di dati e log vengono replicati con frequenza diversa, al momento del ripristino non saranno sincronizzati.
  - b. Relink Oracle binary: Poiché il binario Oracle viene trasferito in un nuovo host, è necessario un relink.
  - c. Ripristino del database Oracle: Il meccanismo di recovery recupera l'ultimo numero di modifica del sistema nell'ultimo log archiviato disponibile nel volume di log Oracle dal file di controllo e ripristina il database Oracle per recuperare tutte le transazioni aziendali che sono state replicate nel sito di DR al momento dell'errore. Il database viene quindi avviato in una nuova incarnazione per portare avanti le connessioni utente e le transazioni di business nel sito di recovery.



Prima di eseguire il playbook di ripristino, assicurarsi di disporre di quanto segue: Assicurarsi che venga copiato su /etc/oratab e /etc/orainst.loc dall'host Oracle di origine all'host di destinazione

# TR-4794: Database Oracle su NetApp EF-Series

Mitch Blackburn, Ebin Kadavy, NetApp

TR-4794 è stato progettato per aiutare gli amministratori dello storage e i database a implementare con successo Oracle sullo storage NetApp EF-Series.

["TR-4794: Database Oracle su NetApp EF-Series"](#)



## Informazioni sul copyright

Copyright © 2024 NetApp, Inc. Tutti i diritti riservati. Stampato negli Stati Uniti d'America. Nessuna porzione di questo documento soggetta a copyright può essere riprodotta in qualsiasi formato o mezzo (grafico, elettronico o meccanico, inclusi fotocopie, registrazione, nastri o storage in un sistema elettronico) senza previo consenso scritto da parte del detentore del copyright.

Il software derivato dal materiale sottoposto a copyright di NetApp è soggetto alla seguente licenza e dichiarazione di non responsabilità:

IL PRESENTE SOFTWARE VIENE FORNITO DA NETAPP "COSÌ COM'È" E SENZA QUALSIVOGLIA TIPO DI GARANZIA IMPLICITA O ESPRESSA FRA CUI, A TITOLO ESEMPLIFICATIVO E NON ESAUSTIVO, GARANZIE IMPLICITE DI COMMERCIALIZZABILITÀ E IDONEITÀ PER UNO SCOPO SPECIFICO, CHE VENGONO DECLINATE DAL PRESENTE DOCUMENTO. NETAPP NON VERRÀ CONSIDERATA RESPONSABILE IN ALCUN CASO PER QUALSIVOGLIA DANNO DIRETTO, INDIRETTO, ACCIDENTALE, SPECIALE, ESEMPLARE E CONSEGUENZIALE (COMPRESI, A TITOLO ESEMPLIFICATIVO E NON ESAUSTIVO, PROCUREMENT O SOSTITUZIONE DI MERCI O SERVIZI, IMPOSSIBILITÀ DI UTILIZZO O PERDITA DI DATI O PROFITTI OPPURE INTERRUZIONE DELL'ATTIVITÀ AZIENDALE) CAUSATO IN QUALSIVOGLIA MODO O IN RELAZIONE A QUALUNQUE TEORIA DI RESPONSABILITÀ, SIA ESSA CONTRATTUALE, RIGOROSA O DOVUTA A INSOLVENZA (COMPRESA LA NEGLIGENZA O ALTRO) INSORTA IN QUALSIASI MODO ATTRAVERSO L'UTILIZZO DEL PRESENTE SOFTWARE ANCHE IN PRESENZA DI UN PREAVVISO CIRCA L'EVENTUALITÀ DI QUESTO TIPO DI DANNI.

NetApp si riserva il diritto di modificare in qualsiasi momento qualunque prodotto descritto nel presente documento senza fornire alcun preavviso. NetApp non si assume alcuna responsabilità circa l'utilizzo dei prodotti o materiali descritti nel presente documento, con l'eccezione di quanto concordato espressamente e per iscritto da NetApp. L'utilizzo o l'acquisto del presente prodotto non comporta il rilascio di una licenza nell'ambito di un qualche diritto di brevetto, marchio commerciale o altro diritto di proprietà intellettuale di NetApp.

Il prodotto descritto in questa guida può essere protetto da uno o più brevetti degli Stati Uniti, esteri o in attesa di approvazione.

LEGENDA PER I DIRITTI SOTTOPOSTI A LIMITAZIONE: l'utilizzo, la duplicazione o la divulgazione da parte degli enti governativi sono soggetti alle limitazioni indicate nel sottoparagrafo (b)(3) della clausola Rights in Technical Data and Computer Software del DFARS 252.227-7013 (FEB 2014) e FAR 52.227-19 (DIC 2007).

I dati contenuti nel presente documento riguardano un articolo commerciale (secondo la definizione data in FAR 2.101) e sono di proprietà di NetApp, Inc. Tutti i dati tecnici e il software NetApp forniti secondo i termini del presente Contratto sono articoli aventi natura commerciale, sviluppati con finanziamenti esclusivamente privati. Il governo statunitense ha una licenza irrevocabile limitata, non esclusiva, non trasferibile, non cedibile, mondiale, per l'utilizzo dei Dati esclusivamente in connessione con e a supporto di un contratto governativo statunitense in base al quale i Dati sono distribuiti. Con la sola esclusione di quanto indicato nel presente documento, i Dati non possono essere utilizzati, divulgati, riprodotti, modificati, visualizzati o mostrati senza la previa approvazione scritta di NetApp, Inc. I diritti di licenza del governo degli Stati Uniti per il Dipartimento della Difesa sono limitati ai diritti identificati nella clausola DFARS 252.227-7015(b) (FEB 2014).

## Informazioni sul marchio commerciale

NETAPP, il logo NETAPP e i marchi elencati alla pagina <http://www.netapp.com/TM> sono marchi di NetApp, Inc. Gli altri nomi di aziende e prodotti potrebbero essere marchi dei rispettivi proprietari.