COHESITY

Version 1.3

July 2024

Streamline Oracle Database Protection Using Cohesity's Oracle Adapter

Table of Contents

About This Guide 4
Intended Audience 4
Cohesity Oracle Adapter Overview 5
Cohesity Oracle Adapter Benefits 5
Cohesity Platform Benefits 7
Cohesity Platform Architecture 8
Cohesity Oracle Adapter Communication 9
Cohesity Oracle Backup Workflow 9
Cohesity Oracle Adapter Requirements 11
Required Credentials and Privileges 11
OS User Authentication 11
DB Authentication 12
Recommended Approach 13
Install the Cohesity Oracle Adapter 14
Install the Cohesity Agent 14
Install the Cohesity Linux Agent 14
Install the Cohesity AIX Agent 15
Register the Oracle Server 16
Register the Oracle Database Instance 18
Create a Cohesity Oracle Data Protection Job 19
Multi-channel and Multi-node Backup Option 26
Cohesity Oracle Database Recovery 27
Oracle Database Restore Types 27
Multi-channel and Multi-node Recovery Option 29
Granular Recovery 30
Cohesity Oracle SBT Plug-in 33
Oracle ZDLRA Backup and Restore Using SBT 33
Backup Oracle ZDLRA Using Cohesity SBT Plug-in 33

Restore Oracle ZDLRA Restore Using Cohesity SBT Plug-in 34
Disaster Recovery and Business Continuity 35
Replication 35
Cohesity CloudArchive and Retrieval 36
Conclusion 37
Your Feedback 38
About the Authors 38
Document Version History 38

Figures

Figure 1: Cohesity Oracle Adapter Overview 8
Figure 2: Cohesity Platform Communicates with Oracle Databases via Cohesity Oracle Adapter 9
Figure 3: Cohesity Oracle Adapter Backup Workflow 10
Figure 4: Cohesity Oracle Adapter Recovery to Original or New Server 27
Figure 5: Cohesity NFS View Exposed to Original or New Server 30
Figure 6: Oracle ZDLRA Backups Using Cohesity 33
Figure 7: Oracle Database Recovery on ZDLRA Appliance using Cohesity SBT 34
Figure 8: Cohesity Native Replication to New Cluster (on-premises or Cloud Edition) .. 35
Figure 9: Cohesity CloudArchive Connects Cohesity Backups to Cloud Storage 36

Tables

Table 1: Cohesity Oracle Adapter Benefits 6
Table 2: Cohesity Platform Benefits 7
Table 3: OS User Authentication Pros and Cons 12
Table 4: Encryption Types and Their Impacts 23
Table 5: Recovery Page Fields and Definitions 32

About This Guide

This solution guide describes the steps and best practices for protecting Oracle® databases using the Cohesity Oracle Adapter.

Intended Audience

This guide is written for IT and database administrators (DBAs) familiar with data protection of Oracle Databases, and assumes some knowledge of Oracle Database storages, backup and recovery technologies, and a fundamental understanding of Oracle Recovery Manager (RMAN), whether in a physical or virtual environment.

Cohesity also recommend being familiar with:

- Cohesity Platform™
- Cohesity DataProtect™

Cohesity Oracle Adapter Overview

With the ballooning growth in volume of data, the number and size of databases expanding rapidly, and compliance mandates to retain data for longer periods of time, application owners and database administrators face a growing challenge in protecting their databases. It is prohibitively expensive and risks data loss to retain this data on production storage. At the same time, having an effective data protection strategy with fast Recovery Time Objectives (RTOs) and Recovery Point Objectives (RPOs) is paramount, especially for mission-critical applications that can tolerate only minimal downtime.

Cohesity Platform provides a simple, fast, and cost-effective backup and recovery solution for growing Oracle Database environments. Our platform provides several different methods for protecting your Oracle Databases:

- RMAN NFS Target
- RMAN NFS Target with Cohesity Source-Side Deduplication
- Cohesity Oracle Remote Adapter with NFS

See Cohesity and Oracle RMAN Guide for more about RMAN, source-side deduplication, and the Cohesity Oracle Remote Adapter with NFS.

NOTE: The Cohesity Oracle Adapter agent is currently only supported on

Oracle Database versions : 11gR2, 12cR1, 12cR2, and 18C

Operating System : Oracle Enterprise Linux, CentOS, RHEL version 6 & 7, and AIX

Oracle Database configurations supported : Non-RAC, Single Node RAC, Multi-node RAC, and ZDLRA.

The adapter supports Oracle deployments across all storage tiers (Block, NFS) using ASM as volume manager or any block device. The adapter also supports Oracle Database deployed on Exadata and Oracle Database Appliances (ODA).

For more details, see Cohesity's list of supported software.

Cohesity Oracle Adapter Benefits

Cohesity Oracle Adapter delivers an integrated and centralized approach in backup and recovery of Oracle Databases. It simplifies Oracle Database backups without the need for RMAN scripting skills. It provides a single location to schedule Oracle Database backups that can span multiple Oracle Database environments. This allows Cohesity Platform to manage data protection processes and schedules, as well as provide a consolidated log of all activity.

The Cohesity Oracle Adapter natively integrates with Oracle Recovery Manager to provide application-consistent backup and recovery, for both Oracle single-instance deployments and Real Application Clusters (RAC). The adapter brings Cohesity's solutions to the challenges that come with traditional backup and target storage to Oracle single-instance Database (See Table 1):

Table 1: Cohesity Oracle Adapter Benefits

DB BACKUP & STORAGE CHALLENGES	COHESITY SOLUTIONS
Application performance degraded when running disruptive backup agents on production database servers.	Lightweight adapter
Expensive use of primary storage as scratch space for temporary backups before it gets transferred to secondary storage.	Auto tiering and global dedupe/compression brings simplicity and cost efficiencies. Distributed web-scale platform designed for secondary data. Performance and capacity can be scaled linearly simply by adding nodes, eliminating the need for forklift upgrades. The system provides 'always-on' availability.
Complex data-protection environments, including media servers, master servers, agents, and target storage.	Simplified data protection. Replace multiple data protection silos (target storage, media servers, master servers, and cloud gateways) with a single hyper-converged solution for backup, recovery, replication, cloud tiering, archiving, and target storage.
Inability to leverage public cloud for long-term retention.	Native cloud integration with Amazon Web Services (AWS), Microsoft Azure, Oracle Cloud, and Google Cloud Platform (GCP), and in support for long-term retention and tiering of cold data.
RMAN scripts maintenance overhead.	No need to maintain any RMAN scripts.
No single pane of glass to manage Oracle Database data protection.	Single dashboard to manage Oracle Database data protection.
Restores are more time consuming as both Full backups and incrementals backups need to be restored.	Faster restores as every database backup is fully hydrated using Cohesity snapshots and Oracle incremental merge capabilities. Ability to clone DB from a backup without moving data across storage tiers.

Cohesity Platform Benefits

When used in conjunction with Cohesity's Oracle Adapter, Cohesity Platform's architecture delivers unparalleled performance and scale-out capabilities.

Table 2: Cohesity Platform Benefits

FEATURES	DETAILS
Simple data protection	Simplify backup environments by eliminating the need for media servers and master servers.
Application-consistent protection	Cohesity Oracle Adapter transparently leverages RMAN, which allows it to perform application-consistent backups to ensure the database can recover faster and avoid data corruption. The system provides 'always-on' availability.
Distributed platform	Scale out capacity and linear performance simply by adding nodes to the cluster. Eliminate the need for forklift upgrades and data migrations.
Native cloud integration	Integrate with AWS, Microsoft Azure, and/or GCP for long-term archival and data tiering.
Copy data management	Ability to spin up clone database copies from backups for test and dev environments. Cohesity's secondary storage platform acts as an NFS target for Oracle datafiles, control files, and redo logs.
Lower TCO (Total Cost of Ownership)	Hyper-converged solution that consolidates backup software licenses, media and catalog servers, and backup targets. Global deduplication, compression, and snapshots dramatically reduce physical storage usage. Pay-as-you-grow expandability which reduces the need to over-provision.

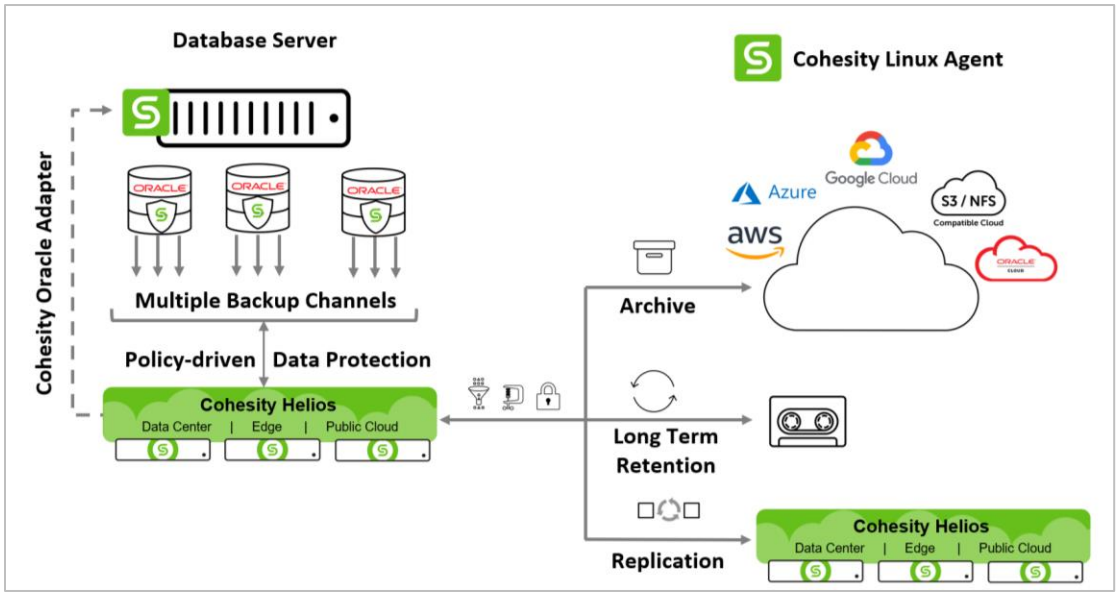
Cohesity Platform Architecture

Cohesity Platform delivers a hyperconverged, web-scale platform that consolidates all secondary storage and data services into a single, efficient solution. Cohesity Platform simplifies data protection, consolidates file and object services, provides instant access to test/dev copies, and performs in-place searches and analytics, all on a software-defined platform that spans from the edge to the cloud.

The core components of the Cohesity Oracle Adapter are:

- Physical or virtual Oracle instance
- Cohesity Platform
- Cohesity Oracle Adapter
- Archive or tiering target at your data center or cloud provider of choice

Figure 1: Cohesity Oracle Adapter Overview



As illustrated in Figure 1, Cohesity Platform can easily be integrated into a new or existing Oracle infrastructure in minutes, by installing the Cohesity Oracle Adapter agent and creating the Protection Policies, jobs, and replication schedules discussed below. Once Oracle Database is protected, it can be seamlessly archived, sent to tape, and/or replicated to another Cohesity Platform (running in the cloud or on-premises).

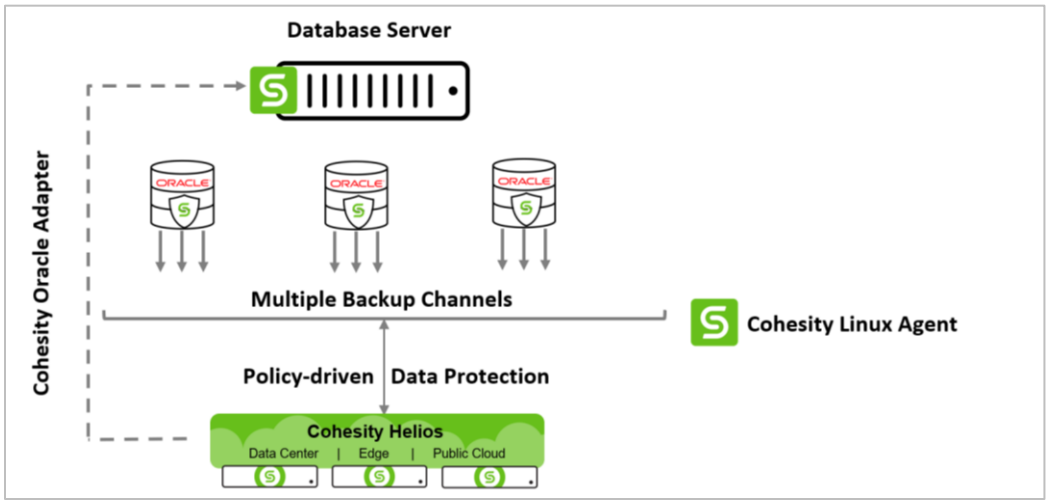
The Cohesity Oracle Adapter uses Oracle's incremental-merge method.

Cohesity Oracle Adapter Communication

Based on the policy settings, the Cohesity Oracle Adapter executes an RMAN backup command to initiate a backup of the Oracle Database. The adapter also automatically determines the optimal number of backup channels, based on the compute resources on the Oracle Database server and the number of nodes in Cohesity Platform. The necessary Cohesity Views are automatically generated (that is, an NFS mount point on the client and an NFS View on Cohesity Platform), where the backups are going to be written.

Communication between Cohesity Platform and the Oracle Databases is managed by the Cohesity Oracle Adapter.

Figure 2: Cohesity Platform Communicates with Oracle Databases via Cohesity Oracle Adapter



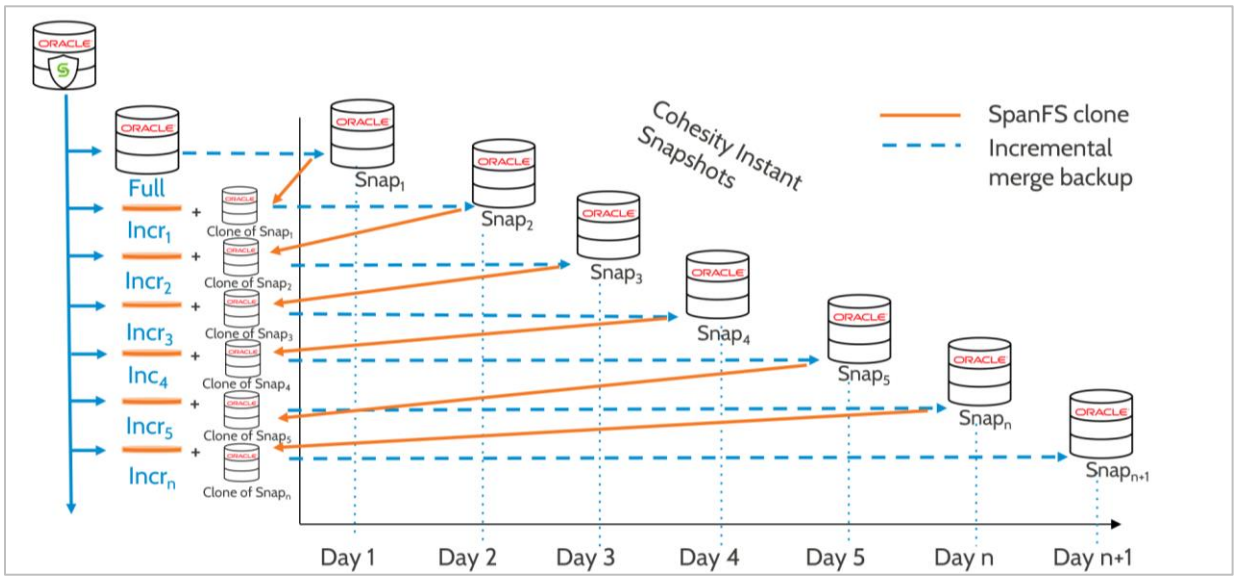
Cohesity Oracle Backup Workflow

Now let's take a walk through each step of Cohesity's backup actions to understand the workflow:

1. The very first backup is executed as a full backup copy, as this is the first backup run. After the initial backup has completed, Cohesity takes an instant snapshot of that initial backup and tags it as *Snap*$_1$ for the Day 1 full.
2. For the next run, the Protection Job retrieves an incremental level-1 backup (*Incr*$_1$) from the database, clones the previous snapshot (*Snap*$_1$), and merges the two to create a new full-image copy of the database (*Snap*$_2$) for the Day 2 full.
3. For each run of the Protection Job after that, it goes through the same process, where a new incremental level-1 backup from the database (*Incr*$_2$), clones the previous snapshot (*Snap*$_2$) and merges the two to create a new full-image copy of the database (*Snap*$_3$) for the Day 3 full.

As we continue to generate backups of the Oracle Database, a pattern emerges: for every backup that is generated, the end result is always a full-image copy of the database, while maintaining a short backup window because it is taking advantage of incremental backups. This means that, for any point in time, we have a full backup of the Oracle Database, as shown in the figure below. In addition to enabling BCT, this enables shorter backup windows and faster recovery of the Oracle Database, which helps improve RTO and RPO.

Figure 3: Cohesity Oracle Adapter Backup Workflow



Cohesity Oracle Adapter Requirements

Once you register your physical and VM servers with Cohesity, Cohesity will discover your Oracle Database automatically, but before they can connect, your user agents must have the appropriate credentials and permissions.

Oracle authentication is set at the instance level by the DBA. Each Oracle instance can have its own authentication type. The Oracle adapter works seamlessly with either OS or database authentication.

Required Credentials and Privileges

You can use either OS user or DB user authentication to connect to the source Oracle DB, but there are differences.

To use the Cohesity Oracle Adapter, you must have the following authentication and privileges:

OS user authentication requirements:

- Sudoers privilege, to give the OS user the ability to mount the exported Cohesity View.
- The OS user must be part of both the ‘oinstall’ and ‘dba’ groups (like the default ‘oracle’ user).
- The OS user must be added to Oracle DB as OS-authenticated using the IDENTIFIED EXTERNALLY clause.

> **NOTE**: OS-based authentication will fail if the OS_AUTHENT_PREFIX parameter is missing or incorrect.

Oracle DB user authentication requirements:

- **In Oracle 11g**: SYSDBA privilege.
- **In Oracle 12c**: Oracle DB users must have a minimum of SYSBACKUP privilege.

OS User Authentication

When you attempt to connect from the local database server using OS Authentication, the OS username is passed to the Oracle server. If the username is recognized, the OS user account connects with ‘sysdba’ privileges. If the username is not recognized, the connection is rejected.

You can authenticate both non-OS and OS users in the same system. Non-OS users are assigned passwords and authenticated by the database.

To authenticate an OS user:

1. Create a user account (separate from the default ‘oracle’ account) that is a member of the ‘oinstall’ and ‘dba’ groups.
2. From Oracle Database **sqlplus** prompt, create a user using the IDENTIFIED EXTERNALLY clause of the CREATE USER statement.
3. Send the OS_AUTHENT_PREFIX initialization parameter to specify the prefix that Oracle Database uses to authenticate users who attempt to connect to the server.

There are several advantages and disadvantages to using OS user authentication:

Table 3: OS User Authentication Pros and Cons

PROS	CONS
Once authenticated by the operating system, users can connect to Oracle Database more conveniently, without specifying a username or password.	A user must have an operating system account on the computer that is accessed. Not all users, particularly non-administrative users, have operating system accounts.
With control over user authentication centralized in the operating system, Oracle Database need not store or manage user passwords, although it still maintains usernames in the database.	If a user has logged in using this method and steps away from the terminal, another user could easily log in because this user does not need any passwords or credentials. This could pose a serious security problem.
Audit trails in the database and operating system can use the same usernames.	When an operating system is used to authenticate database users, managing distributed database environments and database links requires special care. OS-authenticated database links can pose a security weakness. For this reason, Oracle recommends you do not use them.

NOTE: Several corporations have security measures that lock out OS-based authentication.

DB Authentication

Oracle Database authentication uses the Oracle credentials to connect to the database. For DB authentication, you have to create a DB user on each applicable database and then provide those credentials during database registration.

To use Oracle Database authentication, you must provide:

- An Oracle Database user with the right roles to perform backup:
  - **Oracle 11g**: Grant create session, resource, and sysdba permissions to the Oracle backup user.
  - **Oracle 12c**: Grant SYSBACKUP privilege, or grant create session, resource, and sysbackup to the Oracle backup user.
- The connect string to connect to the database as sysdba (or sysbackup for Oracle 12g). You can find this string in the Oracle listener service ‘tnsnames’ file.

The same user and credentials will be used for all database backups specified in a Cohesity Protection Job.

NOTE: For release 6.1, we only support one standard user for DB authentication.

When performing a database restore to an *alternate* location, use OS authentication.

Recommended Approach

As you plan to deploy the Cohesity Oracle Adapter, Cohesity recommends you follow these best practices:

1. Assign sudoers to the Cohesity agent user.
2. Make Cohesity agent user part of the oracle oinstall and dba group.
3. Create a generic db_user for backups.

   NOTE: This user will have to be created on all databases and is only used for backups.

4. When restoring a database to an alternate location, use OS authentication.
5. Both the Oracle and Cohesity user agents should have permission to write to the `adump` and `diag` directories, the control file, and the database restore locations.

Install the Cohesity Oracle Adapter

Installing the Cohesity adapter for Oracle involves a set of sequential procedures:

1. Install the Cohesity Agent:
   a) Install Cohesity Linux Agent or
   b) Install Cohesity AIX Agent
2. Register the Oracle Database servers (Physical server or VM).
3. Register the Oracle Database instance.
4. Create a Cohesity Oracle Data Protection Job.
5. Set the QoS Policy for the Protection Job.

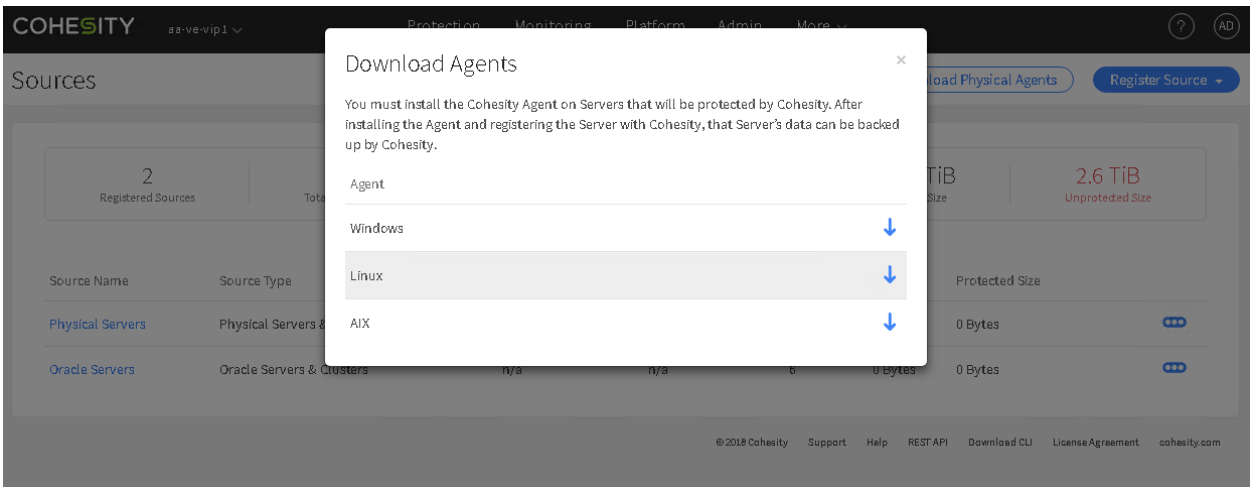
Install the Cohesity Agent

Depending on your choice of OS (Linux/AIX), Cohesity offers two different versions of agents which you can install from Cohesity Platform.

Install the Cohesity Linux Agent

Follow the steps below, to install the Cohesity Linux agent on RHEL/CentOS/OEL/SLES:

1. Log in to Cohesity and select **Protection > Sources > Download Physical Agents** and choose **Linux**.



NOTE: If you downloaded the agent onto a Windows desktop or MAC OS, use FTP or winSCP to move it onto the Oracle Database server that needs Cohesity data protection.

2. Add the following entry to the sudoers file: `Defaults:oracle_user !requiretty`
3. Grant sudo permission to the Oracle user.
4. Run the following command on each Oracle Database server that requires Cohesity protection:

```
$sudo ./cohesity_agent_6.0_linux_x64_installer -- --install -I
/home/oracle -S oracle -G oinstall -c 0
```

Where:

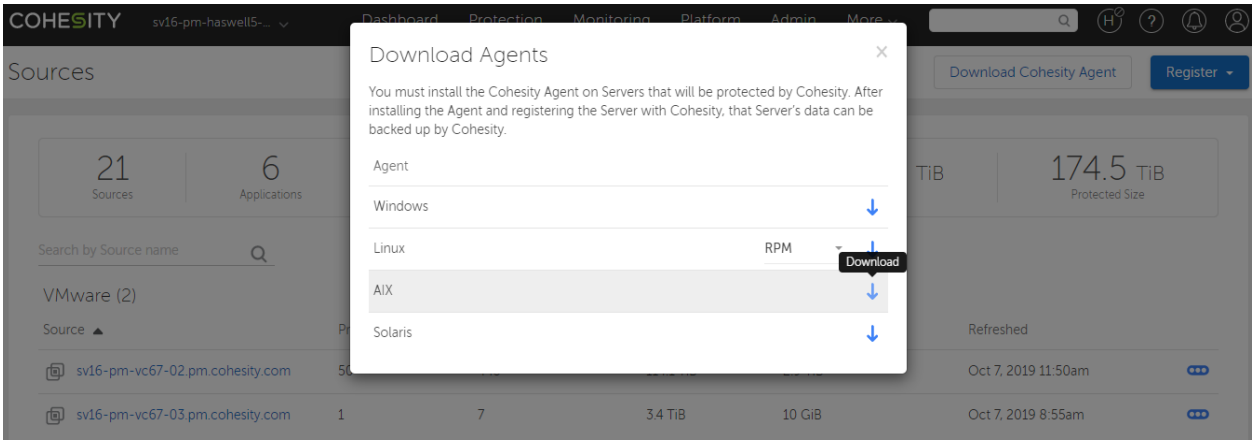
- **-**I is the agent installation directory.
- -S is the Oracle OS user.
- -G is the group where oracle_user is a member.
- -c prevents the user being created if it already exists.

NOTE: For RAC and VCS, the Cohesity Linux Agent must be installed on all Nodes that are part of an Oracle RAC/VCS.

Install the Cohesity AIX Agent

To install the Cohesity Oracle Adapter Agent for AIX:

1. Log in to Cohesity and select **Protection > Sources > Download Physical Agents** and choose **AIX**.



NOTE: If you downloaded the agent onto a Windows desktop or MAC OS, use FTP or winSCP to move it onto the Oracle Database server that needs Cohesity data protection.

2. Add the following entry to the sudoers file: `Defaults:oracle_user !requiretty`

3. Grant sudo permission to the Oracle user.
   Run the following command on each Oracle Database server that requires Cohesity protection:

```
$ installp -ad cohesity_agent_aix.bff all
```

Where:

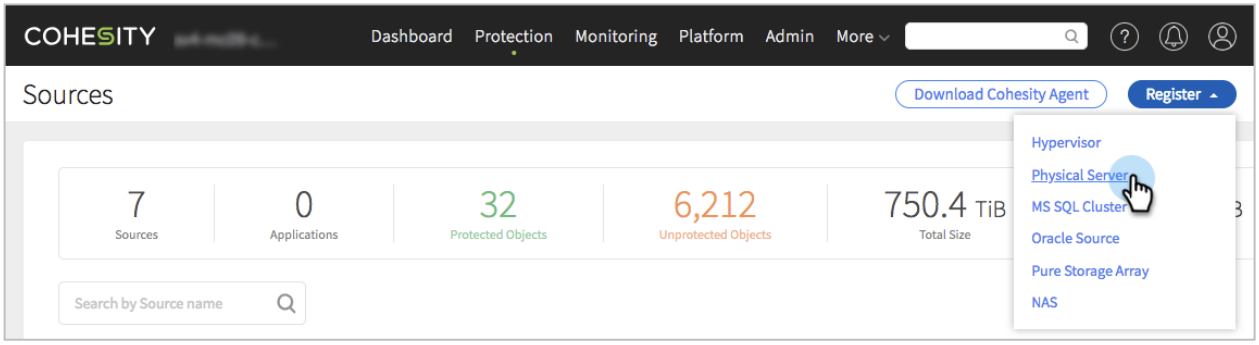
- a — Applies one or more software products or updates.
- d — Specifies where the installation media can be found.
- all — which indicates that all software contained on the specified installation media is to be installed.

NOTE: For RAC and VCS, the Cohesity Linux Agent must be installed on all nodes that are part of an Oracle RAC/VCS.

Register the Oracle Server

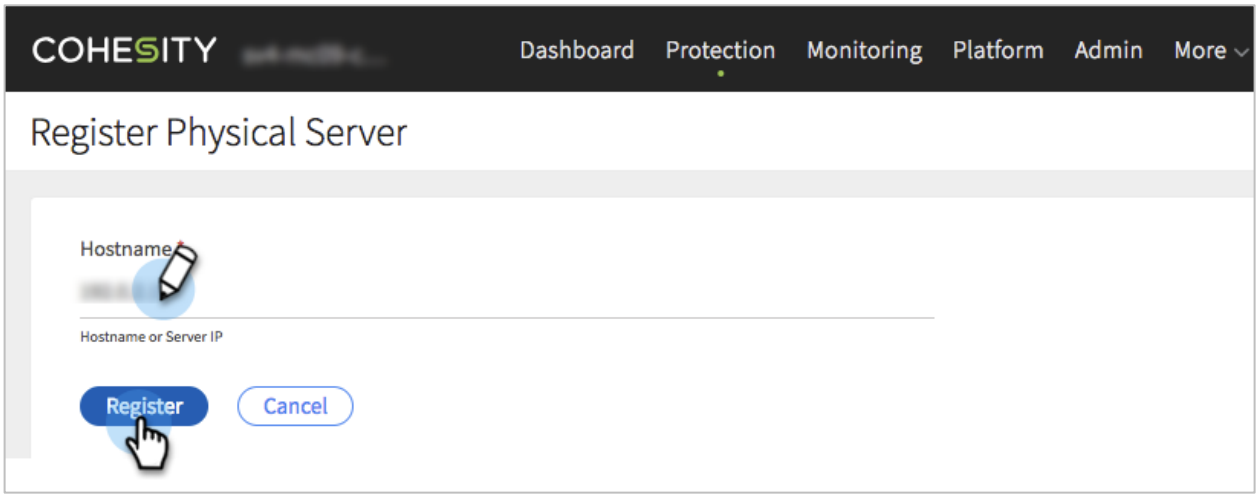
To register the Oracle Database server:

1. From the Cohesity Dashboard, select **Protection > Sources**. On the Sources page, click **Register > Physical Server**.



NOTE: In Cohesity, a 'physical server' refers to a bare metal server or a VM running the supported versions of the OS and Oracle Database, with the Cohesity Linux Agent installed. If you register Oracle without first selecting the 'Physical Server' here, you will have to populate some of the fields manually when you register the databases in the next step.

2. Enter the **Hostname** or **Server IP** address of the Oracle Database server and click **Register**.



   Once the registration is successful, the registered server appears in the list on the Sources page. However, Only one IP address or hostname can be entered in the field.

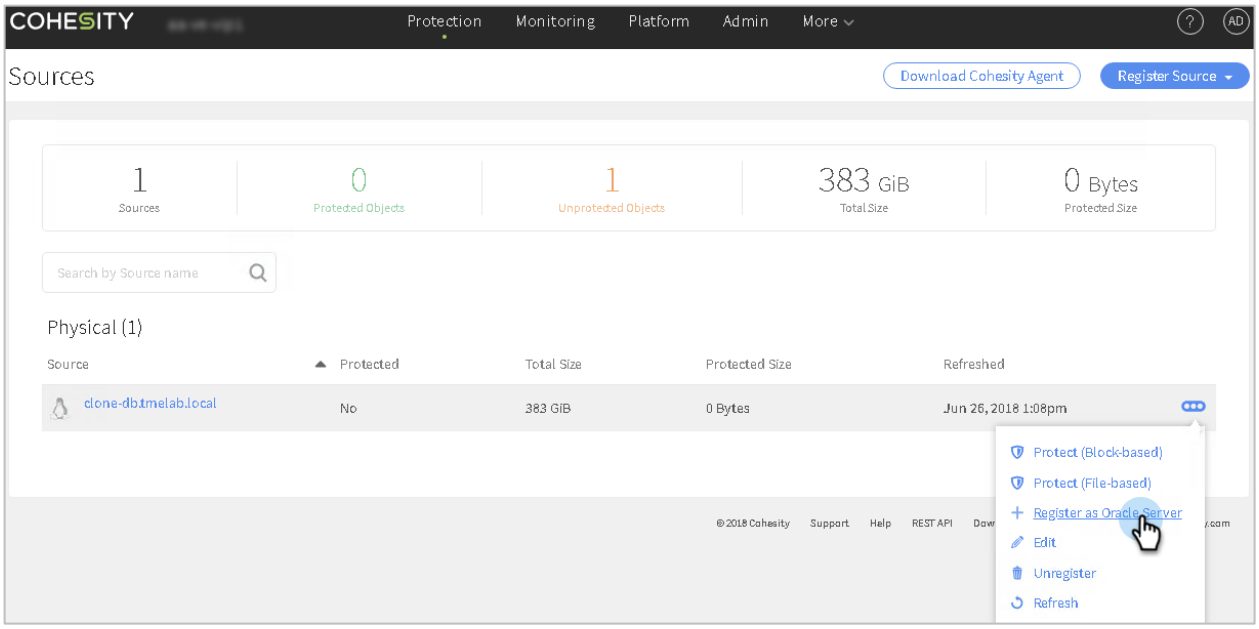
3. For RAC/VCS configurations, it's advised to use SCAP-NAME/SCAN-VIP for successful registration and protection of Oracle Database.

Register the Oracle Database Instance

Once the database server is registered, you're ready to register the database instance itself.

To register the database:

1. From the Cohesity Dashboard, select **Protection > Sources**. On the Sources page, find the server you added in the list. On the right, hover over the menu button for that source and select **Register as Oracle Server**.



   On the Register Oracle Source page that opens, some tables are populated automatically. Verify that the hostname is correct. If the **Selected Source Type** is incorrect, enter the correct type (Single Instance by default, or Oracle RAC).

> **NOTE**: If you registered without selecting Physical Server in the previous section, you will have to populate some of the fields in the Register Oracle Source page manually.

2. Choose your authentication preference: OS authentication (the default) or DB authentication. (For more on this choice, see Required Credentials and Privileges above.)

3. For RAC database, Cohesity Agent should be installed in all nodes which are pertaining to an Oracle RAC cluster where SCAN VIP is configured and to register Oracle RAC, user must use SCAP-NAME/SCAN-VIP, Otherwise, registration/backup/recovery of the Oracle databases will fail.

4. Click **Register**.

   If the Oracle Database registration is successful, the discovered Oracle Database instance host appears on the **Sources** page, under **Oracle**.

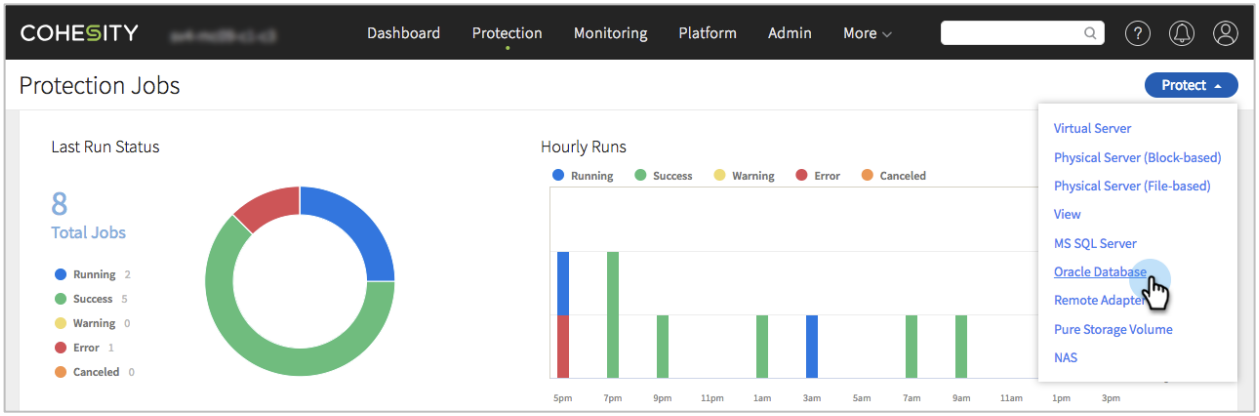
   To view the actual Oracle Database instances, click the source and open the **All Objects** tab.

Create a Cohesity Oracle Data Protection Job

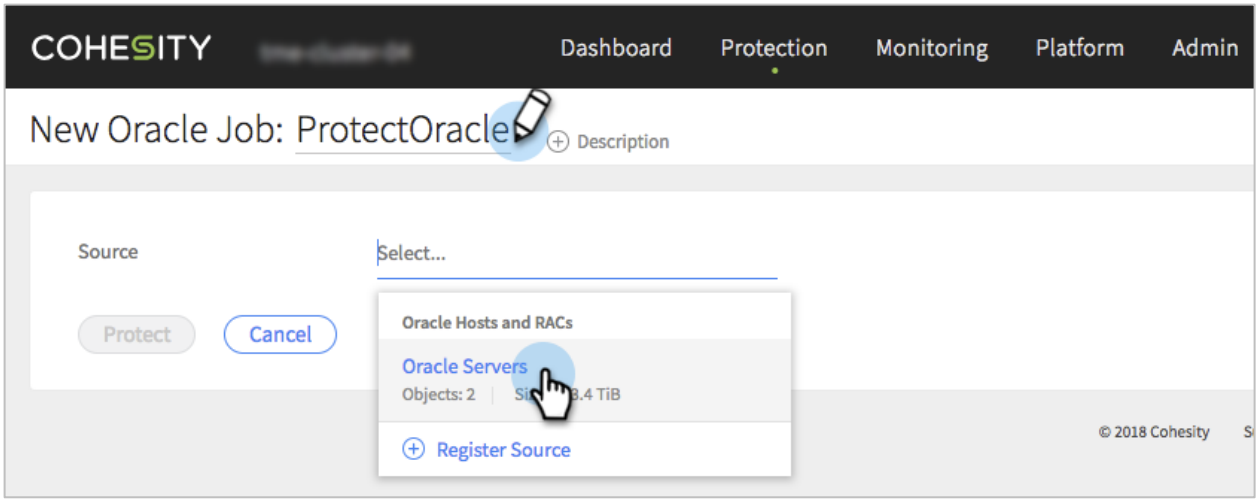
Now it's time to create a Cohesity Protection Job to protect the database.

To create a Protection Job:

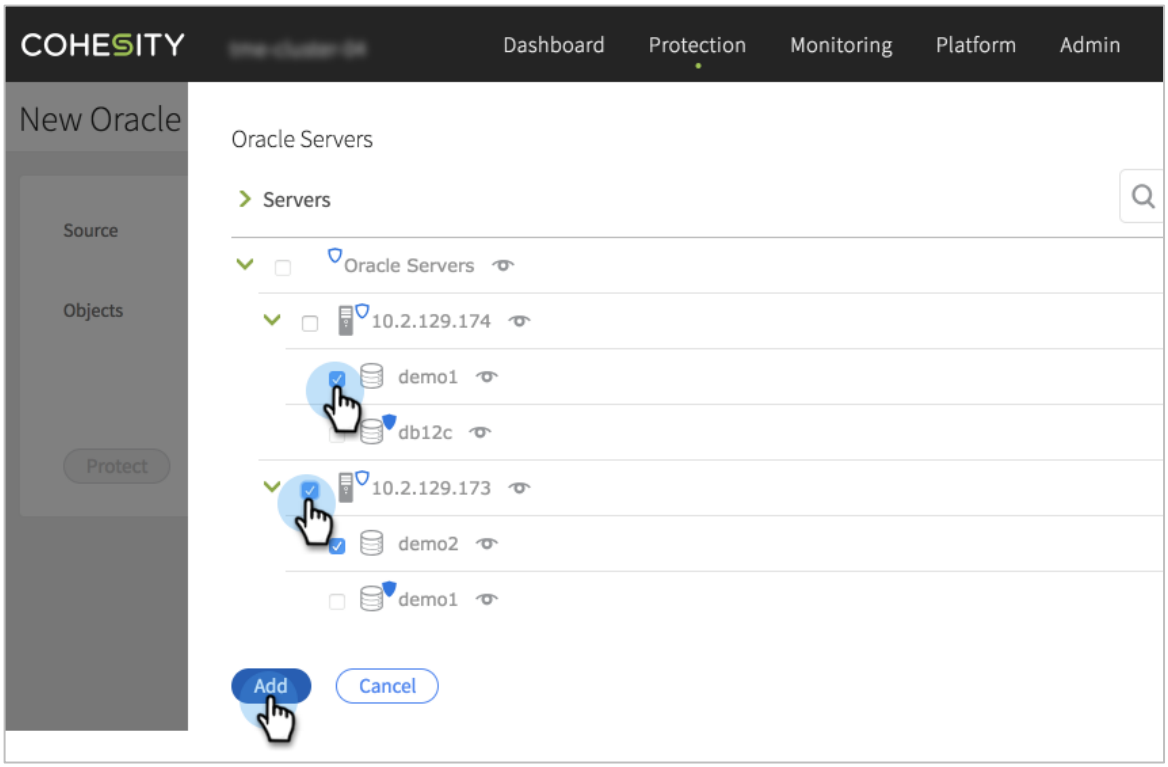
1. From the Cohesity Dashboard, select **Protection > Protection Jobs**. On the Protection Jobs page, click **Protect > Oracle Database**.



2. On the **New Oracle Job** page, enter the job title and select the **Source**.

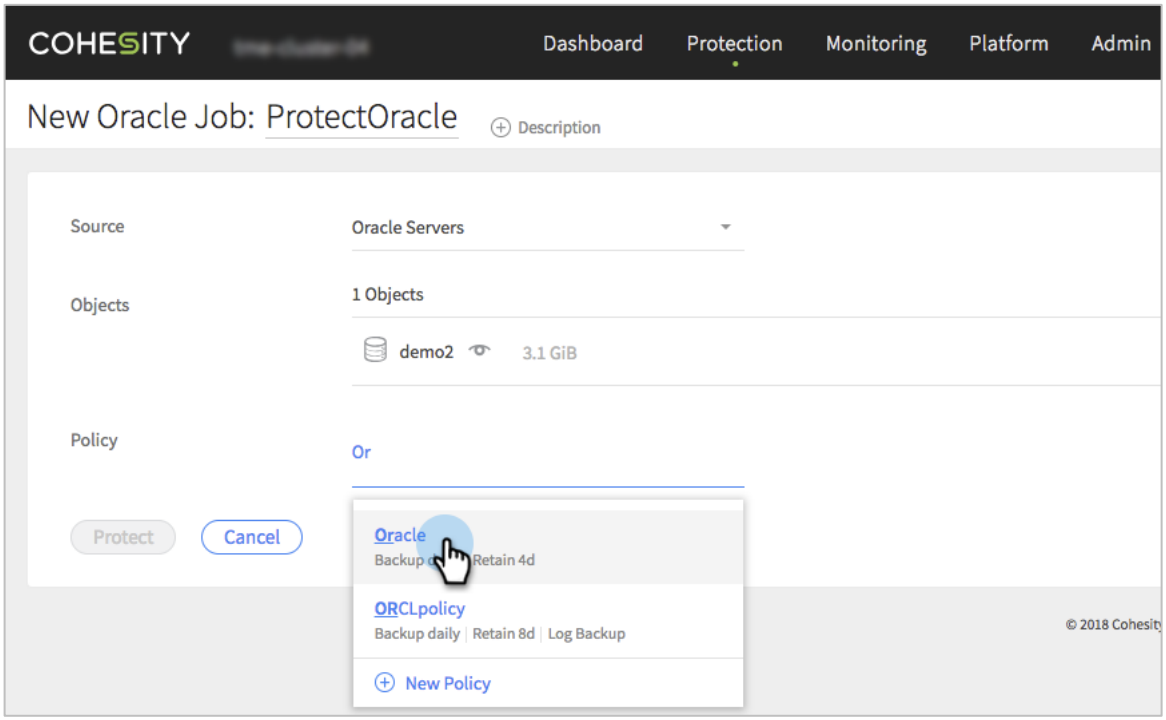


3. On the **Oracle Servers** page that opens, select the database instances to protect, and then click **Add**.



4. The New Oracle Job page returns. Under **Objects**, verify that the database instances you just selected are listed.

5. Set the Quality of Service (QoS) **Policy** for your Protection Job. Under **Policy**, select the appropriate Protection Policy, or choose one of the defaults (Bronze, Silver, or Gold), or create a **New Policy** if those available do not meet your schedule and retention needs.



QoS policies are designed for workloads such as backups, which keep a lot of IOs outstanding. Cohesity recommends one of the following:

Backup Target HDD (Default): This policy assumes higher latency is OK and the workload will get high throughput if it issues a lot of writes in parallel. The data typically lands on hard disks if it is sequential and on SSDs if it is random writes.

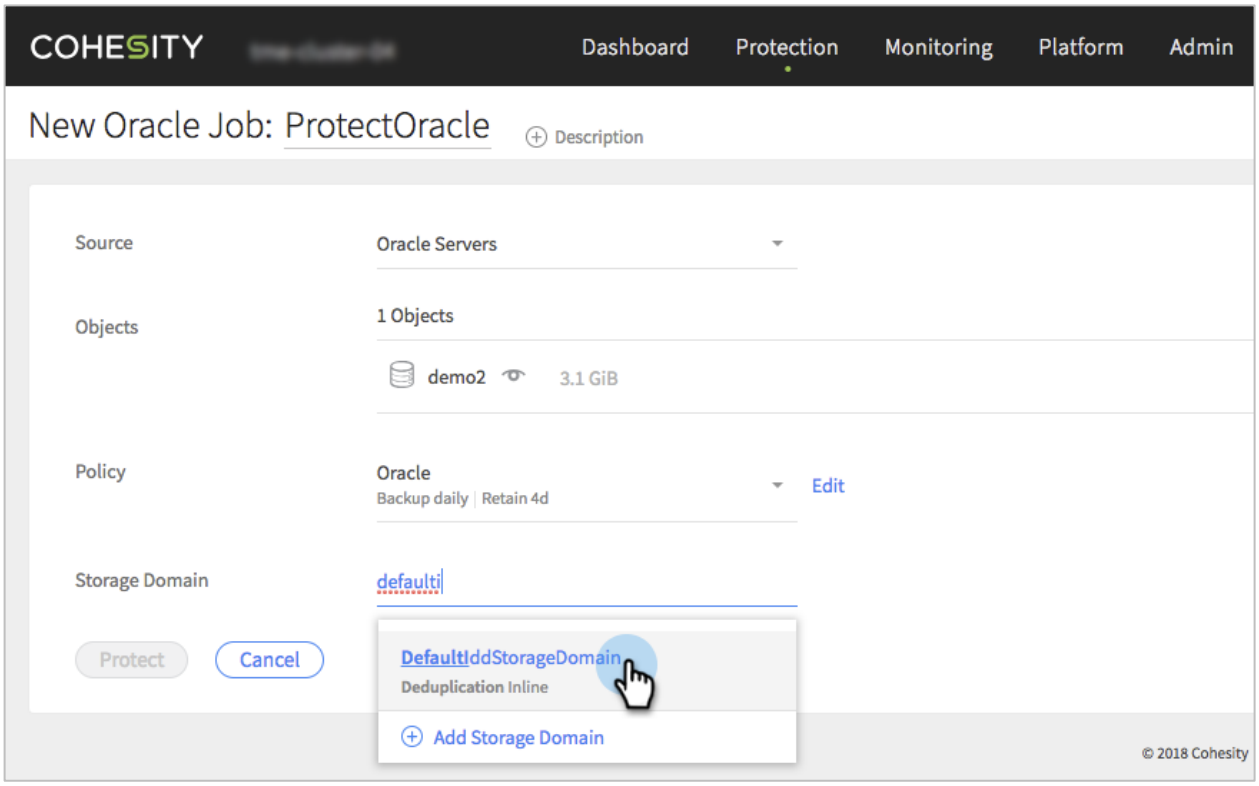
Backup Target SSD: With this policy, both sequential and random IOs land on SSDs, but the IOs are not treated as high-priority IOs and Cohesity assumes the workload keeps lots of IOs outstanding. For best performance, Cohesity recommends the Backup Target SSD policy. If necessary, you can change the policy at any time later.

NOTE: If you create a **New Policy**, define the policy based on data-protection requirements that can be shared across Protection Jobs. Also, follow a naming scheme that allows easy identification of policy attributes.

6. Select the appropriate **Storage Domain**.

   A Storage Domain defines the level of storage efficiency, security, and resilience for the data being protected in Cohesity Platform. In a Storage Domain, you can enable or disable deduplication, compression, and/or encryption for the domain, and select the necessary level of resilience (that is, number of faults tolerated).

   If a Storage Domain with the right settings exists, select that one. Otherwise, create a new one with **Add Storage Domain**.

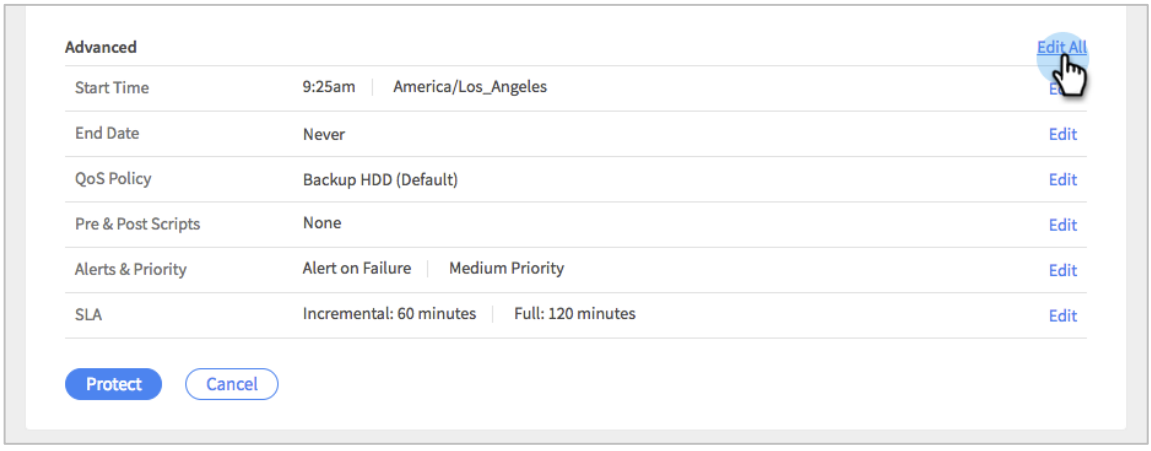


   If you need to create a new Storage Domain, note that, while you can enable or disable deduplication and compression at any time, you can only enable or disable encryption when you first create the Storage Domain. Because deduplication and compression produce no gain when performed on encrypted data, Cohesity recommends these settings if you create a new Storage Domain:

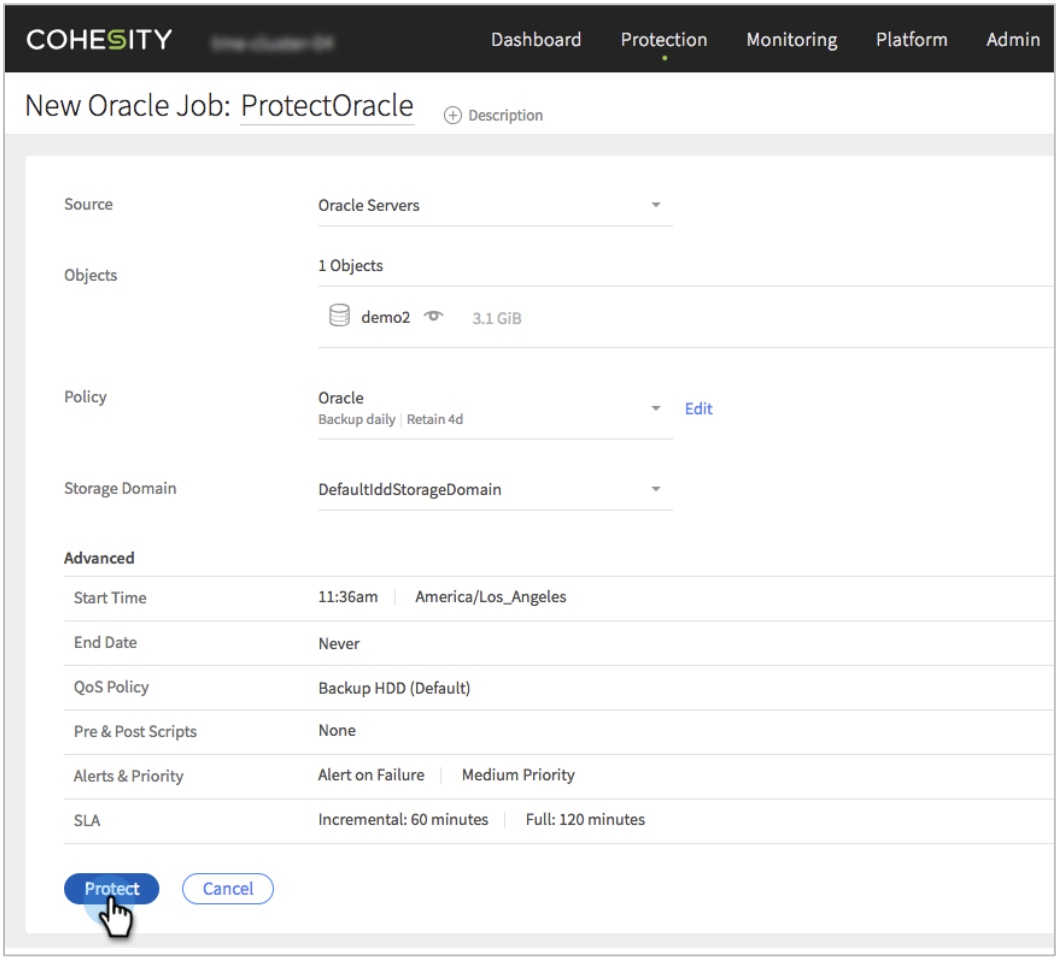
Table 4: Encryption Types and Their Impacts

ENCRYPTION TYPE	DETAILS	COHESITY'S RECOMMENDATION	Storage Domain Settings
Cohesity Encryption	This at-rest encryption has no impact on deduplication and compression of backup data.	Yes	Enable inline deduplication and compression.
Transparent Data Encryption	With this file-level encryption on databases (on both storage and backup media), deduplication and compression produce no gains.	No	Disable deduplication and compression.
RMAN encrypted backups	RMAN creates backups that are encrypted. Deduplication and compression produce no gains when using RMAN encrypted backups	No	Disable deduplication and compression.

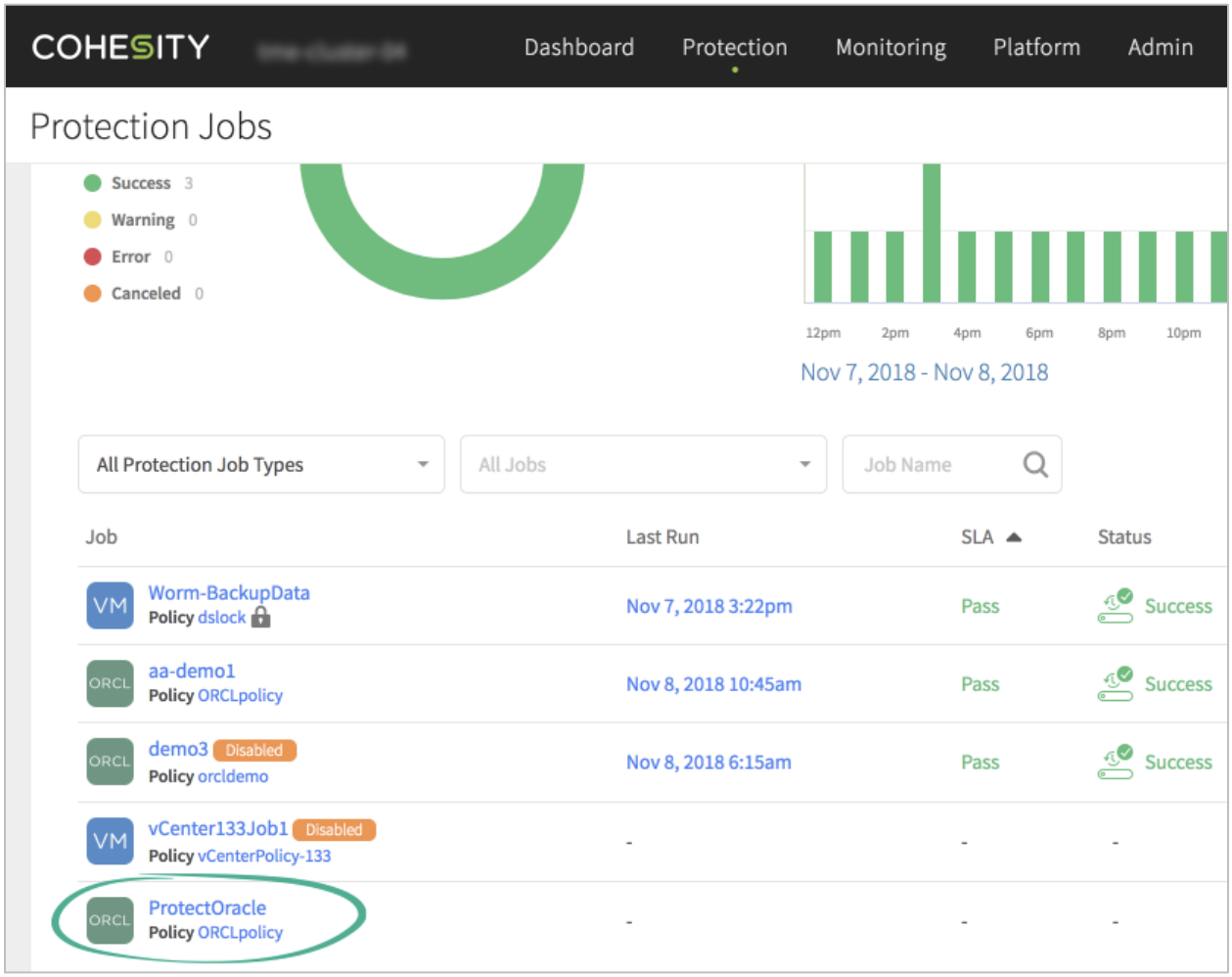
7. After selecting a Storage Domain, if you need to change any of the **Advanced** settings on the New Oracle Job page, scroll down and click **Edit All** on the right.



8. Click **Protect**.



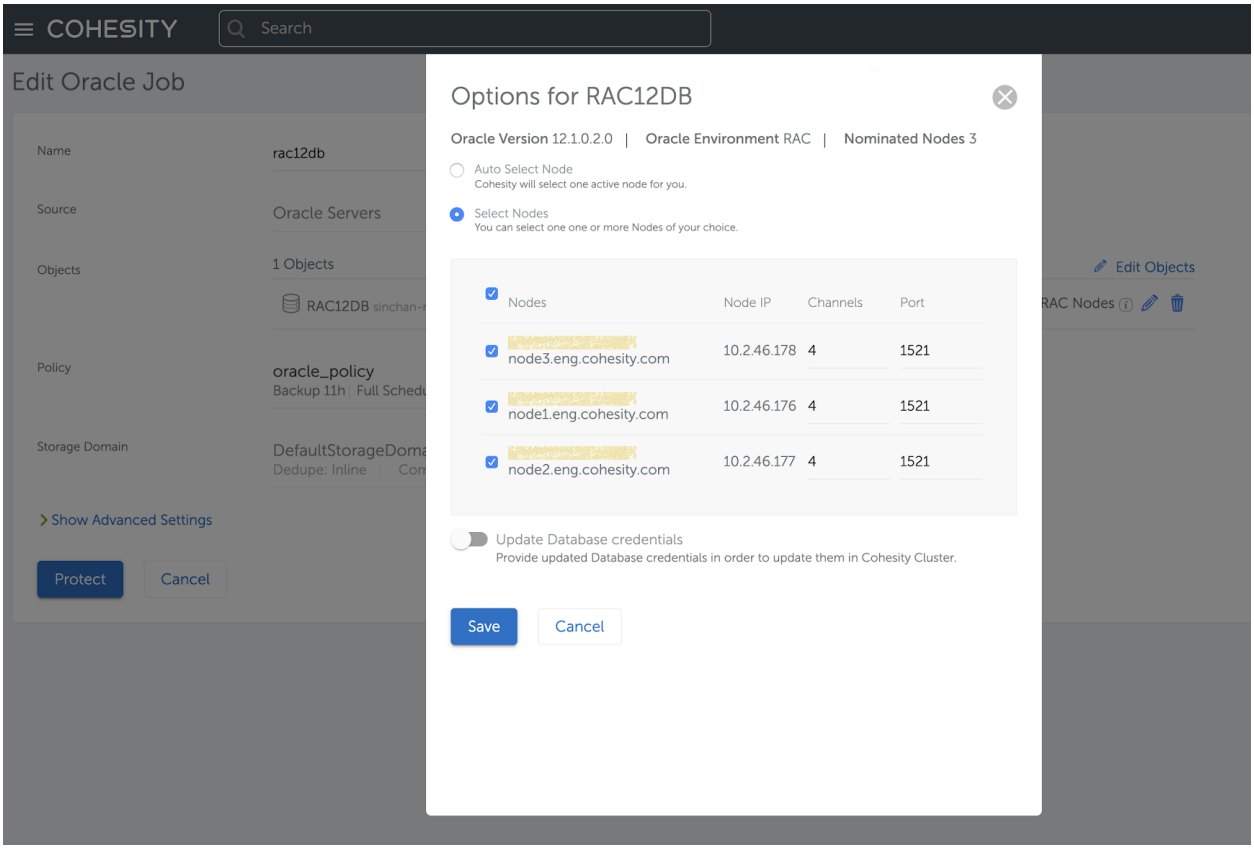
Your Protection Job is set to run, and appears in the list on the Protection Jobs page.



Once your Protection Job completes its first run, you will be ready for database recovery, replication, and archival, but first, it helps to understand the workflow, described in the next sections.

Multi-channel and Multi-node Backup Option

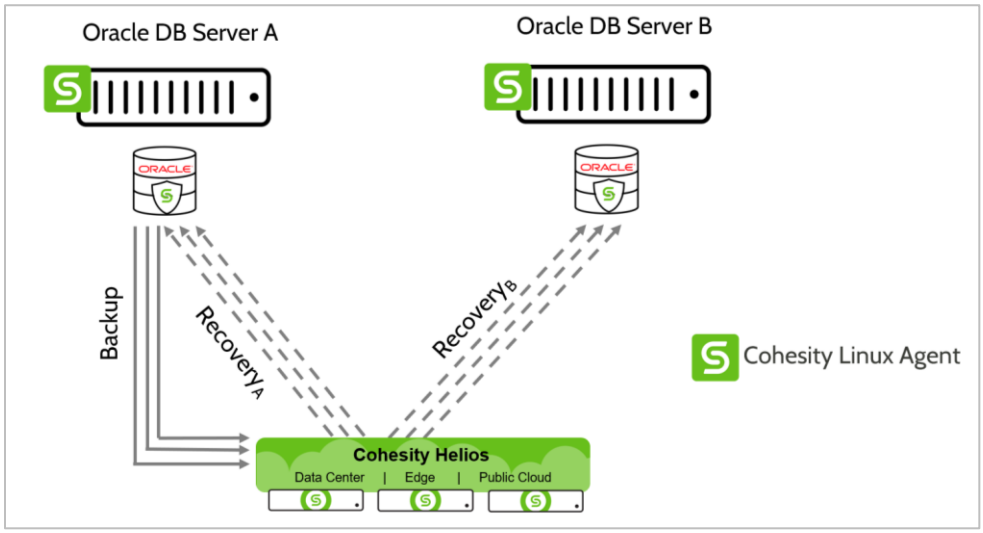
Once the Database server has been selected for protection, you can significantly improve the backup times by opting for parallelism (data transfer through RMAN channels) and selecting the number of nodes (only for RAC) to distribute the load.



Cohesity Oracle Database Recovery

The restore process is straightforward; there are no RMAN scripts to deal with and everything is accomplished through the Cohesity user interface. You can recover to the original Oracle DB server or to a new one:

Figure 4: Cohesity Oracle Adapter Recovery to Original or New Server



Oracle Database Restore Types

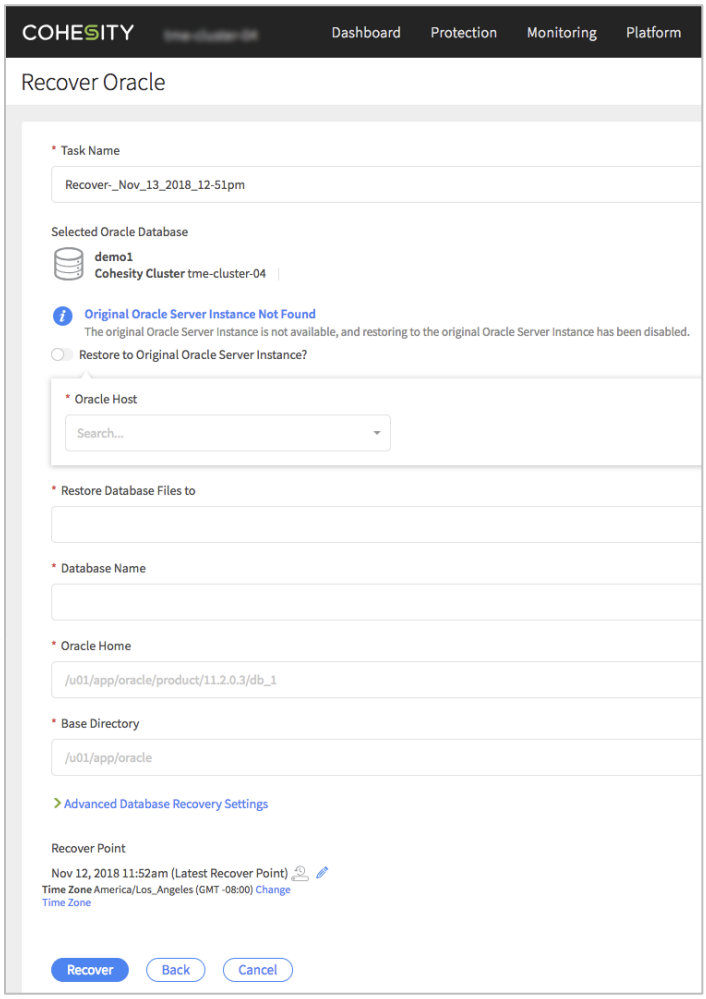
- Restore to original Oracle server instance and overwrite the original database.
- Restore to original Oracle server instance as a new database.
- Restore to an alternate Oracle server.
- Database point-in-time recovery (DBPITR).
- Restore Oracle Database in recovery mode.

NOTE: If restoring to an alternate Oracle server, the Oracle server must be registered with Cohesity Platform beforehand. Cohesity also expects the Oracle software installed on the Oracle server.

To recover a recent or previous backup of an Oracle Database, from the Cohesity UI, select **Protection** > **Recovery > Recover > Oracle**. In the Recover page, you can search for the Oracle instance that needs to be restored or use a wildcard name to search for a specific all database instance.

Once the desired instance has been selected for recovery, the following items in the Recover page need to be filled out in order for the recovery to proceed: When recovering an Oracle Database, you are prompted to make several choices:

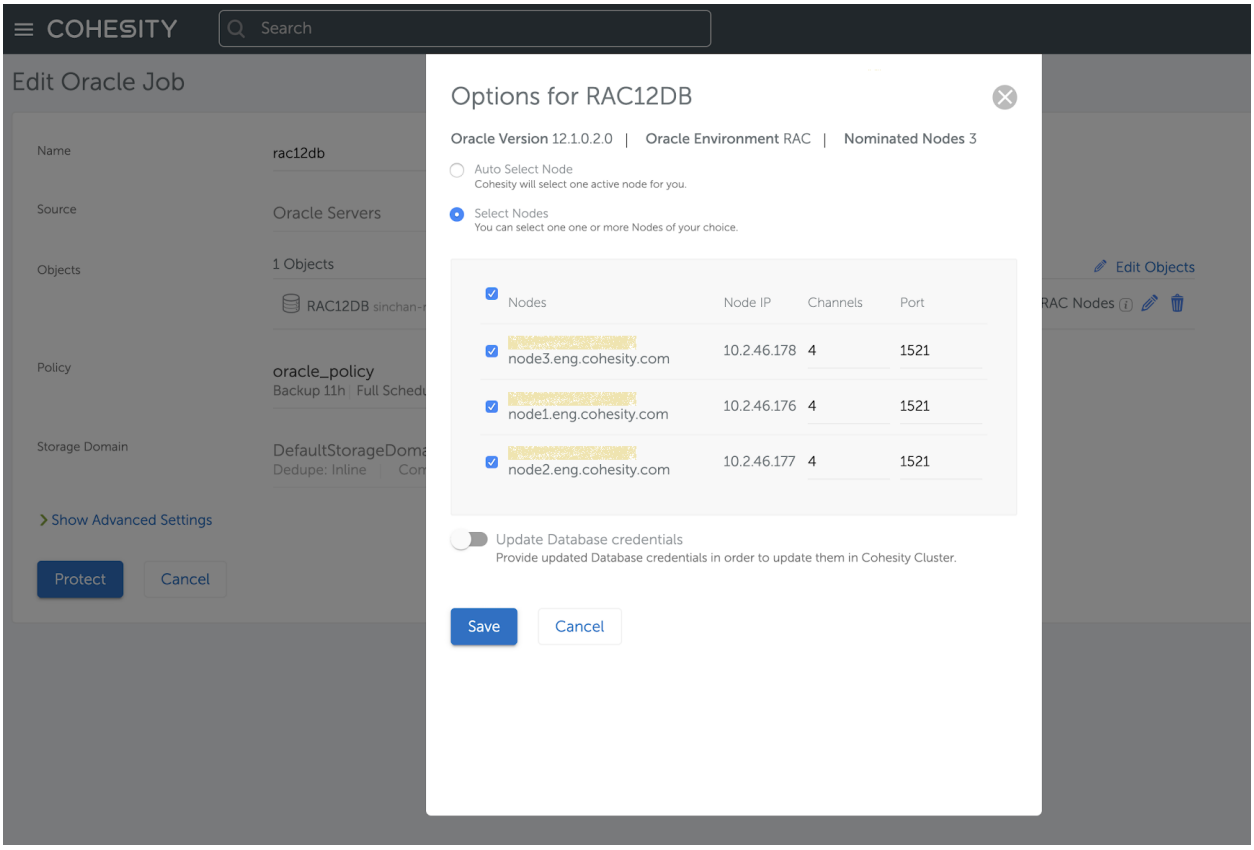
- **Oracle Host**: Necessary when restoring to a new host server. If you are restoring to the original Oracle server, this field is disabled.
- **Restore Database Files to**: If the Oracle server is configured with ASM, this field requires the directory path with the syntax: +DATA. If the target is non ASM then exact path to the volume is required.
- **Database Name**: Limited to eight characters, and cannot be an existing database name.
- **Oracle Home**: Typically, the directory in the ORACLE_HOME oracle user profile.
- **Base Directory**: Typically, the directory in the ORACLE_BASE oracle user profile.
- **Recovery Point**: If there are multiple backups that were taken, the user can expand this list to select which point in time the user wants to restore the Oracle Database.



When you complete this form, click **Recover** to initiate the restore process.

Multi-channel and Multi-node Recovery Option

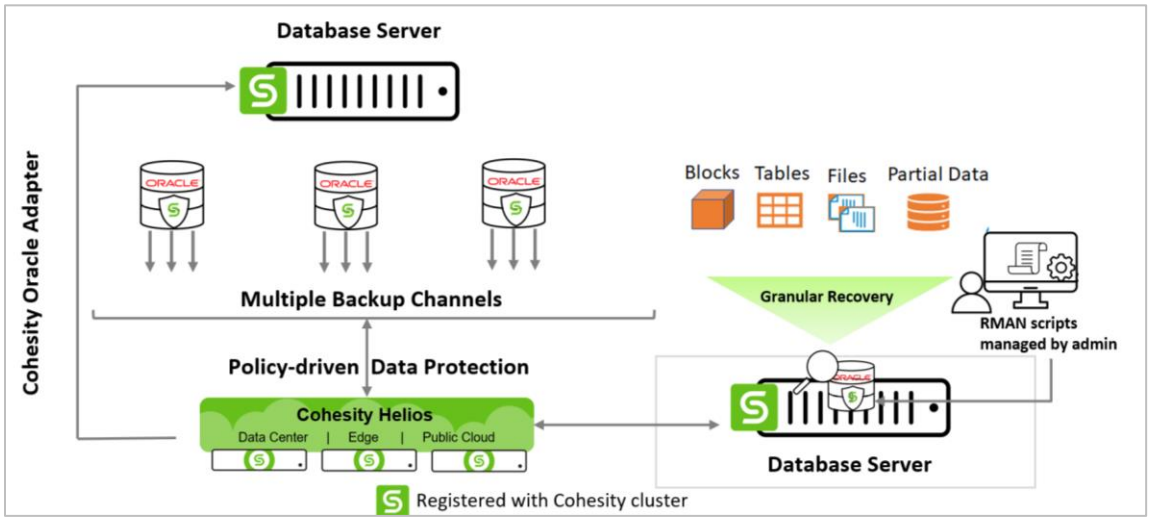
Once the database server has been selected for recovery, you can significantly improve the restore times by opting for parallelism (data transfer through RMAN channels) and selecting the number of nodes (only for RAC) to distribute the load.



Granular Recovery

Granular recovery of Oracle databases can be done by mounting the corresponding NFS Views from Cohesity. Once the Oracle server is protected by Cohesity Platform, you can leverage the simplicity of Platform UI to easily mount the NFS View in which the RMAN backup files are stored as an image copy.

Figure 5: Cohesity NFS View Exposed to Original or New Server

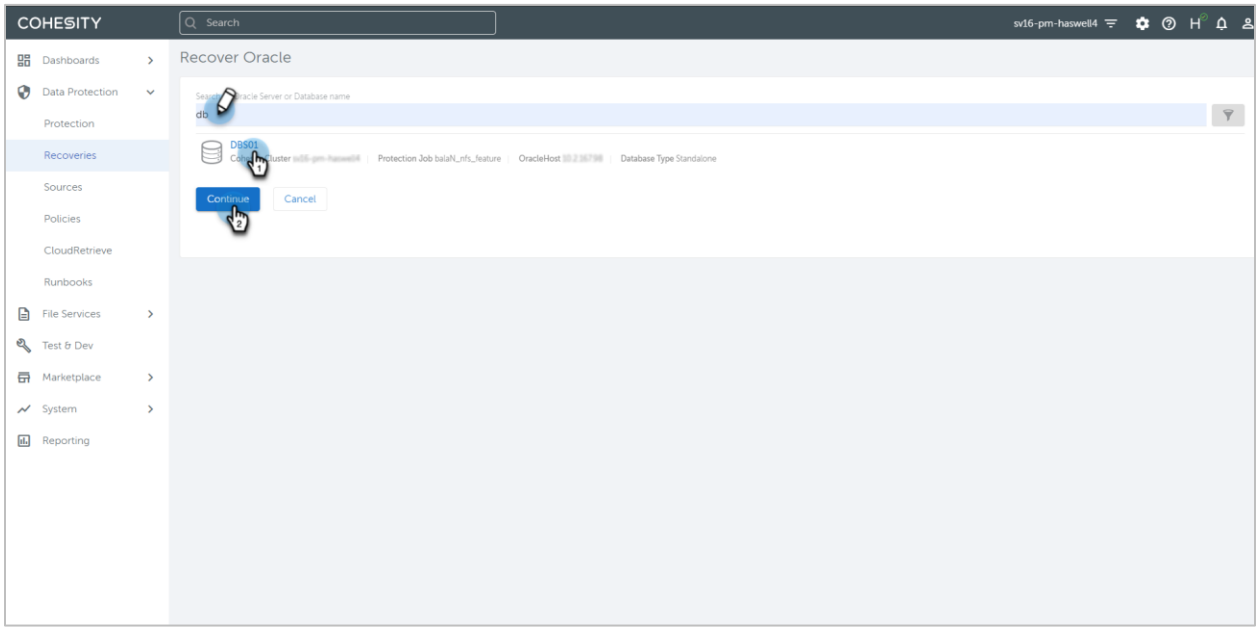


The following recovery actions could be achieved via NFS mount:

- Block-level recovery
- Table and tablespace recovery
- DB files (controlfile, datafile, SPfile) recovery
- Partial data recovery
- Cohesity backup view exposing as NFS through UI.
- Restore Oracle Database in recovery mode.

NOTE: To mount NFS to an alternate Oracle server, you must first register the Oracle server with Cohesity Platform. Cohesity also expects the Oracle software installed on the Oracle server and Recovery Admin should be self-sufficient with RMAN scripts needed for granular recovery. This feature requires Cohesity Linux Agent with version 6.4 and above.

To recover a recent or previous backup of an Oracle database from the Cohesity UI, select **Protection > Recovery > Recover > Oracle**. In the **Recover Oracle** page, search for the protected Oracle instance and select the database planned for granular recovery or use a wildcard name to search for a specific all database instance.



Once the desired instance is selected for recovery, the following items in the Recover page need to be filled out in order to proceed with the recovery View expose task.

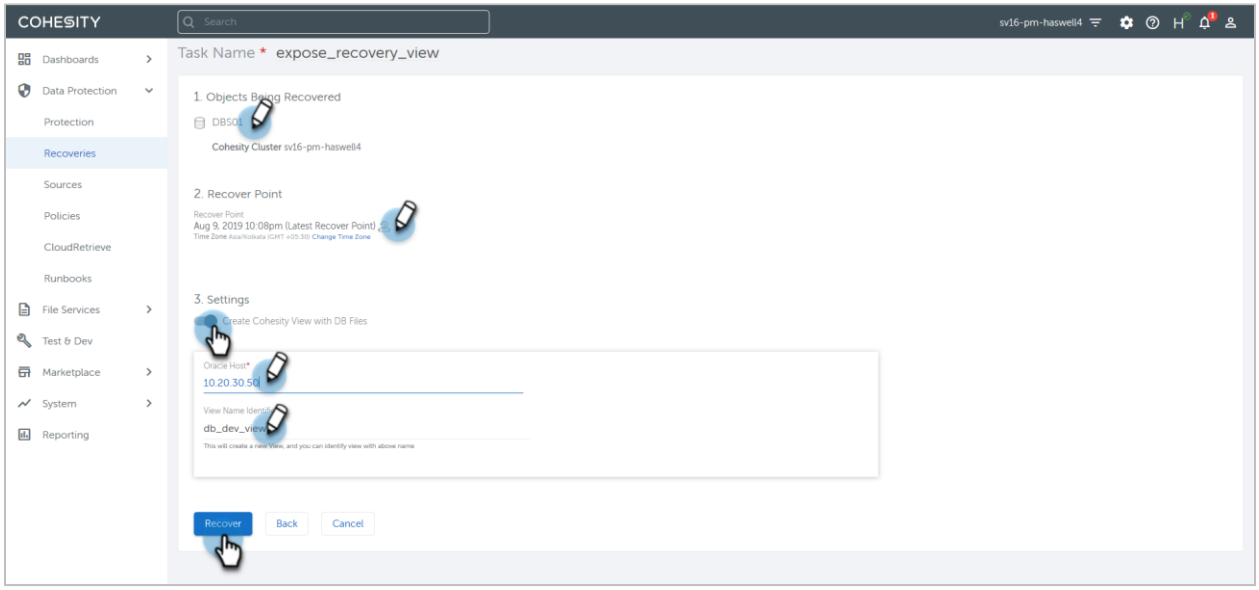


Table 5: Recovery Page Fields and Definitions

FIELD	DEFINITION
Objects Being Recovered	Under **Objects Being Recovered**, verify that the database instances you have selected are listed.
Recover Point	If there are multiple backups taken, select the point of time that corresponds to the desired restore point from the drop-down.
Settings	Toggle **"Create Cohesity View with DB Files" o**n and enter the server details to which Recovery View NFS mount is planned.
Oracle Host	From the drop-down list, browse for the host which is registered as Oracle server. View expose works only with registered Oracle servers.
View Name Identifier	Provide a name to identify the View at the target server when mounted successfully.
Objects Being Recovered	Under **Objects Being Recovered**, verify that the database instances you have selected are listed.

Once all the fields are filled, click **Recover** to restore your Oracle server.

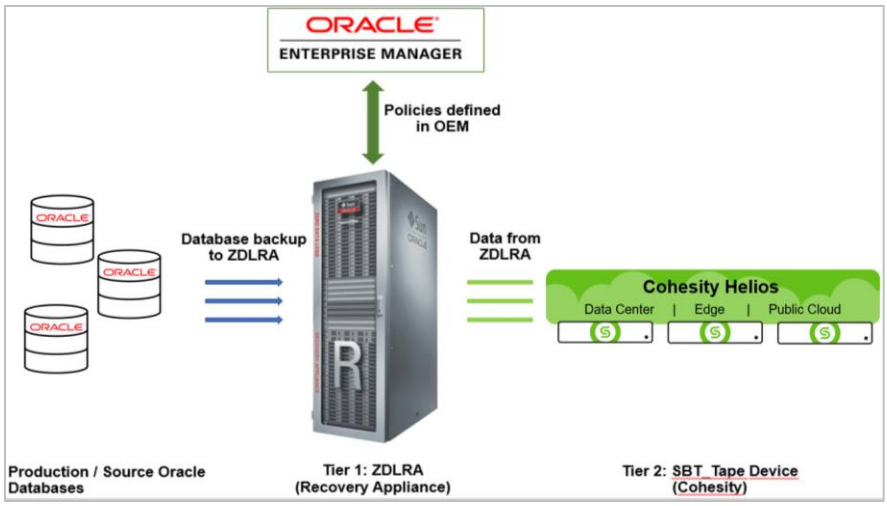
Cohesity Oracle SBT Plug-in

The Cohesity Oracle SBT Plug-in allows customers to use Cohesity Platform as their extended storage for RMAN backups. Oracle RMAN Backup leverages Cohesity SBT plugin to save dumps into Cohesity Storage.

Oracle ZDLRA Backup and Restore Using SBT

The Cohesity Oracle SBT plug-in allows offloading of data from Oracle Zero Data Loss Recovery Appliance (ZDLRA) or Oracle databases running on non-ZDLRA appliances to Cohesity Platform leveraging Cohesity SBT plugin.

Figure 6: Oracle ZDLRA Backups Using Cohesity



Backup Oracle ZDLRA Using Cohesity SBT Plug-in

An Oracle Zero Data Loss Recovery Appliance (ZDLRA) can be backed up to Cohesity Platform using Cohesity Oracle SBT plugin with the following steps:

1. Create and mount a Cohesity view to ZDLRA where backups are written to.
2. Configure the RMAN job and specify the SBT library path and Cohesity view RMAN jobs writes to.
3. Set the policies in Oracle Enterprise Manager (OEM) for RMAN backup job schedule or leverage Remote Adapter to create the job. And track all backup tasks, schedules, and alerts from Cohesity Platform Management UI. Also, manage QoS policy for workload optimization.

   For detailed procedure, see Cohesity Oracle SBT Plugin in the online help.

Restore Oracle ZDLRA Restore Using Cohesity SBT Plug-in

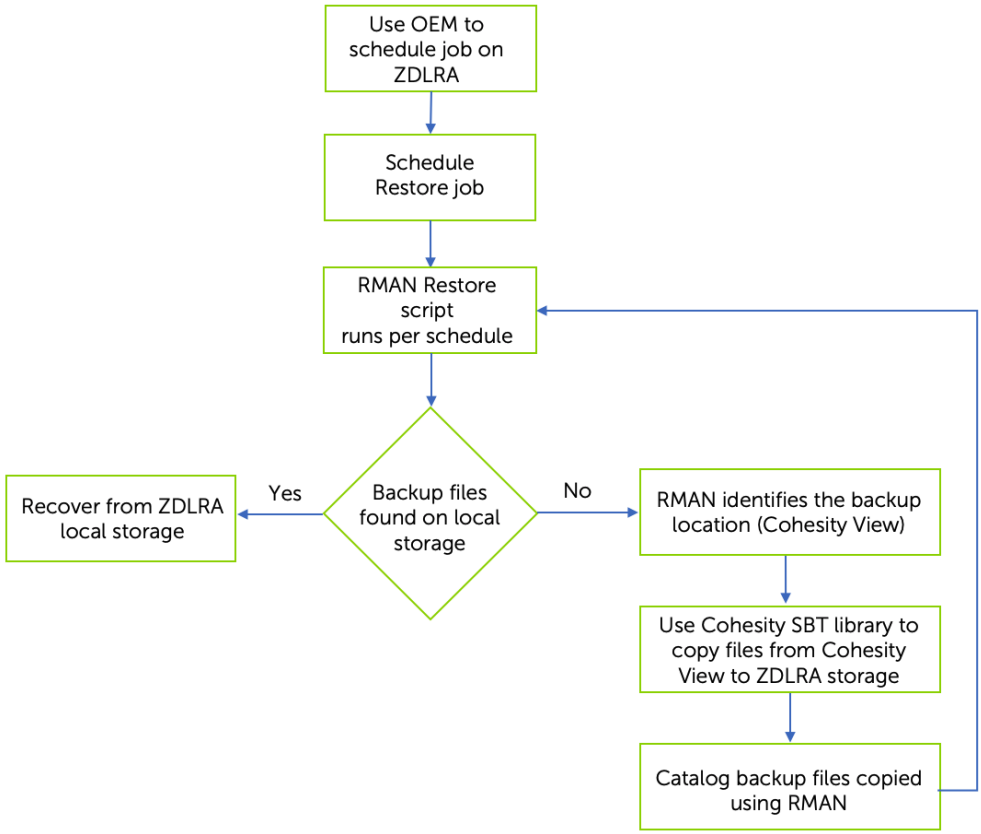
Recovery of ZDLRA databases is initiated by configuring an RMAN backup job via OEM (Oracle Enterprise Manager). RMAN starts the recovery from last backup dumps saved on the ZDLRA appliance. In case the backup sets are not available on the ZDLRA appliance and the RMAN catalog indicates the backupset is on a Cohesity View, the recovery would be triggered from the Cohesity storage attached to the ZDLRA.

Recovery workflow:

1. RMAN retrieves backup from the Cohesity View attached to the ZDLRA appliance leveraging the Cohesity SBT library plugin.

2. RMAN starts recovery from ZDLRA via the backups copied to the ZDLRA in the step above.

Figure 7: Oracle Database Recovery on ZDLRA Appliance using Cohesity SBT



Note: If backup sets are not available on ZDLRA local disk, then customer has to catalog the backups available in the Cohesity View leveraging RMAN catalog command. Once the backups have been cataloged, the backup can be restored.

Disaster Recovery and Business Continuity

Data and application retention is critical to organizations that seek to prevent data loss and meet security, legal, compliance, disaster-recovery, and business-continuity requirements. Cohesity provides two solutions for disaster recovery and business continuity:

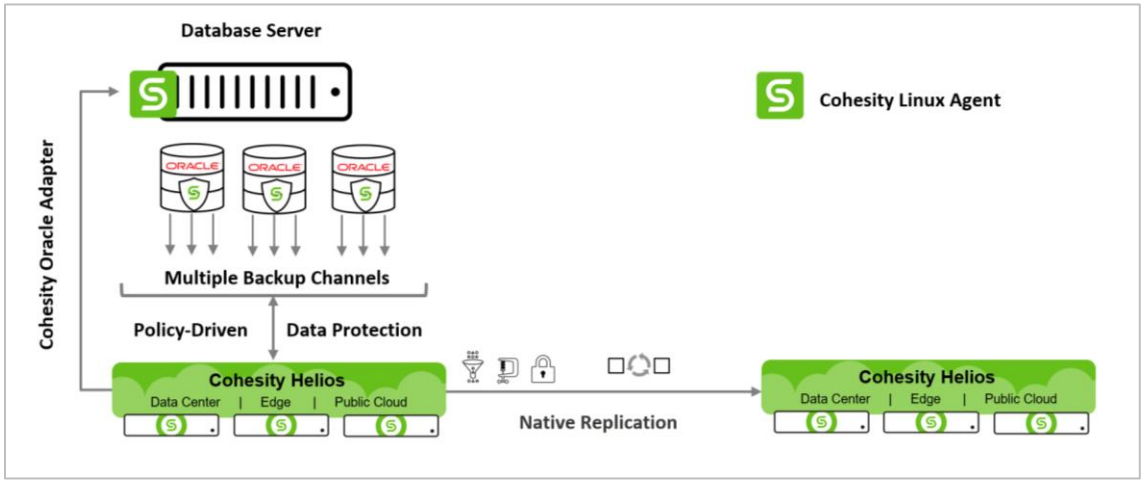
- Replication
- CloudArchive and Retrieval

Replication

Oracle application administrators can take advantage of Cohesity replication for cost-effective disaster recovery (DR). Cohesity Platform provides a policy-based data replication solution from the core to the cloud to the edge, from one cluster to another cluster in your DR site. As part of replication, Cohesity always performs source-side compression and deduplication first, and sends only the changed data over the network.

In the event that the primary site becomes unavailable, application and backup admins can failover to the DR site for backup and recovery of their data.

Figure 8: Cohesity Native Replication to New Cluster (on-premises or Cloud Edition)



Cohesity CloudArchive and Retrieval

The exponential growth of data volumes and the resulting IT management demands have prompted businesses to seek out more cost-effective, reliable data storage and protection solutions, especially in environments deploying Oracle Databases that can grow to terabytes and even petabytes in size.

Cohesity Platform provides a policy-based method to archive to public clouds (AWS, Azure, GCP, and Oracle Cloud), to any S3-compatible storage, to tape, and/or to any NFS mount point. Oracle application administrators can take advantage of Cohesity CloudArchive to address long-term data retention requirements, and can choose to restore an entire database or recover individual database files. The archived data is efficiently transferred and stored by sending only deduplicated, compressed incremental backups, thereby reducing network and storage utilization.

Once the data is archived, Oracle application administrators can also take advantage of Cohesity CloudRetrieve feature as a cost-effective alternative for disaster recovery, geo-redundancy, and business continuity. In the event the cluster you archived from becomes unavailable, you can retrieve your data onto a different cluster.

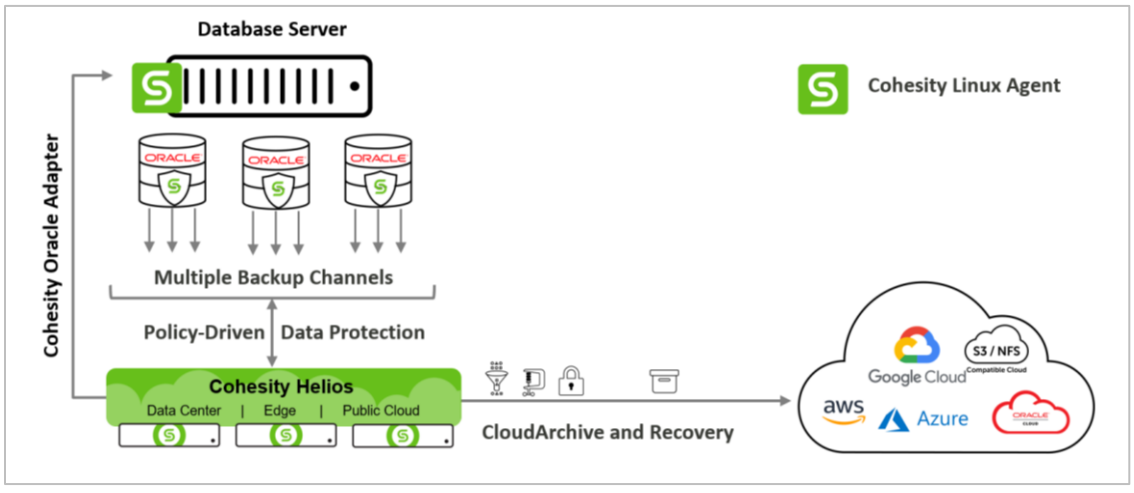
This approach involves three steps:

1. Register the cloud container (where you archived the Oracle Database) with the new Cohesity cluster.

2. Provide the search parameters, such as cluster name, Protection Job name, and date range.

3. Cohesity retrieves the data that you specify onto the new cluster, based on your search of the metadata.

Figure 9: Cohesity CloudArchive Connects Cohesity Backups to Cloud Storage



Conclusion

Cohesity provides a simplified and scalable platform to solve the dual challenges of increasing database sizes and decreasing backup windows for Oracle Database environments. In addition, Cohesity Oracle Adapter simplifies and centralizes Oracle Database protection and management, thereby eliminating the RMAN scripting task. With Cohesity, you can not only perform faster Oracle backups more reliably with less exposure to scripting errors, but also deliver significant TCO savings through a more efficient and consolidated infrastructure.

Your Feedback

Was this document helpful? Send us your feedback!

About the Authors

Balarameshwar Naik is a solution Engineer at Cohesity. In his role, Balarameshwar focuses on Enterprise databases, and data protection with Enterprise Cloud Storage.

Document Version History

VERSION	DATE	DOCUMENT HISTORY
1.3	July 2024	Republishing
1.2	Aug 2019	Added Granular Recovery
1.1	Jan 2019	Formatting update
1.0	Nov 2018	First release

ABOUT COHESITY

Cohesity is a leader in AI-powered data security and management. Aided by an extensive ecosystem of partners, Cohesity makes it easier to protect, manage, and get value from data – across the data center, edge, and cloud. Cohesity helps organizations defend against cybersecurity threats with comprehensive data security and management capabilities, including immutable backup snapshots, AI-based threat detection, monitoring for malicious behavior, and rapid recovery at scale. Cohesity solutions are delivered as a service, self-managed, or provided by a Cohesity-powered partner. Cohesity is headquartered in San Jose, CA, and is trusted by the world's largest enterprises, including six of the Fortune 10 and 44 of the Fortune 100.

Visit our website and blog, follow us on Twitter and LinkedIn and like us on Facebook.